NTNU
Innovation and Creativity

A Routing Protocol for MANETs

Luis Gironés Quesada

Master of Science in Communication Technology
Submission date: May 2007
Supervisor: Yuming Jiang, ITEM

Norwegian University of Science and Technology
Department of Telematics

Problem Description

Mobile ad hoc networks (MANETs) are an important type of networks, which have attracted a lot of research interest. For MANETs, many routing protocols have been proposed in the literature. They include AODV (Ad hoc On Demand Distance Vector) and OLSR (Optimize Link State Routing protocol), where AODV is a representative reactive routing protocol and OLSR is a representative proactive routing protocol. It has been studied in the literature that reactive and proactive routing protocols have their own advantages and disadvantages. As a consequence, they suit different network scenarios.

The main purpose of this thesis work is to propose a routing protocol for MANET, which integrates reactive and proactive routing protocols so as to make use their advantages and suit wider network environments. A second goal of the thesis is to conduct a comprehensive review of existing routing protocols for MANETs. In addition, a comparison between reactive, proactive and the proposed hybrid routing protocols for MANETs will be presented.

Assignment given: 10. November 2006
Supervisor: Yuming Jiang, ITEM

NORWEGIAN UNIVERSITY OF SCIENCE AND TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY, MATHEMATICS AND ELECTRICAL ENGINEERING
DEPARTMENT OF TELEMATICS

MASTER'S THESIS

Student's name:	Luis Gironés Quesada
Area of study:	Telematics
Title:	**A Routing Protocol for MANETs**

Start date:	November 10, 2006
Deadline:	May 10, 2007
Handed in:	May 8, 2006
Department:	NTNU Department of Telematics
Supervisor:	Professor Yuming Jiang

Trondheim, 8th of May, 2007

Yuming Jiang
Professor

**Thanks to my parents, to my girlfriend
Ana Belén, to my grandfathers, to my
aunt Mari, to Professor Yuming
Jiang and Jing Xie, to my friends
Eduardo and Gerard and
to all the people who put
their trust in me.**

ABSTRACT

Mobile ad-hoc networks (MANETs) are self-configuring networks of nodes connected via wireless without any form of centralized administration. This kind of networks is currently one of the most important research subjects, due to the huge variety of applications (emergency, military, etc...). In MANETs, each node acts both as host and as router, thus, it must be capable of forwarding packets to other nodes. Topologies of these networks change frequenly. To solve this problem, special routing protocols for MANETs are needed because traditional routing protocols for wired networks cannot work efficiently in MANETs.

The objective of this master thesis is to research the current state of the art of existing routing protocols for MANETs, and compare different approaches. There are three main classes of routing protocols for MANETs: reactive, proactive and hybrid. By studying advantages and disadvantages of each one, a new hybrid routing protocol is proposed. The new scheme called Penaguila, considers utilizing merits of both reactive and proactive protocols, and implements them as a hybrid approach. Penaguila allows that a mobile node flexibly runs either a proactive or a reactive routing protocol with its velocity and its traffic.

The new routing protocol is evaluated qualitatively. To verify the feasibility, a performance comparison with other typical existing routing protocols is discussed in this thesis also.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

ABR: Area Border Router

ACK: Acknowledgement

ANSN: Advertised Neighbour Sequence Number

AODV: Ad hoc On-demand Distance Vector

AR: Access Router

BGP: Border Gateway Protocol

BPS: Bits Per Second

BRP: Bordercast Resolution Protocol

CBR: Constant Bit Rate

CGSR: Clusterhead Gateway Switch Routing protocol

DCF: Distributed Coordination Function

DHCP: Dynamic Host Configuration Protocol

DSDV: Dynamic Destination-Sequenced Distance Vector

DSR: Dynamic Source Routing

DYMO: Dynamic MANET On-demand Routing Protocol

ELN: Explicit Loss Notifications

GPS: Global Positioning System

HNA: Host Network Association

HSLS: Hazy Sighted Link State

ICMP: Internet Control Message Protocol

IARP: Intrazone Routing Protocol

IERP: Interzone Routing Protocol

IP: Internet Protocol

LS: Link State

LSA: Link State Advertisement

OLSR: Optimized Link State Routing Protocol

OSPF: Open Shortest Path First

MANET: Mobile Ad-hoc Network

MN: Mobile Node

MPR: Multipoint Relay

P1: Proactive 1

P2: Proactive 2

PDA: Personal Digital Assistant

PSN: Packet Sequence Number

R1: Reactive 1

R2: Reactive 2

RERR: Route Error

RFC: Request For Comments

RREP: Route Reply

RREQ: Route Request

SACKs: (TCP) Selective Acknowledgment Options

TBRPF: Topology Dissemination Based on Reverse-Path Forwarding

TC: Topology Control

TCP: Transmission Control Protocol

TORA: Temporally-Ordered Routing Algorithm

TTL: Time To Live

UDP: User Datagram Protocol

WRP: Wireless Routing Protocol

ZRP: Zone Routing Protocol

1. INTRODUCTION

1.1. BACKGROUND

Mobile ad hoc networks (MANETs) are autonomous systems of mobile hosts connected by wireless links. This kind of networks is becoming more and more important because of the large number of applications, such as:

1. Personal networks: Laptops, PDA's (Personal Digital Assistants), communication equipments, etc.
2. Military applications: tanks, planes, soldiers, etc.
3. Civil applications: Transport service networks, sport arenas, boats, meeting centers, etc.
4. Emergency operations: searching and rescue equipment, police and firemans, etc.

To achieve efficient communication between nodes connected to the network new routing protocols are appearing. This is because the traditional routing protocols for wired networks do not take into account the limitations that appear in the MANETs environment.

1.2. PROBLEM DEFINITION

A lot of routing protocols for MANETs have been proposed in the last years. The IETF is investigating this subject and for example, protocols like AODV (Ad hoc On Demand Distance Vector) and OLSR (Optimize Link State Routing protocol) have been proposed as RFC's (Request For Comments). But, none of the existing protocols is suitable for all network applications and contexts.

The routing protocols for MANETs can be classified in three groups: reactive, proactive and hybrid.

The proactive protocols are based on the traditional distributed protocols shortest path based. With them, every node maintains in its routing table the route to all the destinations in the network. To achieve that, updating messages are transmitted periodically for all the nodes. As a consecuence of that, these protocols present a great bandwith consumption. Also, there is a great routing overhead. However, as an advantage, the route to any destination is always available. Thus, the delay is very small.

The reactive protocols determine a route only when necessary. The source node is the one in charge of the route discovery. As a main advantage, the routing overhead is small since the routes are determinated only on demand. As a main disadvantage the route discovery introduces a big delay.

The hybrid ones are adaptative, and combine proactive and reactive protocols.

Reactive protocols are advisable for networks with mobility, which are not sensitive to the delay. Proactive protocols are advisable to semistatic networks with small delay requeriment. There is no perfect routing protocol for all kinds of MANETs. Each routing protocol has its own stregths in some specific networking environments, but mobile nodes should be able to operate in every environment. A challenge is how to achieve that each node has the routing performance as high as possible when it crosses over different environments (e.g. from a low movility environmet to a high mobility environmet).

This master thesis proposes a routing protocol for MANETs with the objective that each node works using the most suitable features. To achieve that, every node checks periodically its speed and its traffic. Depending on these two parameters, the node will decide which features to use. The name of the protocol proposed here is Penaguila, and can be classified as a hybrid one.

1.3. LIMITATIONS IN THIS WORK

The major part of this work has been to find and study information on the current state of the art in MANETs, the routing protocols that are used (taking into acount the advantages and disadvantages of each one depending on the kind of MANET), and to design a new routing protocol using the acquired knowled. Since this thesis has only been carried out for over a single semester, it has not been possible to make any simulation or implementation of the Penaguila protocol. Hence, its evaluation has been done qualitatively and not quantitatively as it should have been.

1.4. CHAPTER OVERVIEW

This thesis looks into the current state of MANETs and the routing protocols for them. In addition it proposes a new protocol to be used in MANETs in which the nodes have different range of mobility and traffic.

Chapter 2 provides an overview of the state of the art regarding MANETs and their routing protocols.

Chapter 3 gives a detailed description of the new protocol proposed, Penaguila.

Chapter 4 presents an evaluation of Penaguila. Here, the new protocol is compared qualitatively with some of the most representative existing protocols.

Chapter 5 is a summary of the contributions and the conclusion of the thesis.

Chapter 6 provides and overviews the required future work to study the real performance of Penaguila.

Finally, in Chapter 7 there are a list of the references and resources that have been used in the research and to write this report. Unique words marked with [square brakets] throughout the text are references that can be looked up in this list for further information.

2. STATE OF THE ART

What are MANETs and why are they so interesting? How do the routing protocols operate in this kind of networks? This chapter gives an overview of these as well as a study to understand which routing protocol is better to use for each environment.

2.1. MOBILE AD-HOC NETWORKS: MANETS

2.1.1. DEFINITION AND ORIGIN

Mobile Ad-Hoc networks or MANET networks [manet_99] are mobile wireless networks, capable of autonomous operation. Such networks operate without a base station infrastructure. The nodes cooperate to provide connectivity. Also, a MANET operates without centralized administration and the nodes cooperate to provide services. Figure 2.1 illustrates an example of Mobile Ad-Hoc network.

Figure 2.1: Example of mobile Ad-Hoc network. In a MANET there is no form of centralized administration. All nodes can perform as hosts and as routers. Also the nodes are mobile. Hence, the topology changes constantly.

MANETs can communicate with different networks that are not ad-hoc. Therefore, they can communicate with wired networks creating hybrid networks. In the ad-hoc networks, the mobility of the nodes makes that the topology changes continuously. Hence, a specific dynamic routing protocol for MANETs which discovers and maintains the routes, and deletes the obsolete routes continuously is necessary.

The routing protocols for MANETs try to maintain the communication between a pair of nodes (source-destination) in spite of the position and velocity changes of the nodes. To achieve that, when those nodes are not directly connected, the communication is carried out by forwarding the packets, by using the intermediate nodes.

Currently there is research on the behaviour of a lot of those routing protocols and the IETF (Internet Engineering Task Force) is working on the standardisation of some of them. The protocols that are in experimental phase RFC (Request For Comments) include DYMO (Dynamic MANET On demand Routing Protocol) [DYMO_06], OLSR [OLSR_03], AODV [AODV_03], DSR (Dynamic Source Routing) [DSR_04] and TBRPF (Topology Dissemination Based on Reverse Path Forwarding) [TBRPF_04].

The origin of MANETs begins in the 70's for the military necessity of the interconnection of different hosts. This type of networks was implanted to avoid the need of a central base of communications. With these networks it was expected to transmit information in a fast and stable way as well as to cover the major part of the possible range without the necessity of having a previous infrastructure.

2.1.2. AD-HOC NETWORKS: CHARACTERISTICS AND PROBLEMS

The main characteristic of MANETs is that the hosts use wireless medium. In addition, they can move freely. Therefore, the network topology is changing constantly and they do not need any previous infrastructure to be used.

Another characteristic is that the hosts perform as routers.

There are some problems in ad-hoc networks as stated below.

2.1.2.1. PROTOCOLS ARCHITECTURE

The TCP (Transmission Control Protocol) is a connection oriented protocol, and it is designed for wired networks. In these networks the data loss rate is very small, hence, the reliability is high. When a packet loss is detected in the wired networks it is to a large extent because of the network congestion and TCP reduces the data emission rate. On the other hand, in wireless networks the main problem is not the congestion, but the data loss is because in these networks there is a greater data error rate.

This is why TCP reduces the sending rate when it is actually not necessary, making worst the performance of the MANET.

For a better operation of this protocol there are improvements, for example New Reno [rfc3782], SACKs (Selective Acknowledgment Options) [rfc2018], ELN (Explicit Loss Notifications) [rfc3135] that can be used in the wireless networks to improve the performance of TCP.

The TCP/IP architecture is chosen for the compatibility with the Internet, but this architecture is not the best for MANETs. It has been demonstrated that there are other architectures better to this kind of networks.

2.1.2.2. ADDRESSING

Currently, there is no mechanism to realize the auto configuration, as for example in the DHCP (Dynamic Host Configuration Protocol) existing in the fixed or infrastructure networks.

2.1.2.3. TOPOLOGY AND ROUTING

The nodes mobility makes the topology change continuously and therefore the nodes create and delete links dynamically.

The routing is not the same as in the wired networks. In wired networks routers are the central elements. In MANETs, there is no such element, but all the nodes can perform as a router, transmitter or receiver element. Hence, the routing is made by the node executing a specific routing protocol for MANETs.

2.1.3. USAGE

Each time there is a greater tendency to use wireless devices. Thus, there are a lot of applications for these networks. Hereafter are some of the most important.

- *Military applications.* The origin of these networks was from the military application. There are a lot of applications in the battle fields difficult to access where there is no previous infrastructure. These networks can be made between tanks, planes, and other mobile elements.

- *Difficult access networks.* These applications are realized in places where it is not possible or not economic to install a wired network, because of the ground topology. In this case, it is more convenient to use an Ad-Hoc network.

- *Emergency service.* These applications are necessary in natural disaster cases (hurricanes, floodings, etc.), since it is not possible to have at one’s disposal a wired network or a previous infrastructure.

- *Mesh networks.* The Mesh networks are Ad-Hoc networks where different nodes are connected by a point to point topology, and intermediate nodes are used to reach the destination if this is not in the coverage area. The main application is the communication between big cover areas by means of hops.

2.2 ROUTING PROTOCOLS FOR MOBILE AD-HOC NETWORKS

As it has been said, MANETs are necessary to have different routing protocols from the wired networks.

There are three types of routing protocols for MANETS:

- *Table-driven (Proactive):* OLSR, TBRPF, DSDV (Dynamic Destination Sequenced Distance Vector), CGSR (Clusterhead Gateway Switch Routing protocol), WRP (Wireless Routing Protocol), OSPF (Open Shortest Path First) MANET, etc.
- *Demand-driven (Reactive):* AODV, DSR, TORA (Temporally Ordered Routing Algorithm), etc.
- *Hybrids:* ZRP (Zone Routing Protocol), HSLS (Hazy Sighted Link State), etc.

In the proactive protocols, each node has a routing table, updated periodically, even when the nodes don't need to forward any message.

In the reactive protocols, the routes are calculated only when required. When a source wants to send information to some destination, it calls on route discover mechanisms to find the best route to this destination.

The hybrids protocols try to use a combination of both to improve them.

2.2.1 REACTIVE ROUTING PROTOCOLS

These protocols find the route on demand by flooding the network with Route Request packets. The main characteristics of these protocols are:

- Path-finding process only on demand.
- Information exchange only when required.
- For route establishment, the network is flooded with requests and replies.

In this section the DSR and AODV protocols are studied as a representative example.

2.2.1.1 THE DYNAMIC SOURCE ROUTING (DSR)

DSR [DSR_F02] is a reactive routing protocol. It uses source routing. The source node must determine the path of the packet. The path is attached in the packet header and it allows to update the information stored in the nodes from the path. There are no periodical updates. Hence, when a node needs a path to another one, it determines the route with its stored information and with a discovery route protocol. This protocol has 2 parts: the discovery and the maintenance of the routes.

Basic Route Discovery

When a node sends a packet to a destination, firstly it looks at its Route Cache the routes previously learned. If no route is found in its cache, then the node begins the route discovery process with a Route Request Packet (RREQ) broadcast. This packet includes the destination address, the source address and an identification number (request id). Each node receiving the RREQ, looks for the destination in its cache. If it does not know the route to the destination, it adds its address to the 'route record' in the RREQ and propagates it by transmitting it as a local broadcast packet (with the same request id). To limit the number of RREQ's, if one node receiving the RREQ has recently seen another RREQ from the same source, with the same request id, or if it finds its own address in the route record, then it discards the RREQ. In Figure 2.2 the development of the route record while the RREQ is spreading through the network is shown.

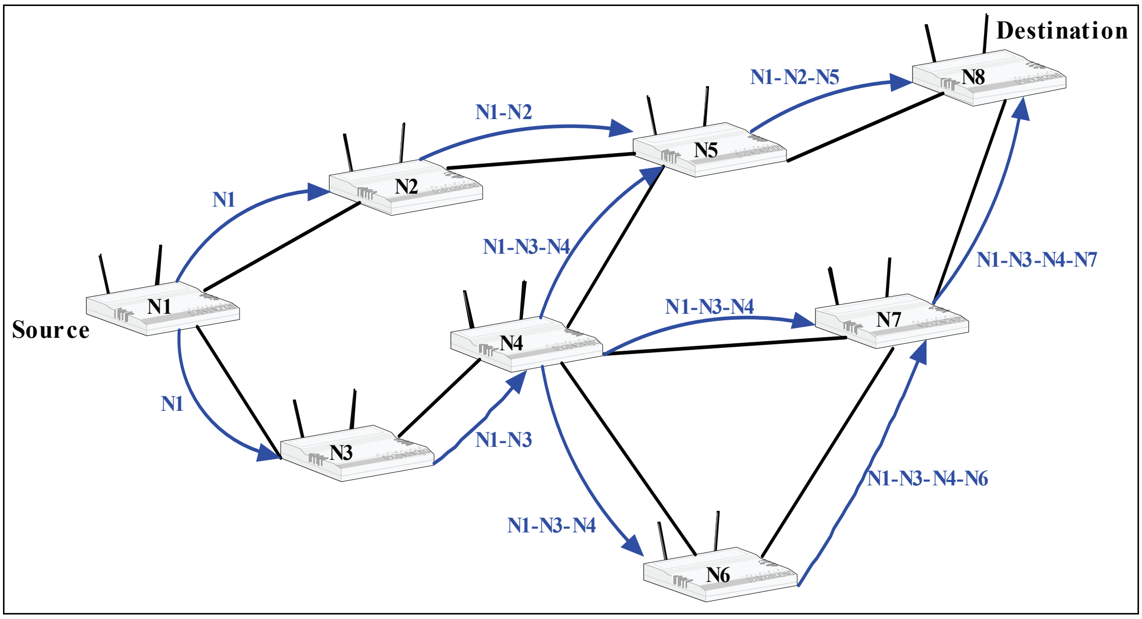


Figure 2.2: Construction of the route record in the route discovery. Each node adds its address to the route record field in the RREQ message. Nx-Ny-… indicates the addresses attached in the RREQ.

A RREP (Route Reply) is sent when the RREQ reaches the destination or an intermediate node that has the route to the destination. When the RREQ reaches the destination, it has the route record with the sequence of nodes crossed. If the node that generates the RREP is the destination, then it copies the route record sent in the RREQ. If the node that generates the RREP is an intermediate node, then it adds to the route record sent the route to the destination stored by it. If the links are bidirectional the RREP is sent by the reverse path. If the links are not symmetric, the node that sends the RREP must update its previous stored entry to the source (or to begin a route discovery to the source). In Figure 2.3, it is shown the RREP broadcast to the source.

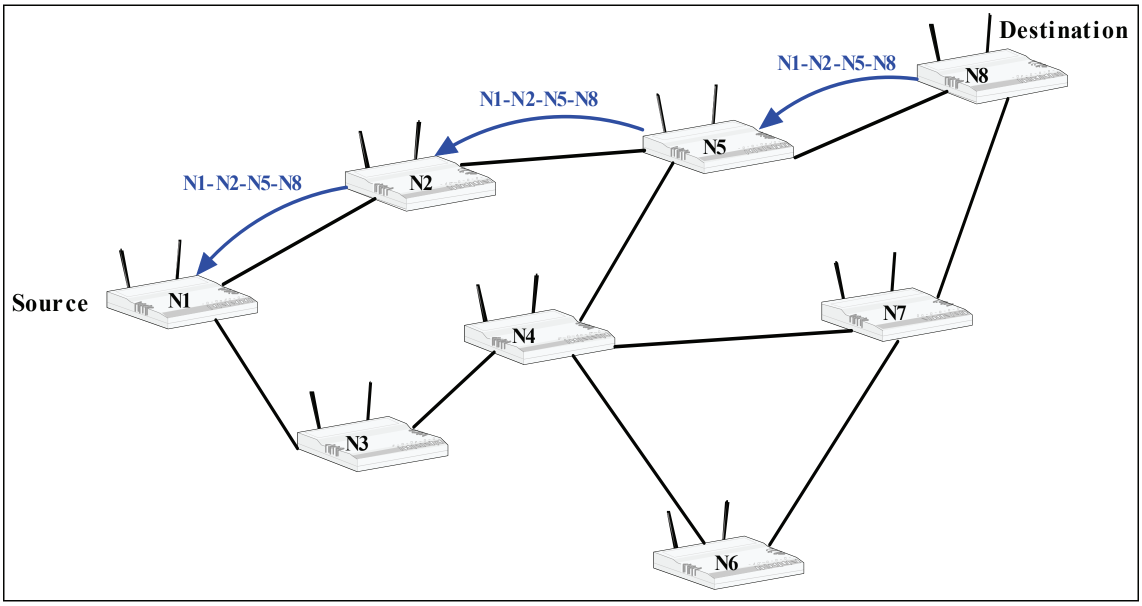


Figure 2.3: Forwarding of the RREP with the route record.

Basic Route Maintenance

The maintenance of the routes is useful to check the operation of a route and to report any routing error to the source. This check is made between consecutive nodes. When there is a problem in the transmission found by the link level, the RERR (Route Error) packets are sent by the node. This RERR has the addresses of both nodes in which the link failed. For example, in the situation illustrated in Figure 2.4 N1 has originated a packet for N8 using a source route through intermediate nodes N2 and N5. In this case, N1 is responsible for the reception of the packet at N2, N2 is responsible for the reception at N5, and N5 is responsible for the reception at the final destination N8.

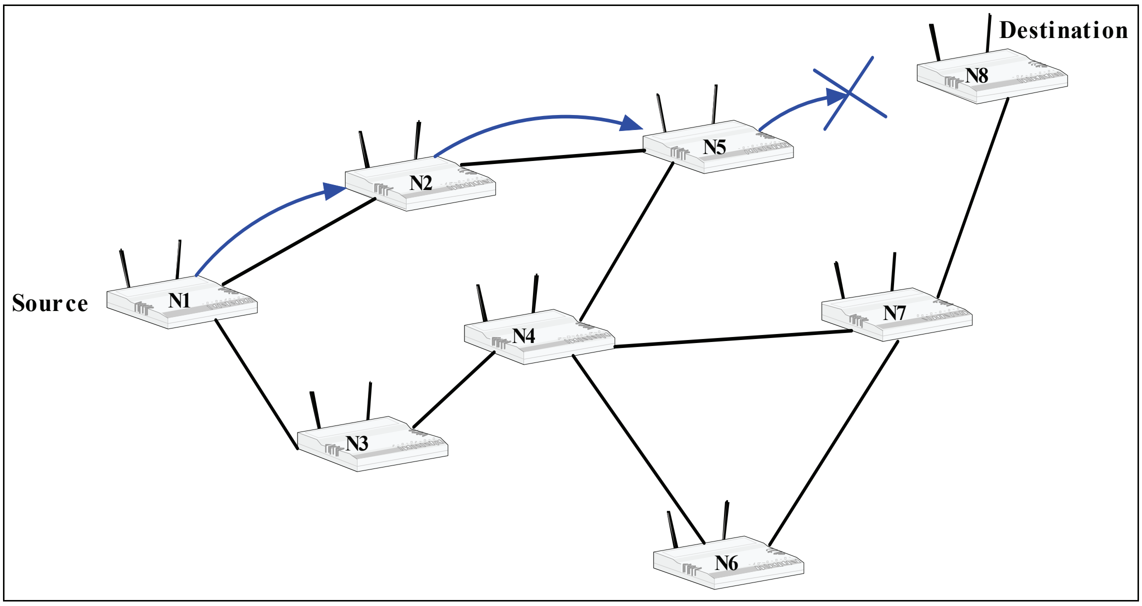


Figure 2.4: Route Maintenance example: N5 is unable to forward a packet from N1 to N8 over its link to next hop N8

As N5 is unable to deliver the packet to N8, N5 returns a Route Error to N1 stating that the link from N5 to N8 is currently 'broken'. N1 then removes this broken link from its cache. In other words, when a node receives a RERR, it deletes the link failed in its routes list, and all the routes that have this link are cut at this point. Besides the RERRs, ACKs (acknowledgements) can be used to verify the links availabity.

Advantages

- Its first advantage is the small overload in terms of packets to obtain routes, since DSR only manages the routes between nodes who want to communicate. Besides, DSR uses caching, and that can reduce the load of future route discovery.
- Another advantage is that only one RREQ process can produce some routes to the destination, thanks to the responses of the caches of intermediate nodes. If we compare the following protocols: DSDV, OLSR, AODV and DSR, the last one is the only who has numerous paths.
- Besides, there are no periodical updates.

Disadvantages

- However, DSR has disadvantages too. Using DSR, when a source sends a packet to any destination, the route is within the header. It is obvious that we are introducing byte overhead if the number of nodes is big in the network.
- Another disadvantage is the flooding. It can reach all the nodes in the network, when it is unnecessary. Besides, we have to prevent the collisions produced by the RREQ broadcasts (we can introduce random delays before sending the RREQ).
- The cache using also creates a problem: An intermediate node can corrupt the other nodes cache sending RREP using an obsolete cache. Therefore, we do not know how often the caches must be updated. If we update the cache very often, we produce overload on the network. But if we rarely update the cache, if the nodes move fast, we will have a wrong route. In [YUMING_04], the authors propose to use a quantity [Tp, L(Tp)] to estimate link status in the future, concluding that Tp*L(Tp) can be an optimal time to update the cache. However, the basics problems of DSR must still be solved.
- Broken links can not be repaired locally.
- It performs badly at high mobility because of the caching.

Resilience

From the point of view of the resilience to topology changes, DSR is a very good protocol for MANETS. In the example of Figure 2.4, when the link fails, a retransmission of the original packet can be sent to the same destination if N1 has in its Route Cache another route to N8 (for example, from additional RREP from its earlier Route Discovery, or from having overheard sufficient routing information from other packets). Otherwise, it may perform a new Route Discovery.

2.2.1.2 THE AD-HOC ON DEMAND DISTANCE VECTOR (AODV)

The AODV protocol [AODV_N02] is a reactive routing protocol. It is a Single Scope protocol and it is based on DSDV [DSDV_A94]. The improvement consists of minimizing the number of broadcasts required to create routes. Since it is an on demand routing protocol, the nodes who are not in the selected path need not maintain the route neither participate in the exchange of tables.

When a node wants to transmit to a destination and it does not have the valid route, it must begin the Path Discovery process. Firstly, it sends a broadcast of the Route Request (RREQ) packet to its neighbours, and they relay the packet to their neighbours and so on until they reach the destination or any intermediate node which has a 'fresh' route to the destination (Figure 2.5). Just like in DSDV [DSDV_94] sequence numbers are used to identify the most recent routes and to solve the loops.

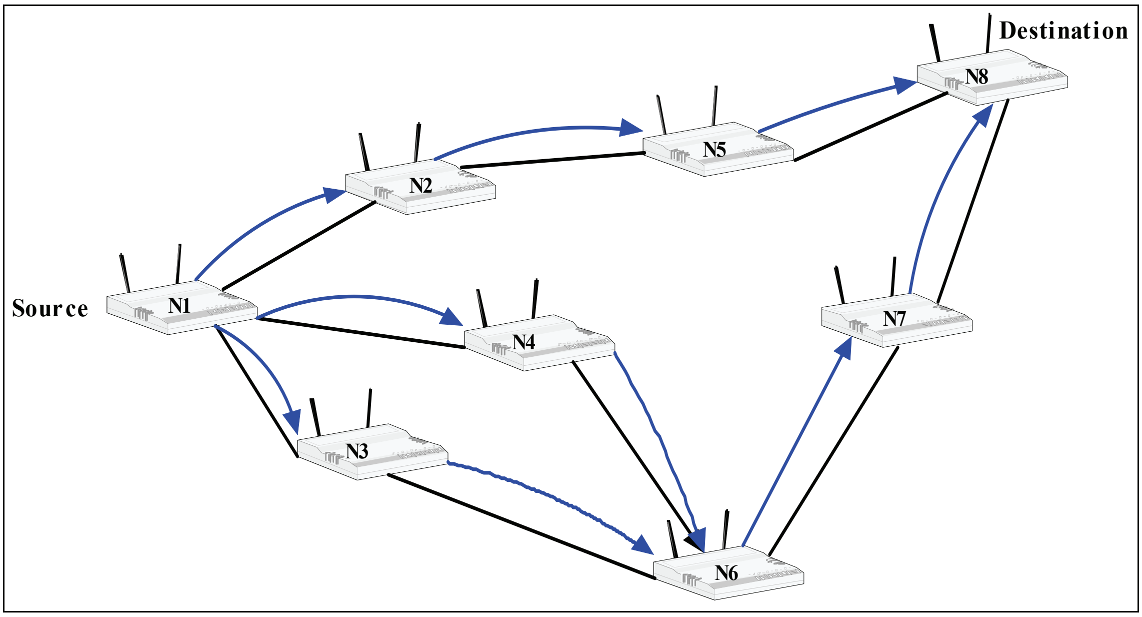


Figure 2.5: Propagation of the RREQ

Each node maintains two counters: the sequence number of the node (to solve the loops) and the broadcast ID which is incremented when a broadcast is started in the node. To identify only one RREQ (see Figure 2.7) it is used the broadcast ID and the IP (Internet Protocol) address of the source node. The RREQ has the following fields: Source address, Source sequence number, Broadcast_id, Destination address, Destination sequence number, and the number of hops to the destination.

The intermediate nodes only answer to the RREQ if they have a path to the destination with a sequence number greater or equal to the sequence number of the RREQ. Hence, only if they have paths equal (in age) or more recent. While the RREQ is sent, the intermediate nodes increase the field 'number of hops to the destination' and, also store in its routing table the address of the neighbour from whom they first received the message, in order to establish a 'Reverse Path' (Figure 2.6). The copies of the same RREQ received later which are coming from the other neighbours are deleted.

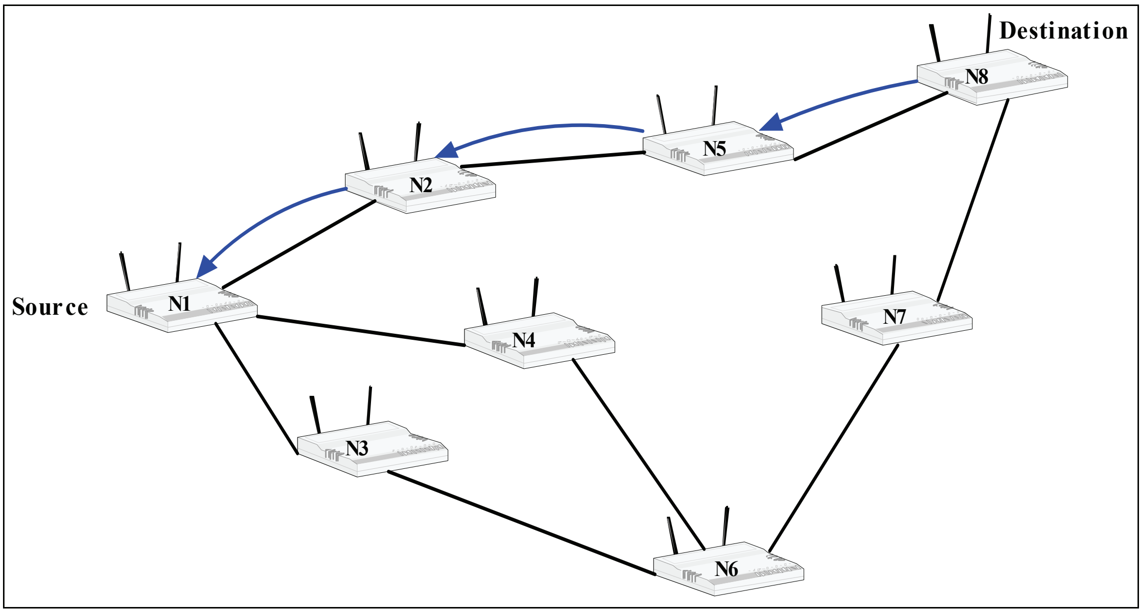


Figure 2.6: Path of the RREP to the Source

When the 'destination node/intermediate node with the fresh route' has been found, it answers with a Route Reply (RREP) to the neighbour from which it received the first RREQ. The RREP has the following fields: Source address, Destination address, Number of Hops to the destination, Sequence number of the destination, Expiration time for the Reverse Path (Figure 2.8). Then, the RREP uses the return path established to the source node. In its path, every node forwarding the RREP sets the reverse path as the freshest path to the destination node. Therefore, AODV can only use bidirectional links.

If a source node moves, it is capable of restarting the discovery protocol to find a new path to the destination. If an intermediate node moves, its previous neighbour (in source-destination way) must forward a RREP not requested with a fresh sequence number (greater than the known sequence number) and with a number of hops to destination infinite to the source node. In this way, the source node restarts the path discovery process if it is still needed.

Hello messages (periodic broadcasts) are used to inform mobile node about all the neighbourhood nodes. These are a special type of RREP not solicited, of which sequence number is equal to the sequence number of the last RREP sent and which has a TTL=1 (Time To Life) to not flood the network. They can be used to maintain the network connectivity, although other methods used more often exist for this function, like for example, to listen to the neighbour nodes transmissions.

Src_addr	Src_Seq_Nr	Bcast_ID	Dest_addr	Dest_Seq_Nr	Hop_cnt

Figure 2.7: RREQ packet

Src_addr	Dest_addr	Hop_cnt	Dest_Seq_Nr	Exp_Time_RP

Figure 2.8: RREP packet

Advantages

- AODV has low control signalization because there are not periodic updates about the routing and the overload in terms of packets is small since it is a reactive protocol. Also, the processing signalization is low because the AODV messages are simple and require small calculus. Besides, the loops are solved.
- AODV is a simple protocol that aims to resolve more recent and shorter paths. DSR, on the other hand, employs multiple optimizations, which in some cases result into worse performance [HIGHDYN_06] e.g. invalid route pollution due to aggressive route learning and caching.

Disadvantages

- AODV works only with bidirectional links. Although AODV only manages the routes between nodes who want to communicate, it uses Hellos messages periodically. Thus, in comparison with DSR the overhead in terms of packets is higher.
- Inconsistent route may appear.
- Multiple RREP can lead to heavy control overhead.
- Periodic beaconing.

Resilience

AODV has a high resilience to mobility and it is very good to be used in highly dynamic environments (over 20 m/s). In the study realized in [HIGHDYN_06] at very high speeds, despite optimizations that address limitations of DSR's aggressive route caching mechanism, DSR is found inferior to AODV. At these speeds AODV exhibits impressive resilience.

2.2.2. PROACTIVE ROUTING PROTOCOLS

These algorithms maintain a fresh list of destinations and their routes by distributing routing tables in the network periodically. The main characteristics are:

- These protocols are extensions of wired network routing protocols.
- Every node keeps one or more tables.
- Every node maintains the network topology information.
- Tables need to be updated frequently.

In this section the OLSR and DSDV protocols as a representative example are studied.

2.2.2.1. OPTIMIZED LINK STATE ROUTING (OLSR)

OLSR [OLSR_01] is a proactive link state routing protocol. It is a point to point routing protocol based in the link state algorithm [LSR_95].

Each node maintains a route to the rest of the nodes of the ad hoc network. The nodes of the ad hoc network periodically exchange messages about the link state, but it uses the 'multipoint replaying' [OLSR_N98] strategy to minimize the messages quantity and the number of nodes that send in broadcast mode the routing messages. The strategy MPR (Multipoint Relay) [MANET_04] lies in that each node uses 'Hello' messages to discover what nodes are in a one hop distance and makes a list. Each node selects a group of neighbours of that list that are able to reach all the nodes in a distance of two hops with regard to the node that is making the selection. For example, in Figure 2.9 the node A selects the nodes B, C, K and N as the MPR nodes, because they are capable of reaching all the nodes at two hops distance with regard to the node A.

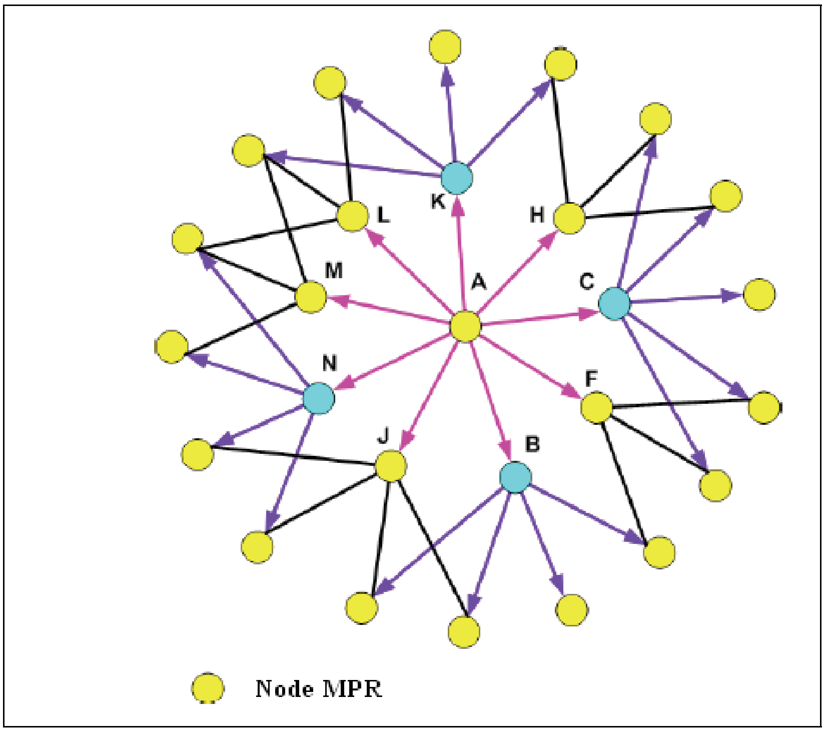


Figure 2.9: Multipoint Relays. Node A selects nodes B, C, K and N as MPR nodes because through them it can reach all the nodes to 2-hop distance.

These neighbours selected are the only nodes in charge to relay the routing packets and are called MPRs (Multipoint Relays). The rest of the neighbourhood process the routing packets that they receive, but they can not relay them.

Each node decides an optimum path (in number of hops) to each destination using the stored information (in its topology routing table and in of their neighbours ones) [Abolhasan_04]. Besides each node stores that information in a routing table for usage when a node wants to sent data.

This protocol selects bidirectional links to send packets [MANET_03], and does not use unidirectional links.

The OLSR protocol is more efficient in networks with high density and highly sporadic traffic. The quality metrics are easy to expand to the current protocol. OLSR requires that it continuously has some bandwidth in order to receive the topology updates messages.

Advantages

- The proactive characteristic of the protocol provides that the protocol has all the routing information to all participating hosts in the network. OLSR protocol needs that each host periodically sends the updated topology information throughout the entire network. This increases the protocol bandwidth usage. However, the use of MPRs minimises the flooding in comparison with other proactive routing protocols.
- OLSR protocol is well suited for the application which does not allow the long delays in the transmission of the data packets. The best working environment for OLSR protocol is a dense network, where the majority of the communication is concentrated between a large number of nodes. [OLSR_01]
- The reactiveness to the topological changes can be adjusted by changing the time interval for broadcasting the Hello messages. It increases the protocols suitability for ad hoc network with the rapid changes of the source and destinations pairs. Also the OLSR protocol does not require that the link is

reliable for the control messages, since the messages are sent periodically and the delivery does not have to be sequential. [OLSR_QoS]

- OLSR has also extensions to allow hosts to have multiple OLSR interface addresses and provide the external routing information giving the possibility for routing to the external addresses. Based on this information there is the possibility to have hosts in the ad hoc network which can act as gateways to another possible network.

Disadvantages

- As proactive routing protocol, a great number of periodical messages are sent. Besides the HELLO messages, there are Topology Control messages, forwarded around all the nodes in the network. The use of MPRs solves in part that problem, but the overhead in terms of packets is still high in comparison with the reactive routing protocols.

2.2.2.2. THE DESTINATION SEQUENCED DISTANCE VECTOR (DSDV)

The DSDV is a distance vector, proactive routing protocol. It is based in the Bellman-Ford algorithm [DYPRO_57] [FLONET_62], but improved to solve the routing loop problem. It uses the distance vector algorithm to find the shortest path to the destination [MANET_03].

Each node within the ad hoc network maintains a routing table with the following information to each destination [DSDV_94].

- Destination IP address.
- Destination sequence number.
- Next hop (IP address).
- Cost (in number of hops).
- Install time: used to delete old routes.

Each node sends periodically broadcasts with the routing table updated to its neighbours [ADHOC_01]:

- Each node adds its sequence number when it sends its routing table.
- When the other nodes receive this information, they update its routing tables.

The routing tables also can be sent if there are topology changes (link creation or breakage). In this case, the update information travelling in the routing messages is:

- Destination IP address.
- Number of hops.
- Sequence number.

The nodes use the sequence numbers to distinguish between old and new routes to a destination. A node increases its sequence number when there is a topology change (a new link is created or deleted). The route to a destination with the biggest sequence number (the more current) is the valid one. If there are two routes with the same sequence number, the valid is the one which number of hops is smaller. Two types of route update are used [Abolhasan_04]:

- Full dump

  This packet carries the whole routing table. It is unusual to send this packet.
- Incremental

  This packet carries only the routing table information of a node that has changed since the last full dump sent. These packets are sent more frequently. Hence, the control overhead and the bandwidth consumption are smaller.

However, DSDV still has control overhead, growing as $O(N^2)$, where N is the number of nodes in the network. For this reason, the protocol is not scalable.

To achieve that a node does not announce a route change when there is a better route in a discovery process, it is necessary that each node waits for a fixed time before announcing a new route with a smaller cost. That fixed time is calculated as the average time necessary to achieve all the update messages of a route. Hence, the neighbours reduce the bandwidth use and the power consumption.

In the next example (Figure 2.10) there is an ad hoc network using DSDV. The node M_4 has a routing table as the shown in Figure 2.11.

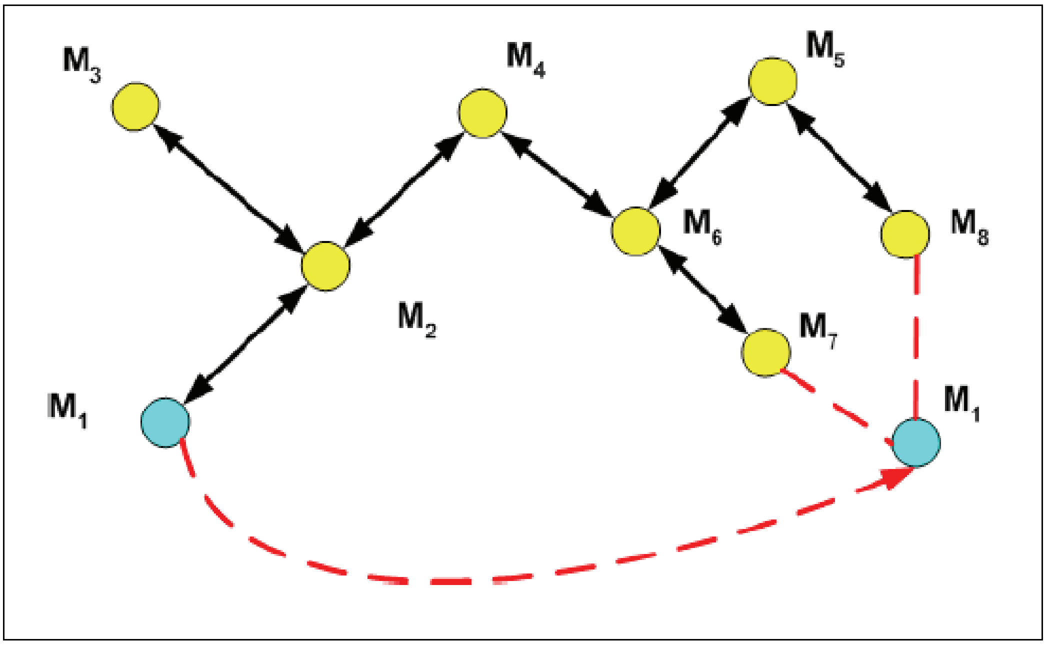
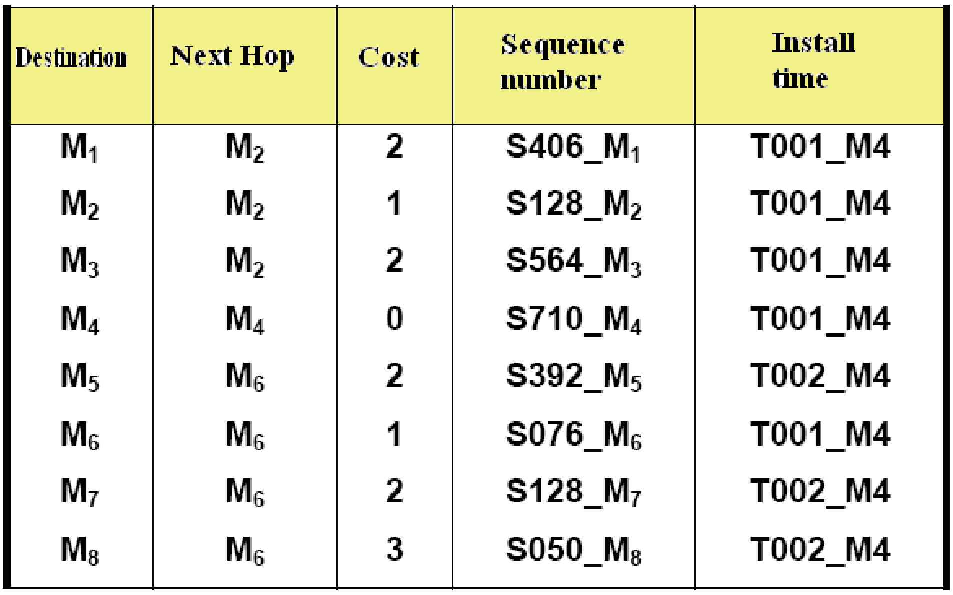


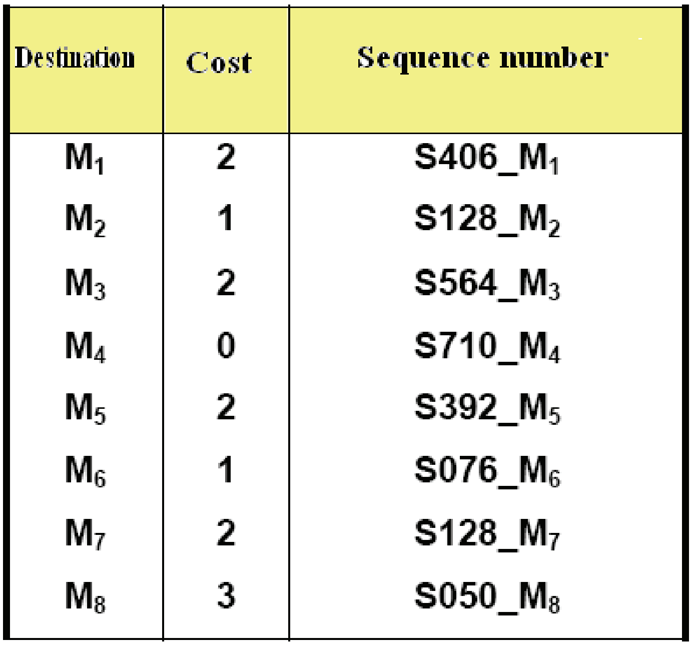
Figure 2.10: Ad hoc network with mobility



Destination	Next Hop	Cost	Sequence number	Install time
M_1	M_2	2	S406_M_1	T001_M4
M_2	M_2	1	S128_M_2	T001_M4
M_3	M_2	2	S564_M_3	T001_M4
M_4	M_4	0	S710_M_4	T001_M4
M_5	M_6	2	S392_M_5	T002_M4
M_6	M_6	1	S076_M_6	T001_M4
M_7	M_6	2	S128_M_7	T002_M4
M_8	M_6	3	S050_M_8	T002_M4

Figure 2.11: Node M_4 routing table

The routing table sent in the update routing message is shown in Figure 2.12.

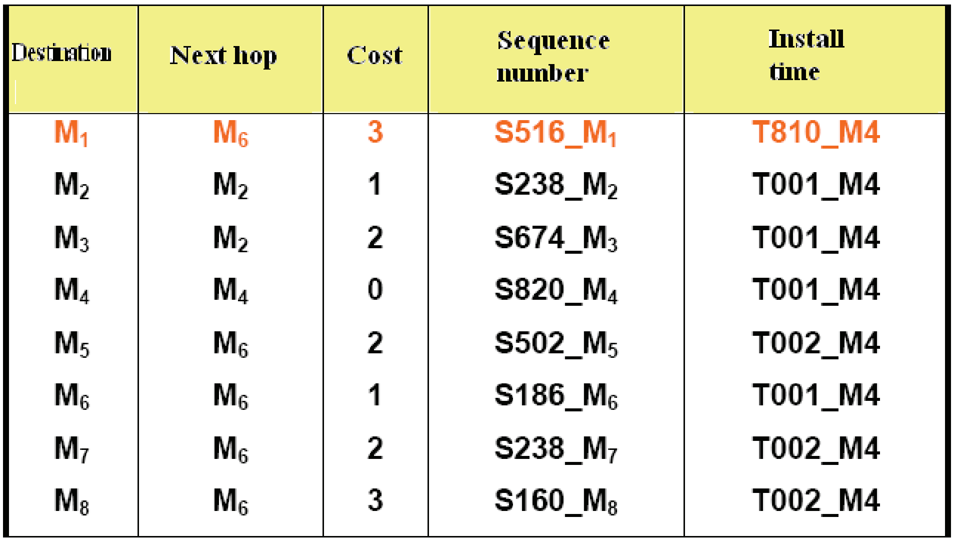


Destination	Cost	Sequence number
M_1	2	S406_M_1
M_2	1	S128_M_2
M_3	2	S564_M_3
M_4	0	S710_M_4
M_5	2	S392_M_5
M_6	1	S076_M_6
M_7	2	S128_M_7
M_8	3	S050_M_8

Figure 2.12: Update routing table of the node M_4

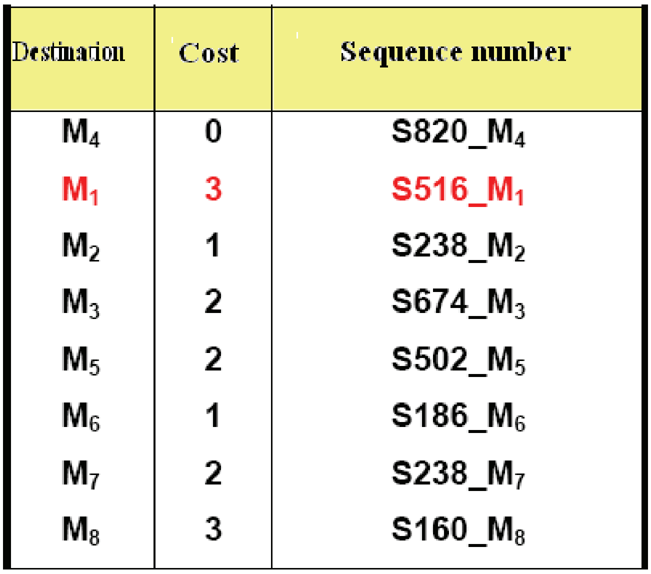
If there is a topology change in the network and if the node M_1 changes to an other place as shown in Figure 2.10, the routing table of the node M_4 and the update routing message are the shown ones in Figure 2.13 and in Figure 2.14 respectively.

There only is a new register in the routing table for the node destination M_1, but during this time it has received new sequence numbers of destinations associated to other registers in the table. The node M_4 must send an incremental routing message to inform its neighbours about the change in the register of the destination M_1 so they can know about that change without waiting for the next 'full dump' packet with the update of the routing table. It also includes the variations of the sequence numbers of the rest of registers in the routing table. Since every register in the table has changed, the result is like changing the whole routing table.



Destination	Next hop	Cost	Sequence number	Install time
M_1	M_6	3	S516_M_1	T810_M4
M_2	M_2	1	S238_M_2	T001_M4
M_3	M_2	2	S674_M_3	T001_M4
M_4	M_4	0	S820_M_4	T001_M4
M_5	M_6	2	S502_M_5	T002_M4
M_6	M_6	1	S186_M_6	T001_M4
M_7	M_6	2	S238_M_7	T002_M4
M_8	M_6	3	S160_M_8	T002_M4

Figure 2.13: Routing table of the node M_4 (updated)



Destination	Cost	Sequence number
M_4	0	S820_M_4
M_1	3	S516_M_1
M_2	1	S238_M_2
M_3	2	S674_M_3
M_5	2	S502_M_5
M_6	1	S186_M_6
M_7	2	S238_M_7
M_8	3	S160_M_8

Figure 2.14: Routing Table of update of the node M_4 sent in the incremental routing message

Advantages

- DSDV does not bloat packets. Source routing algorithms, on the other hand, put the whole route in packets, adding to their size, increasing the chance of collisions, and reducing throughput.
- Routes to all destinations are always available.
- Less delay for route setup.

Disadvantages

- DSDV discovers routes even if they are not needed.
- Heavy control overhead because of updates.
- Updates can choke the whole bandwidth.
- Not scalable.
- Very bad for large networks or high mobility.

2.2.3 HYBRID ROUTING PROTOCOLS

These protocols are a combination of reactive and proactive routing protocols, trying to solve the limitations of each one.

Hybrid routing protocols have the potential to provide higher scalability than pure reactive or proactive protocols. This is because they attempt to minimise the number of rebroadcasting nodes by defining a structure (or some sort of a backbone), which allows the nodes to work together in order to organise how routing is to be performed. By working together the best or the most suitable nodes can be used to perform route discovery.

2.2.3.1. THE ZONE ROUTING PROTOCOL (ZRP)

The Zone Routing Protocol (ZRP) [ZRP_02] is a hybrid routing protocol. It combines the advantages from reactive and proactive routing protocols. This protocol divides its network in different zones. These zones are the nodes local neighbourhood. Each node has its own zone. Each node can be into multiple overlapping zones, and each zone can be of a different size. The size of a zone is given by a radius of length [MANET_04], where the number of hops is the perimeter of the zone. Within each zone it is used a proactive routing protocol. Therefore, each node into the zone knows how to reach its neighbours. However, if the packets are sent to a node outside of the zone, it is used a reactive routing protocol.

ZRP [OPTZRP_99] runs three routing protocols:

- Intrazone Routing Protocol (IARP)
- Interzone Routing Protocol (IERP)
- Bordercast Resolution Protocol (BRP)

IARP is a link state routing protocol. It operates within a zone and learns the routes proactively. Hence, each node has a routing table to reach the nodes within its zone.

IERP uses the border nodes to find a route to a destination node outside of the zone. IERP uses the BRP.

BRP is responsible for the forwarding of a route request.

When the Route Discovery process begins, the source node asks to its routing table and if necessary, it starts a route search between different zones to reach a destination. If a route is broken by a node's mobility into the same zone where the node was, the routing tables used for the proactive routing protocol must be updated. If the node's mobility is from one zone to another one, then it is necessary to execute a query between zones.

To use a reactive routing protocol to find a route from a source node to a destination node placed in another zone reduces the control overhead (in comparison with the proactive ones) and the delays in the Route Discovery (in comparison with the pure reactive ones), since these routes are discovered much faster. The reason is because to find a route to a node placed outside the routing zone, the route request is send only to the border router within the zone where the destination is. This border router can answer to the request since it has a routing table to do the proactive routing and knows how to reach the destination.

The disadvantage of ZRP [Abolhasan_04] is that it becomes a proactive routing protocol if the radius is big. Otherwise, if the radius is small, it becomes a reactive routing protocol.

In Figure 2.15 a Route Discovery process is shown; the node S sends information to the node X, and by IARP decides X is not in the same zone that S. The search travels through the border nodes to find the zone where X is. Finally, the border node G discovers that X is in its zone and sends a route response to S.

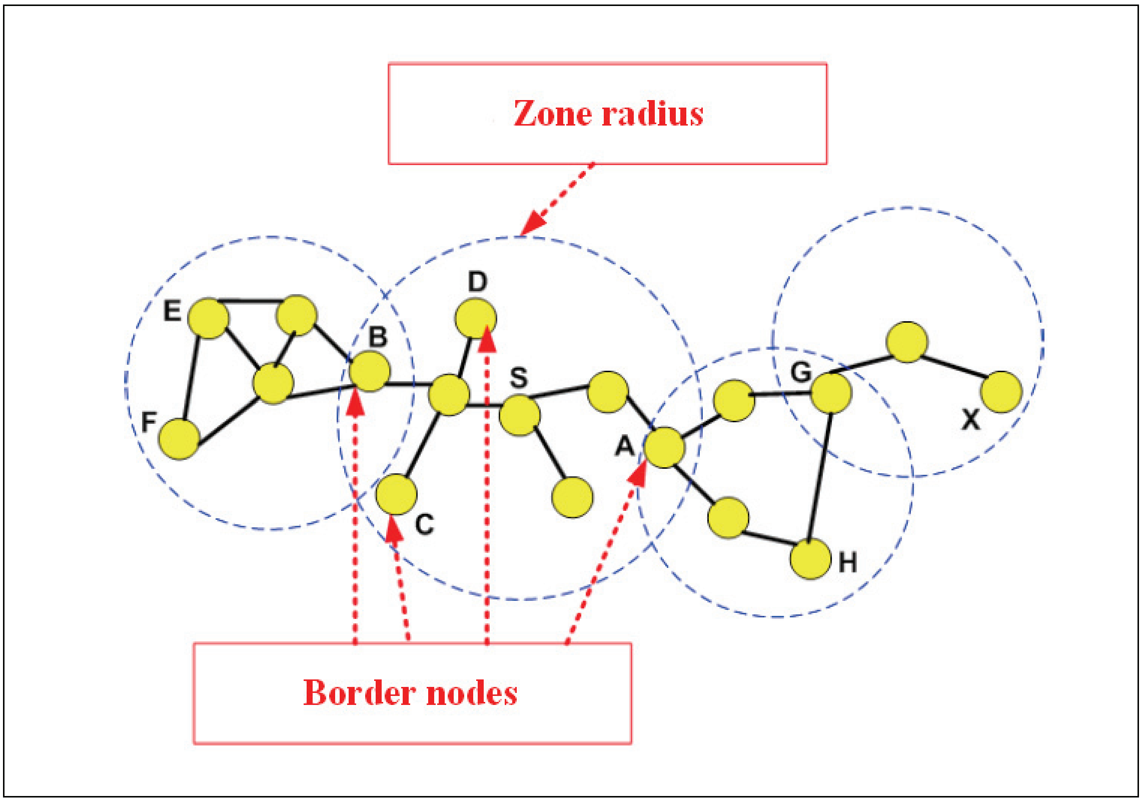


Figure 2.15: Example of a Route Discovery in an ad hoc network using the routing protocol ZRP. For node S, the zone radius is 2, and the border nodes are A, B, C and D.

Even though the hybrid nature of the ZRP seems to indicate that it is a hierarchical protocol, it is important to point out that the ZRP is in fact a flat protocol. ZRP is more efficient for large networks.

Advantages

- ZRP is more suitable than other protocols for large networks spanning diverse mobility patterns by providing the benefits of both reactive and pro-active routing in a flat network that takes advantage of a near-hierarchical approach.

Disadvantages

- If zones greatly overlap, redundant Route Request messages flood the network.
- Optimum zone radius must be determined for each situation
- High stress for intermediate nodes on link failure

2.2.4 REACTIVE VS PROACTIVE

Proactive routing protocols loose more time updating their routing tables. Therefore when the topology changes frequently, most of the current routes in the tables can be wrong. Hence, these protocols are recommended for ad-hoc networks semi dynamics.

Reactive routing protocols have delay in route determination, because of the flooding mechanism. They are recommended for networks with nodes moving constantly.

Intuitively, we can think in the advantages and disadvantages of both looking the table 2.1:

Parameters	Proactive/Table-driven	Reactive/On Demand driven
Route availability	Always available	Available when required
Latency	Minimum	Long delays when there is not an available route
Route updating periodically	Yes	No
Movement	Advertises to other nodes to update the routing tables.	Only advertises if affect to the source node. Uses alternative routes.
Control traffic	Greater than On Demand driven	Increase if mobility of the active routers increase.
Energy consumption	Greater	Depends of the nodes mobility

Table 2.1. Comparison between proactive and reactive routing protocols

Proactive protocol:

Advantages

- A route can be selected immediately without delay.

Disadvantages

- Produce more control traffic.
- Takes a lot of bandwidth.
- Produce network congestion.

Reactive protocol:

Advantages

- Lower bandwidth is used for maintaining routing tables.
- More energy-efficient.
- Effective route maintenance.

Disadvantages

- Have higher latencies when it comes to route discovery.

Reactive protocols face scaling problems when the number of nodes is large and have many “active nodes”. But how big this problem is it depends on which protocol is used and in which scenario it is working.

In the Table-driven case, the problem is the time to update the routing tables. These protocols require nodes to exchange their routing tables periodically (or if changes in the topology happen each). Thus, each node has its routing table updated. However, this information exchange can cause message broadcast storm when the mobility is high.

2.2.5 FLAT VS. HIERARCHICAL

Both architectures have strengths and weaknesses. The flat architecture has the following advantages over the hierarchical:

- More reliability and survivability.
    - No single point of failure.
    - Alternative routes in the network.
- More “optimal routing”.
- Better coverage, i.e. reduced use of the wireless resources.
- Route diversity, i.e. better load balancing property.
- All nodes have one type of equipment.

The *no single point of failure* means that if one node goes down, the rest of the network will still function. In the hierarchical if one of the cluster heads goes down, that section of the network won’t be able to send or receive messages to other sections for the duration of the downtime of the cluster head.

The flat routing algorithm doesn't have a good scalability. When the network becomes larger the routing overhead will increase rapidly.

The hierarchical architecture has the following advantages over the flat:

- Easier mobility management procedures (just ask the cluster head).
- Better manageability.

The flat network architecture is shown in Figure 2.16 and the hierarchical network architecture is shown in Figure 2.17. These two figures show the differences between the architecture of the two approaches:

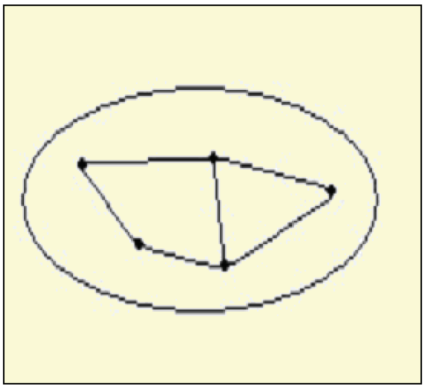


Figure 2.16: Flat network architecture. In a flat network all the nodes are in the same level. There are no "special" nodes.

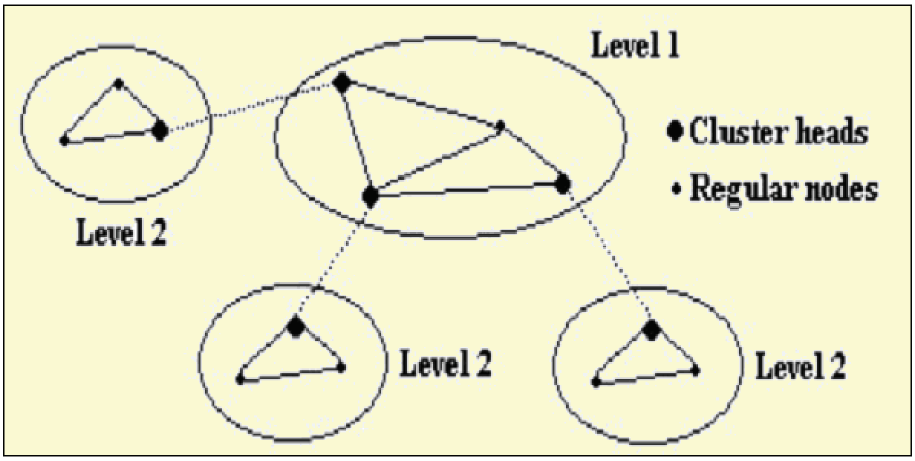


Figure 2.17: Hierarchical network architecture. In the hierarchical architecture there are two levels. The cluster heads centralize the communication between the regular nodes.

2.2.6 UNICAST VS. MULTICAST

In multicast routing a single packet is sent simultaneously to multiple receivers. In unicast routing a single packet is only sent to one recipient every transmission. Thus the multicast method is very efficient and a useful way to support group communication when bandwidth is limited and energy is constrained.

Due to the broadcast characteristics of the multicast protocol it is better suited for MANET then the unicast protocol [RPO].

2.2.7 UNIPATH VS. MULTIPATH

In a multipath routing protocol the packets can be sent via multiple paths between the source and destination. This increases the packet delivery ratio with regard to unipath. A. Nasipuri and S. R. Das in [ODM_99] prove this. This also means that there is no necessity of finding new routes, decreasing the route discovery traffic.

2.2.8 QUALITY OF SERVICE (QoS)

Quality of service can be used as a measurement of how good the routes in the network are. The routes should guarantee a set of pre specified service attributes, such as delivery, bandwidth and delay variance (jitter). It also involves the specification of latency, loss, availability etc...

For a protocol to provide good QoS it must determine new routes rapidly and with minimal bandwidth consumption. There are several metrics that directly affect the QoS of every protocol, for example: Packet delivery ratio, control packet overhead (packets and total bytes), average hop count, end-to-end latency and power consumption to mention a few. Using a protocol that provides good quality of service will greatly affect the MANETs performance.

2.3. DISCUSSION

This section discusses essential characteristics for designing an optimal MANET routing protocol, which should be scalable to support new nodes joining.

When deciding how to maintain routing information, the reactive approach is better when the mobility is high and the traffic is small. However, the proactive approach is advisable when the network is semi static and the traffic is high. If the routing protocol has to consider a network with each node at a different speed and with different traffic patterns, it is obvious that it is necessary that each node runs a different routing protocol. The combination of both classes of protocols, reactive and proactive, is implemented as a hybrid approach.

If the networks have to be large, it is better to use a hierarchical architecture, using different levels, and achieving that each node has only in its routing table the rest of the nodes working in its same level.

As said before, multicast is better suited for MANET since the bandwidth in these kind of networks is limited. AODV and ZRP have multicast capabilities but DSR and OLSR do not. However, there are extensions for multicast OLSR [MOLSR].

With regard to the other characteristics, a high QoS and a multicast approach is always desirable in a network but not always a priority.

2.4. CONCLUSION

For a network with a large number of nodes, which move with changing velocities and have different traffic patterns, a hybrid routing protocol is the best choice. The nodes moving slowly and with high traffic should run the proactive routing features, and the rest of nodes implement the reactive ones.

Besides, the choice should be a hierarchical approach to achieve a big scalability.

3. A NEW ROUTING PROTOCOL FOR MANETS: PENAGUILA

3.1. INTRODUCTION

The routing protocol for MANETs described in this thesis is called Penaguila. It is a hybrid and a hierarchical routing protocol. The nodes which move slowly or have high traffic will work (or will try to work) in proactive mode, joining a proactive area. The others will work in reactive mode. Since there are many typical routing protocols proposed, Penaguila uses two existing protocols directly. For proactive areas, OLSR is utilized because it is very popular and performs well compared with other proactive routing protocols. Reactive nodes run AODV for no additional overhead introduced with the network growing. Besides, when the mobility is very high, AODV has impresive resilience.

3.2. PROTOCOL DESCRIPTION

The description of Penaguila routing protocol is quite easy. Each node checks its velocity and its traffic periodically. If the velocity is smaller than a threshold X, or the traffic is higher than a threshold Z, then the node will try to join or to create a proactive area. Within this area, the features to use are the same that in the OLSR. If not, the node will work in reactive mode, using the same features that AODV. The proactive areas have a limited size in number of nodes. The number of nodes within an area can not be greater than a threshold Y. If a node that wants to join an area does not find an area with less than Y nodes, it has to create a new area or it can not work in proactive mode.

But not all the nodes inside the area work like pure OLSR. There are some nodes that have to work as gateways to communicate the area with the outside. Similarly, not all the nodes outside the area work in the same way that AODV. Some of them have special features to allow the communication between reactive and proactive nodes. How each node decides which features it has to use, as well as the description of this features is explained in the next chapters.

3.2.1. HOW A NODE DECIDES ITS FEATURES

3.2.1.1. PENAGUILA ROUTING PROTOCOL PARAMETERS

First of all, there are some parameters that have to be described to understand the operation of Penaguila.

V=velocity

Periodically, the node checks its velocity to know if topology changes can happen. The velocity to have into account to switch from an operation mode to another is the average velocity. That is, the node checks with GPS (Global Positioning System) its position periodically. The average velocity necessary to change from the last position to the current position is the V.

X= threshold velocity=3.5 m/s

If we review different performance studies as [Perf_MIL04], we can see that AODV is better than OLSR in all the range of mobility since the point of view of the throughput, the total amount of generated network traffic, and the resilience. However, when the nodes are semi-static (at very low velocities) the OLSR can perform better in terms of delay end-to-end. This is because in a network with not many topology changes OLSR can almost always give the shortest path available. As mentioned, AODV usually performs better than OLSR in every mobility environment, but at less than 3.5 m/s it can be interesting to use OLSR since the network is more similar to a static network than to a Mobile Ad-hoc network. When the network has no topology changes, the throughput and the resilience to topology changes are similar (there are not topology changes). Therefore, we can compromise the control traffic load to achieve a better delay end-to-end using the proactive routing protocol.

N=number of nodes in the area

N is the number of nodes working in the same area using the proactive features. Later, in the section 3.2.1.3, we will see that these nodes can work in Proactive 1 and Proactive 2 mode.

Y= threshold number of nodes in an area = 90

The proactive area works in the same way that OLSR. OLSR reduces the number of “superfluous” forwarding, reduces the size of LS updates, and reduces the table size. However, while the number of nodes into an OLSR area increases, the number of control packets increase. For the study made in [ScalOLSR], the OLSR should not exceed 400 nodes because it generates excessive control packets. Also, if the number of nodes increases, the local storage (Kbytes/node) increases. In the same study it is demonstrated that the packet delivery ratio decreases if the number of nodes is bigger than 100.

Therefore, a good threshold to the number of nodes in an OLSR network could be 90. OLSR allows choosing a big value for the number of nodes in a network, but when this value exceeds 100 the performance of the protocol may decrease. With the number of 90, there is a margin of 10 nodes to reach this critical point.

T= Traffic

T is the traffic that a node manages. This traffic is just data traffic (with no control traffic), and can be both the traffic generated by the node and the traffic routed by the node and generated in others nodes.

Z= threshold value of traffic= 300 kbps

As explained before, when the traffic in the network is high, the nodes need to know the route to the destination as fast as possible. In this case a proactive routing protocol outperforms the reactive one because it already has the route when necessary. However, it is quite difficult to define a threshold value for the traffic of a fixed node. That is because the traffic analysis as [TrafficP], [TrafficO], study the effect of the traffic injected to the network, not to one node. In this protocol we define a high value for Z, because AODV can perform well for a lot of values of the traffic injected, and to decide to change to OLSR the traffic must be quite high, for example 300 kbps.

3.2.1.2. A SINGLE NODE

If we have a node implementing the Penaguila routing protocol, this one must know its velocity (for example, using GPS) and its traffic. If the velocity V, is ≤ than a threshold velocity X or the traffic T is > than a threshold Z, then the node knows that it is better to use the proactive features since the nodes with low mobility and high traffic always perform better with a proactive protocol than with a reactive one. Hence, the node will try to join an area with other nodes in the same situation.

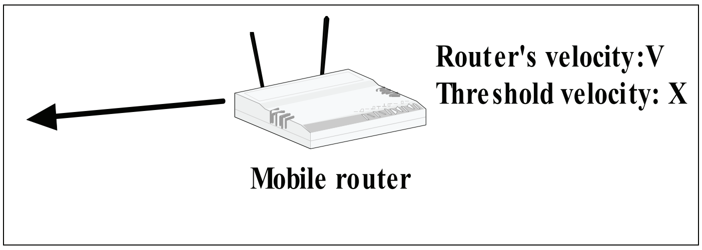


Figure 3.1: A mobile node knows its velocity by GPS

If none of both conditions mentioned before happens, then the node knows that it is better to use the reactive features. If the V is not very small, the topology is changing fast and is not efficient to change the routing information periodically all the time (even when these routes are not being used). Also, if the T is not very high it is not efficient to maintain routes constantly because these are not being used very often.

3.2.1.3 A NODE OPERATION

A node working with Penaguila protocol will work using different features depending on its velocity, traffic and environment. Penaguila defines 6 different states for a node: Initial, R1 (Reactive 1), R2 (Reactive 2), R3 (Reactive 3), P1 (Proactive 1), P2 (Proactive 2) and P3 (Proactive 3) states. Figure 3.2 illustrates a diagram state describing the behaviour of a node. Hereafter, each state is described:

- Initial state: When a node is reset it begins in an initial state. In this state the node must check its velocity and its traffic to decide in which mode it has to work. We define "condition 1" as: "(V<=X) OR (T>Z)". If condition 1 doesn't happen then it will work in the reactive mode (Reactive 1), but if condition 1 happens, then it will try to work in the proactive mode. Hence, the node will pass to the Reactive 3 state.

- Reactive 1: In this state, the node works using the AODV features. While condition 1 is not fulfilled and the node does not have connectivity with an area it will remain in the same mode of operation. In the case that the node discovers a node or more working in the Proactive 1 or Proactive 2 modes then it will work in the Reactive 2 mode. If condition 1 is fulfilled, then it will try to work in proactive mode (Reactive 3).

- Reactive 2: In this state, the node works using the AODV features, but also must process the control messages coming from the proactive zone. This is because it needs these messages to have, in its routing table, the proactive destinations. While there is no condition 1 and while the connectivity with any node working in the Proactive 1 or Proactive 2 modes continues the node will remain in the same state. If condition 1 is not fulfilled but the router looses the connectivity with the mentioned routers, then it will come back to the Reactive 1 state. If condition 1 occurs then it will try to work in proactive mode (Reactive 3 state).

- Reactive 3: This state exists for the reason that when a node decides that to work in proactive mode is better, firstly it must join or create an area. In this state the node still works using the AODV features, but also has to generate and to process the proactive control messages. If there is no condition 1 happening the node will come back to the Reactive 1 state. But while condition 1 happens, the node will try to join or to create an area. If it listens another node working in Reactive 3, Proactive 1 or Proactive 2 modes, then it will join the area unless in the area the number of nodes N is > Y. If N>Y the node remains in the same state waiting to listen to other area with less number of nodes.

- Proactive 1: In this state the router works using the OLSR features. If condition 1 is not fulfilled, the node will go to the Reactive 1 state. But when condition 1 is fulfilled, the node will continue working in this state unless it discovers a node working in the Reactive 1 or Reactive 2 states. Then it will go to the Proactive 2 state.

- Proactive 2 (Area Border Router): In this state the node works using the OLSR features but it has to understand the reactive routing messages (RREQ, RREP and RERR) because it needs to have in its routing table all the reactive 2 nodes connected with it.

  When an ABR (Area Border Router) receives a reactive routing message (RREQ, RREP or RERR) it must look for the destination. If the destination is inside its own area, then it answers to that message reactively. If not, it forwards them to all the others ABRs of its area. The intermediate nodes are purely reactive, but they know what they have to do with those packets looking at the two flags attached in all the packet explained in the section 3.2.8. These exit ABRs will change the flags again.

  If condition 1 is not fulfilled the node will go to the Reactive 1 state. But while condition 1 occurs the node will continue working in this state unless it lost all the connectivity with the nodes working in the Reactive 1 or Reactive 2 states. In this case it will go to Proactive 1 mode.

  A node goes to Initial State from every state when it is reset.

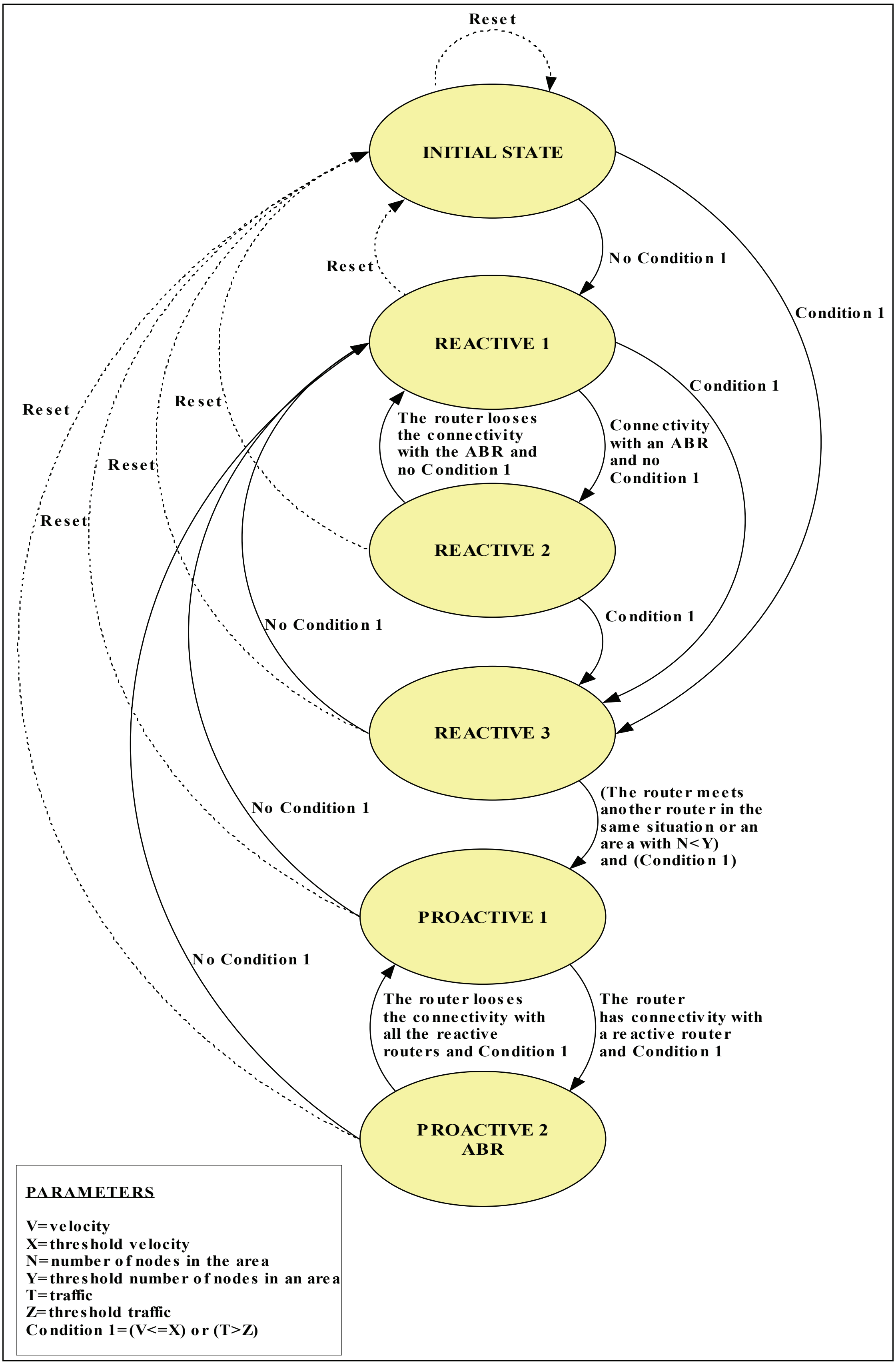

Reset
INITIAL STATE
No Condition 1
Reset
Condition 1
REACTIVE 1
Condition 1
Reset
Reset
The router looses the connectivity with the ABR and no Condition 1
Connectivity with an ABR and no Condition 1
Reset
Reset
REACTIVE 2
No Condition 1
Condition 1
REACTIVE 3
No Condition 1
(The router meets another router in the same situation or an area with N<Y) and (Condition 1)
PROACTIVE 1
No Condition 1
The router looses the connectivity with all the reactive routers and Condition 1
The router has connectivity with a reactive router and Condition 1
PROACTIVE 2
ABR
PARAMETERS
V=velocity
X=threshold velocity
N=number of nodes in the area
Y=threshold number of nodes in an area
T=traffic
Z=threshold traffic
Condition 1=(V<=X) or (T>Z)

Figure 3.2: State Transition of a Mobile Node. The text in the arrows represents a condition to change of state. If no one of these conditions happens, then the node will remain in the same state.

3.2.2. PROACTIVE 1 STATE: SAME WORKING AS OLSR

3.2.2.1. INTRODUCTION

The routing protocol OLSR (Optimized Link State Protocol) is defined in the [OLSR_03]. OLSR is an improvement of the pure link state protocol. Since it is a proactive routing protocol, the routes are always available. The route discovery process is based in the broadcast of the HELLO and TC (Topology Control) messages. Besides, the OLSR is based in the MPR (Multipoint Relay) concept [Benzaid et al., 2003].

In OLSR each node selects a group of neighbours as MPR. Hence, to each node MPR is associated a group of neighbours called MPR selectors. Only the nodes selected as MPR forward the control information in the network. Thanks to that it is possible to employ only a few transmissions and minimizing the load [Qayyum, et al., 2002].

The control information is updated periodically by the transmission of the control messages sent out by the MPR and the operation of OLSR does not depend on any central entity.

OLSR is considered as a good protocol for dense and big mobile networks, thanks to the optimization of the MPR nodes. It is also a hop to hop routing protocol. Thus, each node uses its local information to forward the packets to the destination.

3.2.2.2. OLSR PACKET FORMAT

The OLSR packets are contained in UDP (User Datagram Protocol) datagram. Up to the date, the specifications only consider IPv4 addresses. Each OLSR packet encapsulates one or more messages. The basic structure of an OLSR packet is showed in Figure 3.3 and has the following fields:

0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1

<table>
<tr><td colspan="2">Packet Lenght</td><td>Packet Sequence Number</td></tr>
<tr><td>Message Type</td><td>Vtime</td><td>Message Size</td></tr>
<tr><td colspan="3">Originator Address</td></tr>
<tr><td>Time To Live</td><td>Hop Count</td><td>Message Sequence Number</td></tr>
<tr><td colspan="3">Message</td></tr>
<tr><td colspan="2">Packet Lenght</td><td>Packet Sequence Number</td></tr>
<tr><td>Message Type</td><td>Vtime</td><td>Message Size</td></tr>
<tr><td colspan="3">Originator Address</td></tr>
<tr><td>Time To Live</td><td>Hop Count</td><td>Message Sequence Number</td></tr>
<tr><td colspan="3">Message</td></tr>
</table>

.. (etc.)

Figure 3.3: OLSR packet format

Packet Header: It consists in two fields:

- *Packet Length:* The length (in bytes) of the packet.
- *Packet Sequence Number:* Defines the sequence number of each OLSR packet. The Packet Sequence Number (PSN) must be incremented by one each time a new OLSR packet is transmitted.

Message Header: It consists in seven fields:

- *Message Type:* Indicates the type of message sent. The Types in the range of 0-127 are reserved.
- *Vtime:* This field indicates for how much time after reception a node must consider the information contained in the message as valid, unless a more recent update to the information is received. The validity time is represented by its mantissa (four highest bits of Vtime field).

$$Vtime = C*(1+a/16)*2^b \quad [seconds]$$

Where

a	Is the integer represented by the four highest bits of Vtime field.
b	Is the integer represented by the four lowest bits of Vtime field.
C	It is proposed a constant value of 1/16 seconds (0.0625 seconds).

- *Message Size:* This gives the size of this message, counted in bytes and measured from the beginning of the “Message Type” field and until the beginning of the next “Message Type” field (or – if there are no following messages – until the end of the packet).
- *Originator Address:* This field contains the main address of the node, which has originally generated this message. This field should not be mistaken with the source address from the IP header, which is changed each time to the address of the intermediate interface which is re-transmitting this message. The Originator Address field never changes in retransmissions.
- *Time To Live (TTL):* This field contains the maximum number of hops a message will be retransmitted. Before a message is retransmitted, the Time To Live must be decremented by 1. When a node receives a message with a TTL equal to 0 or 1, the message is not retransmitted under any circumstances. Normally, a node would not receive a message with a TTL of zero. Thus, by setting this field, the originator of a message can limit the flooding radius.
- *Hop Count:* This field contains the number of hops a message has attained. Before a message is retransmitted, the Hop Count must be incremented by 1. Initially, this is set to '0' by the originator of the message.
- *Message Sequence Number:* While generating a message, the "originator" node will assign a unique identification number to each message. This number is inserted into the Sequence Number field of the message. The sequence number is increased by 1 (one) for each message originating from the node. Message sequence numbers are used to ensure that a given message is not retransmitted more than once by any node.

3.2.2.3. HELLO MESSAGES

OLSR is based in the detection of the neighbours to one hop distance. Each node has to detect the neighbour nodes with which it has a direct and symmetric link. The neighbours discovery process is performed by the broadcast of HELLO messages. The HELLO messages are generated for every node in the network. In Figure 3.4 the HELLO message broadcast process is illustrated.

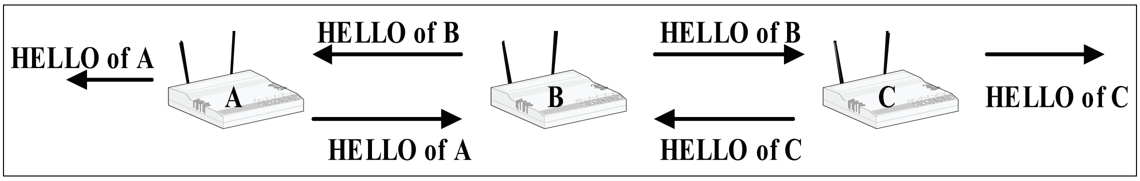


Figure 3.4: Broadcast and generation of the HELLO messages

To achieve the neighbour discovery to one hop, each node broadcasts HELLO messages. These messages are broadcasted to all the neighbours to one hop, but are not forwarded for the nodes that receive them. The HELLO message has:

- A list of the neighbour addresses that have a symmetric link.
- A list of the neighbour addresses that have been 'listened'.
- A list of the neighbours that have been selected as MPR.
- A list of the neighbours that are ABRs.
- A list of the neighbours that are R2 nodes.

3.2.2.3.1. HELLO MESSAGE FORMAT

The Hello Message Format is sent as the data-portion of the general packet format described in Figure 3.3. The field "Message Type" is set to HELLO_MESSAGE and the TTL field set to 1. The HELLO message format, as defined in [OLSR_03] is showed in Figure 3.5.

0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1

Reserved	Number of nodes	HTime	Willingness
Link Code	Reserved	Link Message Size	
Neighbour Interface Address			
Neighbour Interface Address			
…			
Link Code	Reserved	Link Message Size	
Neighbour Interface Address			
Neighbour Interface Address			
…			

Figure 3.5: HELLO message format

Description of the fields of the HELLO message:

Number of nodes: This field does not exist in the normal OLSR HELLO message. In the Penaguila routing protocol it is introduced with the objective that each node that works in an area, sets here the number of nodes that it has in its routing table working into the area. If a node working in the Reactive 3 mode receives a HELLO message with this number greater than 90, then the router will not join the area.

HTime: This field specifies the HELLO emission interval used by the node on this particular interface, i.e., the time before the transmission of the next. The HELLO emission interval is represented by its mantissa (four highest bits of Htime field) and by its exponent (four lowest bits of Htime field). In other words:

HTime=C*(1+a/16)*2^b [in seconds]

Where:

a Represents the 4 highest bits of HTime field.

b Represents the 4 lowest bits of HTime field.

C Represents a constant value and is defined as:

$$C=1/16 \text{ seconds } (0.0625 \text{ seconds})$$

Willingness: This field specifies the willingness of a node to carry and forward traffic for other nodes. If the willingness of a node is defined as WILL_NEVER, this will never be selected as MPR node.

Link Code: This field specifies information about the link between the interface of the sender and the following list of neighbour interfaces. It also specifies information about the status of the neighbour. Link codes, not known by a node, are silently discarded.

The link code specifies, for example, if the neighbours have been selected as MPR, as ABR or as R2 node by the sender.

Link Message Size: This is the size of the link message, counted in bytes and measured from the beginning of the "Link Code" field and until the next "Link Code" field (or – if there are no more link types – the end of the message).

Neighbour Interface Address: The address of an interface of a neighbour node.

When a node receives a HELLO message, it can update the information of the neighbour with the address of the node sending the message. But if necessary and by security reasons it can ignore the HELLO messages.

3.2.2.4. TC (TOPOLOGY CONTROL) MESSAGES

The MPR nodes broadcast the TC messages. Figure 3.6 illustrates the MC message broadcast process by a MPR node. The MC messages carry the list of neighbours that have selected the emitter node as MPR, that is, the MPR selector set. The information of the TC messages is necessary to calculate the routing table.

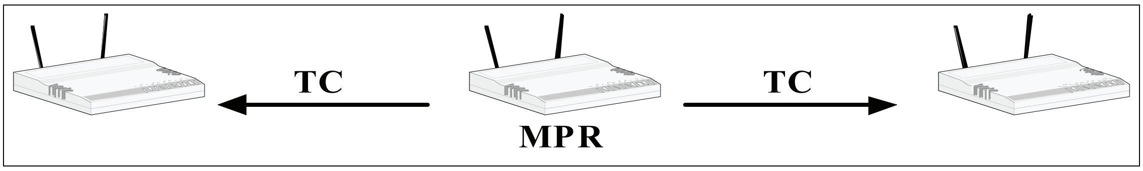


Figure 3.6: Broadcast of the TC messages

3.2.2.4.1. TC MESSAGE FORMAT

The TC format message is showed in Figure 3.7. Next, a description of the TC message fields is presented:

Advertised Neighbour Sequence Number (ANSN): A sequence number is associated with the advertised neighbour set. Every time a node detects a change in its advertised neighbour set, it increments this sequence number. This number is sent in this ANSN field of the TC message to keep track of the most recent information. When a node receives a TC message, it can decide on the basis of this Advertised Neighbour Sequence Number, whether the received information about the advertised neighbours of the originator node is more recent than what it already has or not.

Reserved: Field reserved for the future use, set to '0000000000000000' according with the RFC 2636.

Advertised Neighbour Main Address: This field contains the main address of a neighbour node. All main addresses of the advertised neighbours of the Originator node are put in the TC message. If the maximum allowed message size (as imposed by the network) is reached while there are still advertised neighbour addresses which have not been inserted into the TC-message, more TC messages will be generated until the entire advertised neighbour set has been sent. Extra main addresses of neighbour nodes may be included, if redundancy is desired.

0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1

<table>
<tr><td>ANSN</td><td>Reserved</td></tr>
<tr><td colspan="2">Advertised Neighbour Main Address</td></tr>
<tr><td colspan="2">Advertised Neighbour Main Address
…………</td></tr>
</table>

Figure 3.7: TC message Format

3.2.2.5. BASIC FUNCTIONALITIES OF OLSR

Neighbour Detection

Each node detects the nodes having direct link with itself. Each node sends HELLO messages, containing the list of the neighbours known by the node and the state of their links (the link can be symmetric, asymmetric, MPR or lost). The Hello message is broadcasted to all the neighbours at one hop distance. These nodes do not forward the Hello message.

By means of the Hello messages broadcast, it is possible to discover neighbours to one and two hops. A parameter called Neighbour-hold-time is associated to these neighbourhoods. This time allows the deleting of those expired entries of neighbours.

Topology dissemination process

Each node of the network maintains information about the topology, obtained by the TC messages. Each node that has been selected as a MPR spreads TC messages. The TC messages are spread throughout the entire network and are only forwarded by MPR nodes. Hence, the load is smaller. The TC message is used to spread the MPR selector list. In this way, the MPR node is declared as the node to one hop of the MPR selector nodes and in such a way that each MPR selector node is reachable directly through the MPR node. [Benzaid et al., 2002].

3.2.2.6. MPR (MULTIPOINT RELAY)

The interest of the MPR is to reduce the load in the spreading of control messages through the network. Each node selects a set of neighbour nodes that can forward its messages to other nodes at two hops. That set of nodes is the MPR set of this node. As showed in Figure 3.8, the neighbours of the node N that are not within the MPR set receive and process the information of the spreading messages, but they don't forward the information coming from the node N.

Each node selects the MPR nodes. These MPR nodes must have connectivity with all the nodes at two hops.

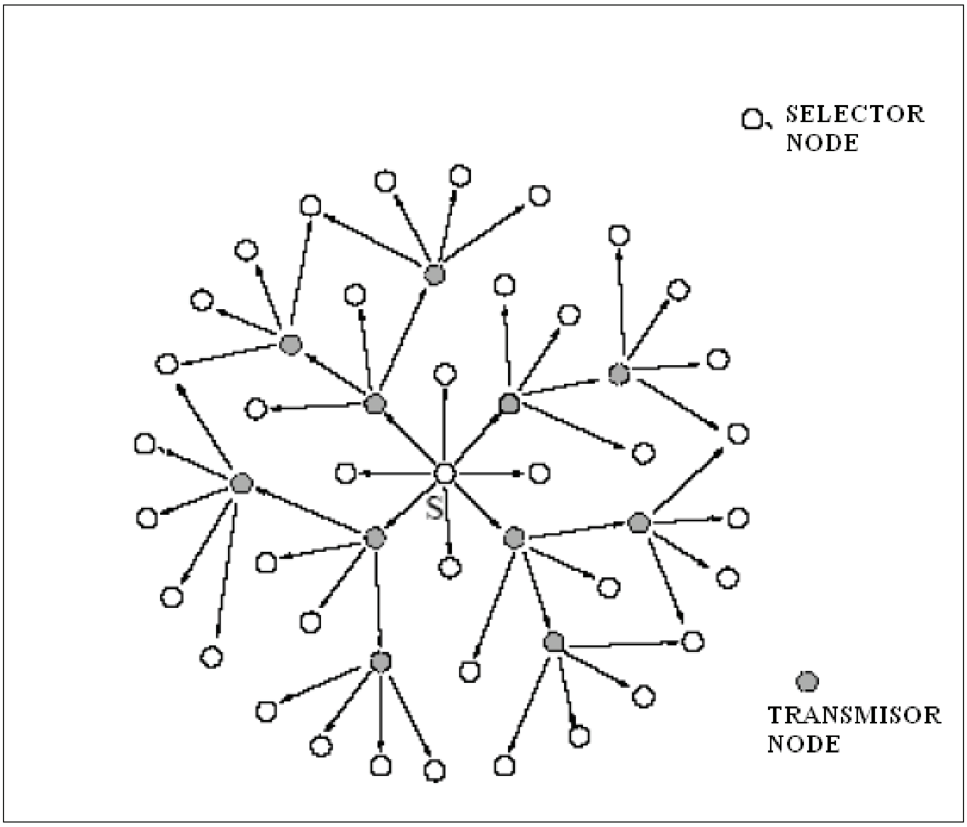


Figure 3.8: Selection of the MPR nodes process

At the same time, the MPR nodes maintain information about the set of neighbours to one hop that have selected it as MPR. This set is known as MPR selector set. This information is acquired by the Hellos messages received from the neighbours to one hop.

3.2.2.6.1. Selection of MPR nodes process

Each node selects a set of neighbours to one hop as MPR. These MPR must be able to reach all the neighbours to two hops distance. When a node selects the MPR nodes, the link state with these nodes must be changed from SYM_LINK to MPR_LINK in the neighbour table. The MPR_Seq_Num value is incremented in one.

The set of MPR is recalculated if any of these cases happens:

- A change in the neighbourhood to one hop is detected. It can be a broken link or a new neighbour added.
- A change in the neighbourhood to two hops is detected. It can be a broken link or a new neighbour added between a neighbour to one hop and another neighbour to two hops.

3.2.2.7. HNA MESSAGES

The HNA (Host Network Association) message carries information about the address and network mask. It is broadcasted periodically, each HNA_INTERVAL.

Although the HNA message is not part of the OLSR core, this is used as part of an auxiliary functionality to extend the possibility to introduce external routing information to an OLSR MANET. The TC and HNA messages are similar since both are used to indicate the reachability of a particular node. Besides, both messages are spread and forwarded through the network by the MPR nodes. An important difference is that the information of a new TC message can cancel previous information (if the received sequence number is greater), while the information coming from the HNA messages, will only be replaced when its lifetime finishes.

3.2.2.7.1. HNA message format

Figure 3.9 illustrates the HNS message structure, consisting in the fields: Network address and Netmask.

Network address
Netmask

Figure 3.9: HNA message format

We are going to see the definition of the HNA message fields:

Network address: (32 bits) This field is used to declare the network address.
Netmask: (32 bits) This field is used to declare the “mask” used in the network.

3.2.2.7.2. Generation and sending of the HNA messages

The HNA messages are spread in the entire network by the MPR nodes. These messages carry the network address and the net mask. Within the Ad Hoc networks can exist Access Routers (AR) that are the routers in charge of providing connectivity to the MANET with the other wired networks and communicate with the ad hoc network nodes via wireless. They have the wired and the ad hoc network protocols. Figure 3.10 illustrates that only the ARs generate the HNA messages, but are forwarded by those nodes selected as MPR.

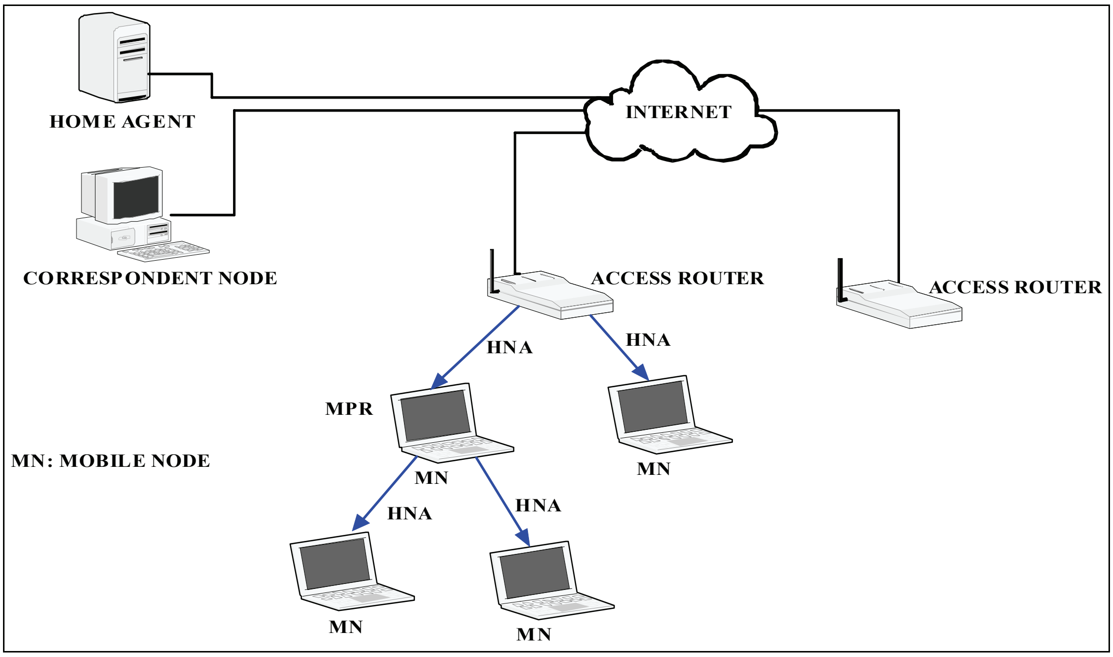


Figure 3.10: HNA messages generation and broadcasting

3.2.2.7.3. Processing of the HNA messages

The nodes that receive a HNA message keep a register with the information provided by these messages. The A_gateway_addr variable is used to declare the address of the node origin of the HNA message. The A_network_addr is used to declare the information that is provided by the HNA message Network Address field. The A_netmask variable is used to declare the information that is provided by the HNA message Netmask field. Finally the A_time variable is used as a timer for the depuration of the registers that have expired.

The HNA messages processing is done following the next algorithm:

1. If the transmitter (not the originator) of the HNA message is not a neighbour to one hop, the message is ruled out.

2. Otherwise, the HNA message is processed and the information given by the network address and netmask fields is considered.

    2.1 If there is a register, where:

        A_gateway_addr = originator address

A_network_addr = network address

A_netmask = network mask

Then, the holding time is updated:

A_time = current time + validity time

2.2 If there is not a register with the information provided by the HNA message, then the A_gateway_addr, A_network_addr, A_netmask variables are updated using the information of the HNA message. In the same way the A_time timer is initialized.

3.2.2.8. ROUTING TABLE CALCULATION

Each node maintains a routing table. This table is calculated with the information obtained from the neighbour tables and from the topology. The nodes that receive a TC message store pairs of linked nodes (previous hop, node), where the “nodes” are the addresses stored in the TC messages list. To find a route to a remote node R, it a pair (Previous_hop, R) must be found. Once it has been found, this previous hop becomes an intermediate destination (Destination_inter) and now it looks for a pair (Previous_hop, Destination_inter). This process is realized successively until it finds a Previous_hop node in the neighbours set of the node that look for the route. In Figure 3.11 the calculation of the complete route from the source to the destination is shown.

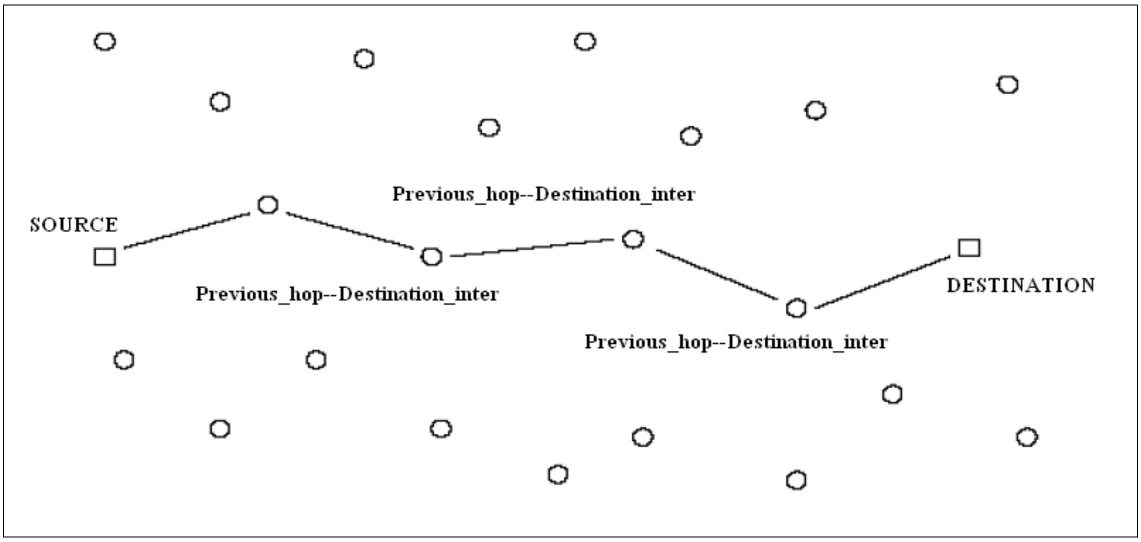

SOURCE
Previous_hop--Destination_inter
Previous_hop--Destination_inter
Previous_hop--Destination_inter
DESTINATION

Figure 3.11: Calculation of a route using the topology table

The following is an example of the calculation of routes process used as part of the routing table maintenance:

1. All the entries in the routing table are removed.
2. New entries in the routing table are added, initiating with the symmetric neighbours to one hop (h=1). These nodes are declared as destination nodes within the routing table. For each entry of the destination nodes, a new entry in the table is added where the destination and next hop addresses are both declared using the neighbour IP address (destination address) and the distance (R_dist) is defined as 1.
3. New entries are added for the nodes being to a distance greater than one hop. The process begins with the nodes at distance h=2 hops and then the h value is incremented in 1. The process will finish when there are no entries to add in an iteration.

   For each entry in the topology table, if its destination address is not in any register of the routing table and its last hop belongs to the destination address of an entry with distance h, an entry is added in the routing table where:

   - The destination is set to the destination address in the topology table.
   - The next hop is set to the next hop of the route entry whom destination is the same as the last hop address.
   - The distance is set to h+1.

4. When the table is calculated, the entries of the topology tables that are used in the calculation of the routes can be deleted, for it to have more memory resources.

The routing table is calculated again each time a change in the topology is detected. In such a way, the updating of the routing table is done when there is a change in:

- The set of neighbours nodes.
- The set of neighbours nodes to two hops.

The topology set is calculated again when a neighbour appears or is lost, when a couple of topology is created or removed.

3.2.3. REACTIVE 1 STATE: SAME WORKING AS AODV

3.2.3.1. INTRODUCTION

A node in the Reactive 1 state has the same operation that a node implementing AODV. Therefore, it is a point-to-point routing protocol. The routes are established on demand [MANET_03]. That means that the routers working in reactive mode do not maintain the routes updated all the time, but are discovered and maintained only when necessary.

The routes are discovered during the Route Discovery process [MANET_02] where the source node looks for a route to a destination node to send information to it. This process ends when the source node knows the path.

3.2.3.2. CHARACTERISTICS

AODV presents the following characteristics:

- *Low control signalization:* There are no periodic updates with information about the routing, since it is reactive.
- *Loop prevention:* There is a mechanism to solve the loops.
- *Only works with bidirectional links.*

To prevent loops each node maintains a sequence number (destination sequence number) that evaluates the validity of the associated routing information and increases in one each time a node sends a new RREQ. If a node receives a RREQ addressed to itself before it generates the RREP it must update its sequence number $NumSeq_D$ to the maximum value between its current sequence number $NumSeq_{D\_current}$ and the destination sequence number carried in the RREQ (RREQ.NumSeq) plus one:

$\text{NumSeq}_{D} = \text{Max}(\text{NumSeq}_{D\_current}, \text{RREQ.NumSeq}+1)$ (formula 1)

The sequence numbers allows selecting the freshest route to a destination. If a source node or an intermediate node receives two routes with the same destination sequence number, then it will choose the route with less number of hops.

3.2.3.3. AODV ROUTING TABLE

AODV uses routing tables that store:

- *Destination IP Addresss.*
- *Destination Sequence Number.*
- *Lifetime (expiration or deletion time of the route).*
- *Hop Count (number of hops needed to reach destination).*

Each routing table entry has associated a lifetime timer. If a route is not used, then the timer expires. On the other hand, if the route is used or if "Hello" messages are received, the timer is updated.

3.2.3.4. AODV OPERATION

The Route Discovery process can be described as follows (Figure 3.12):

- When a source node wants to send packets to any destination, firstly it has to check in its routing table if there already exists an updated route to that destination. If this route exists, the source node will use it to send the packet to the next hop in the direction to the destination. If the route does not exist, the node will begin a Route Discovery process sending a RREQ in broadcast mode.
- Any node in the network that knows an updated route to the destination (included the destination) can send a RREP to the source node.

- The information about the route is maintained in the routing table of each node.
- The obtained information with the sending of the RREQ and RREP packets is stored together with other information in the routing table.
- The sequenced numbers are used to delete old routes.
- The routes with old sequence numbers are deleted.

If a node begins a Route Discovery process, then it sends a RREQ packet in broadcast mode with the following information:

- *Source IP address.*
- *Source sequence number.*
- *Destination IP address.*
- *RREQ ID (broadcast ID).*
- *Hop count.*

The broadcast ID of the RREQ is a number that is increased each time a node begins to send a RREQ.

When a node receives a RREQ it must:

- Check the broadcast ID of the RREQ and the source node IP address to know if it has already received it. Each node maintains a register of the source node IP address and of the broadcast ID of the RREQ during a time for each RREQ received; this time depends on the network congestion, size and topology.
- If the node sees that the RREQ has been received previously, then it rejects the packet (Figure 3.14).
- If the node did not receive the packet before, then it records that information and processes the RREQ.
- The RREQ processing is done as follows: The node establishes an entry in the routing table recording the reverse path (Figure 3.13). The fields recorded are (besides other fields):

  - Source node sequence number.
  - Number of hops to source node (it is increased by one the recorded value in the RREQ).
  - IP address of the neighbour node that sent it the RREQ.

The reverse path has a lifetime and when this lifetime expires, the associated information is deleted. The reverse path has its utility when the node later receives a RREP that must be delivered to the source through the reverse path created.

To be able to answer the RREQ:

- The node must have a routing table with an entry to the destination that has not expired.
- On the other hand, the destination sequence number stored in the routing table must be greater or equal to the destination sequence number of the RREQ, that is:

$$SeqNum_{table}>=SeqNum_{RREQ}$$

If both conditions are true, then the node increases the Hop Count of the RREQ and it generates a RREP.

Otherwise, the node increases the Hop Counter of the RREQ and forwards it in broadcast mode to its neighbours because it has not a fresh route to that destination.

The RREP packet contains the source IP address and the destination IP address. Besides, if it is the destination who answers (Figures 3.15, 3.16 and 3.17) then:

- It puts its sequence number in the packet (first, it calculates this sequence number as explained in the formula 1).
- Initializes the hop counter from zero.
- Puts in the lifetime timer a time value.

- Sends the packet in the direction to the source node choosing as a first hop the same node from which it received the RREQ, since this node has established the reverse path.

If it is an intermediate node who responds:

- It puts the destination sequence number in the packet.
- It puts in the number of hops counter the number of hops from this node to the destination.
- It puts in the lifetime timer a time value.
- It sends the packet in the direction to the source node choosing as a first hop the same node from which it received the RREQ, since this node has established the reverse path.

When an intermediate node receives a RREP:

- Increases by one the number of hops counter of the RREP.
- Establishes a "forward path" (Figure 3.18), representing an entry towards the destination in its routing table. The intermediate node uses the node from which it received the RREP as next hop to the destination. Therefore, all the intermediate nodes from the source to the destination will know this path to transmit data if it is chosen by the source.
- This entry contains:
    - Destination node IP address.
    - Next hop IP address.
    - Hop number to the destination (adds one to the counter).
    - Lifetime timer.
    - Destination sequence number.
- The node forwards the RREP to the next hop in the way to the source.

If a node receives a RREP to a destination from more than a neighbour node:

- It forwards the first RREP.

- It may later forward another RREP only if:
    - The destination sequence number carried in this last RREP is greater.

    or

    - The number of hops counter carried in this last RREP is smaller.
- Otherwise, it rejects this last RREP.

When a source node receives a RREP, it can begin to use the stored route to send data packets (Figure 3.19). If a source node receives many RREP's it will select the route with the greatest destination sequence number and the smaller number of hops to this destination.

It is interesting to notice that AODV not necessarily brings the shortest path in terms of number of hops from a source node to a destination node. In AODV each node accepts and processes only one RREQ, while it rejects those RREQ's reaching it later and that have associated the same broadcast ID of the RREQ and IP address of the source node than the RREQ received before. For this reason it is not possible that AODV always achieve the shortest path, because when the RREQ's are broadcasted looking for the path, an intermediate node has previously received other RREQ from another path, causing that the following RREQ pertaining to this path will be rejected. However, between the chosen paths, it selects the one with the smallest number of hops.

Each node checks the links state communicating itself with the next hop through a route that is being used at this moment (active route). In case that it detects the breakage of the link, this node invalidates in its routing table all that entries to the destinations that are not available now because of the breakage of the link.

Besides, the node sends a Route Error packet (RERR) to the source node to inform it about that breakage. This process is known as the Route Maintenance process [MANET_04]. The RERR informs about those destinations that are now unreachable. If there are precursor nodes (between the source node and the node detecting the link breakage) that were using the link, the RERR is propagated in broadcast mode or if not, in unicast mode.

When a node receives a RERR it checks if the node that sent the message is its next hop to any of the destinations, and if that is true, the node invalidates these entries in its routing table and forwards the RERR to the source. When finally the RERR reaches the source, the source can decide whether to begin a new Route Discovery process if it considers it necessary or not.

To know if there is connectivity [EV_AODV], each node sends "Hello" messages with the node IP address, its current sequence number and the link lifetime to its neighbours periodically. Then each neighbour can take advantage of this information to update the routing table entry to this neighbour. If during a determined interval of time a node stops receiving "Hello" messages from a concrete neighbour, it deletes the entry of the routing table associated to that neighbour. The routing message exchange is not necessary if there exists another mechanism to ascertain if connectivity is existing or not, as can be retro alimentation from the data link layer.

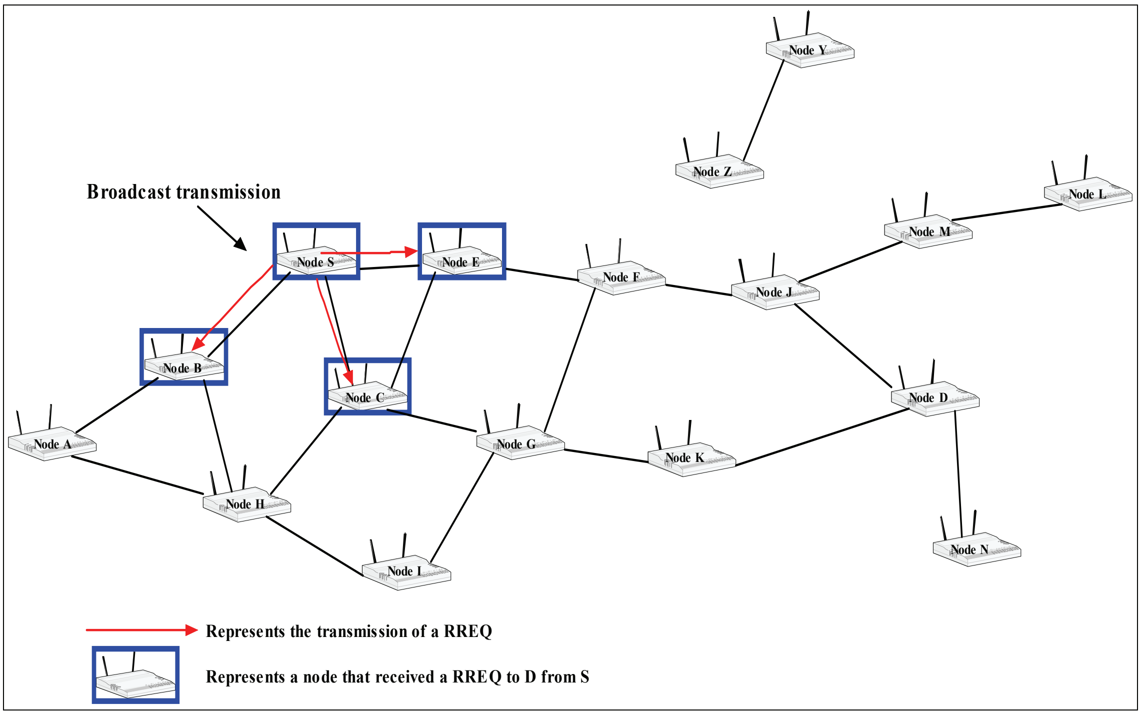


Figure 3.12: Route Discovery (1). Route Discovery in an ad hoc network using the AODV routing protocol to send data from a source 'Node S' to a destination 'Node D' (1)

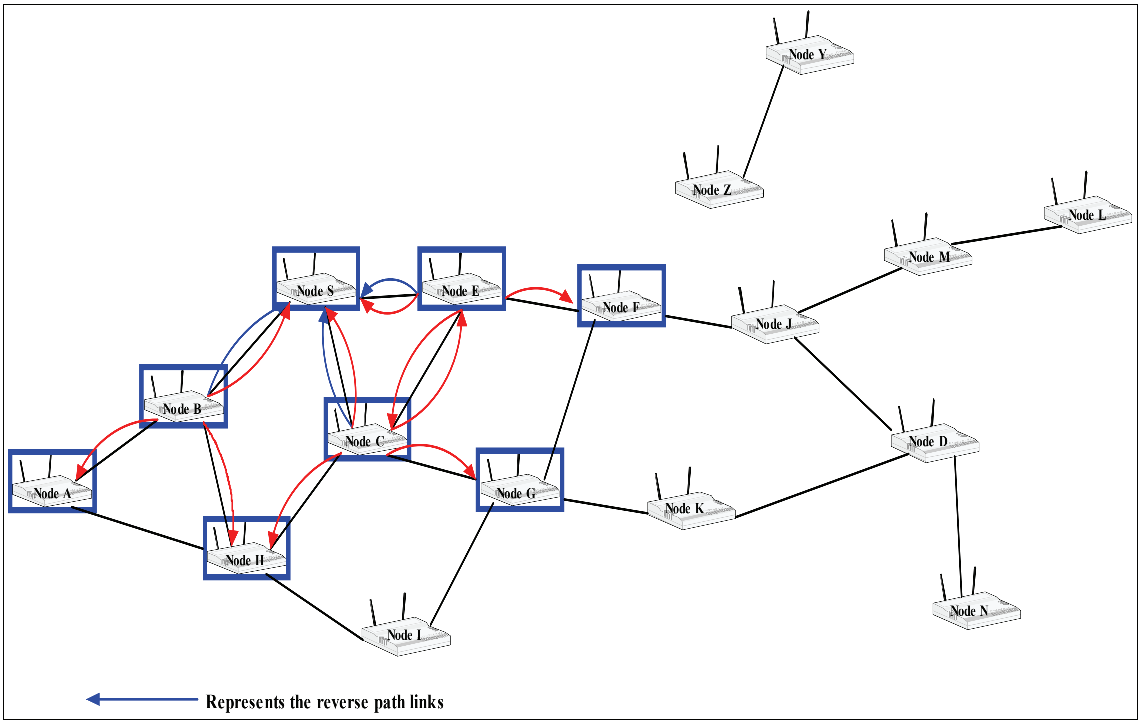


Figure 3.13: Route discovery (2)

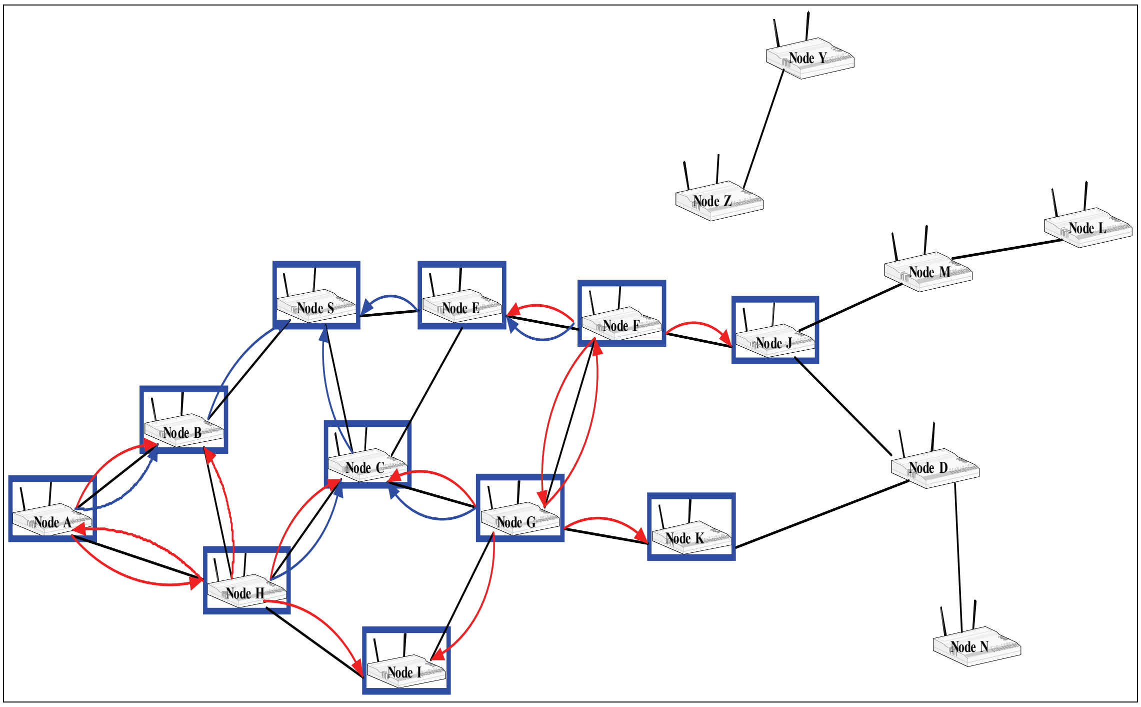


Figure 3.14: Route Discovery (3). The 'Node C' receives the RREQ from nodes G and H, but it don't make a broadcast because it being already sent once (3)

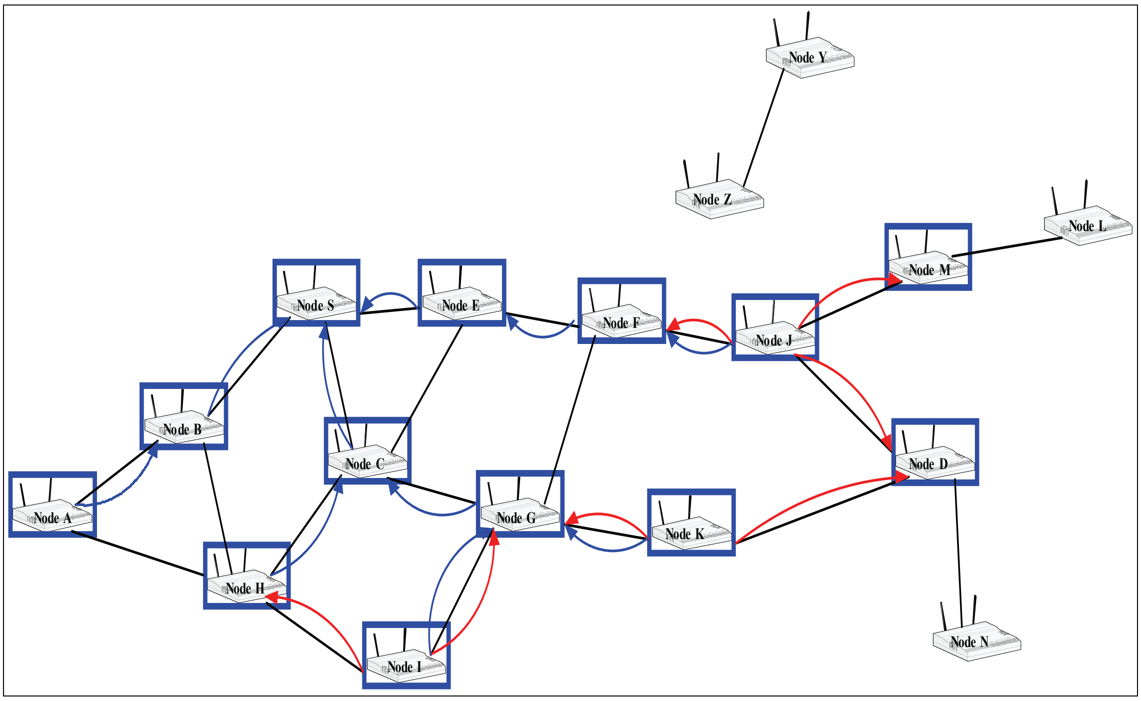


Figure 3.15: Route Discovery (4)

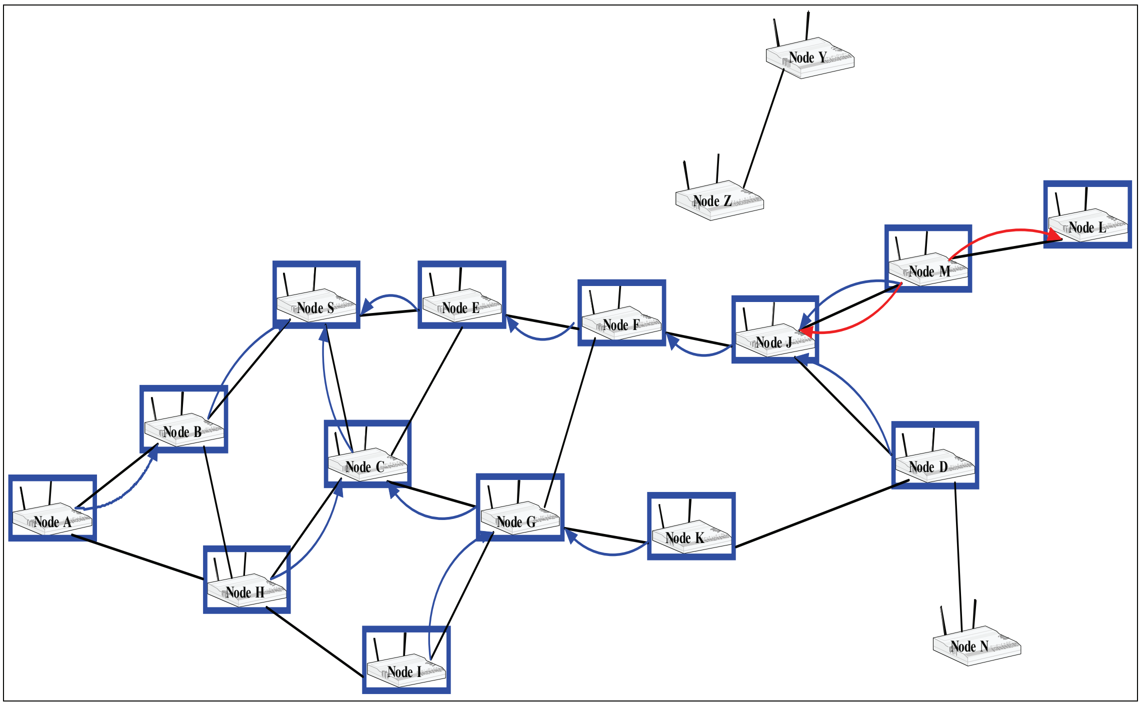


Figure 3.16: Route Discovery (5). The 'Node D' don't broadcast the RREQ, because is the destination node in the Discovery Route process. (5)

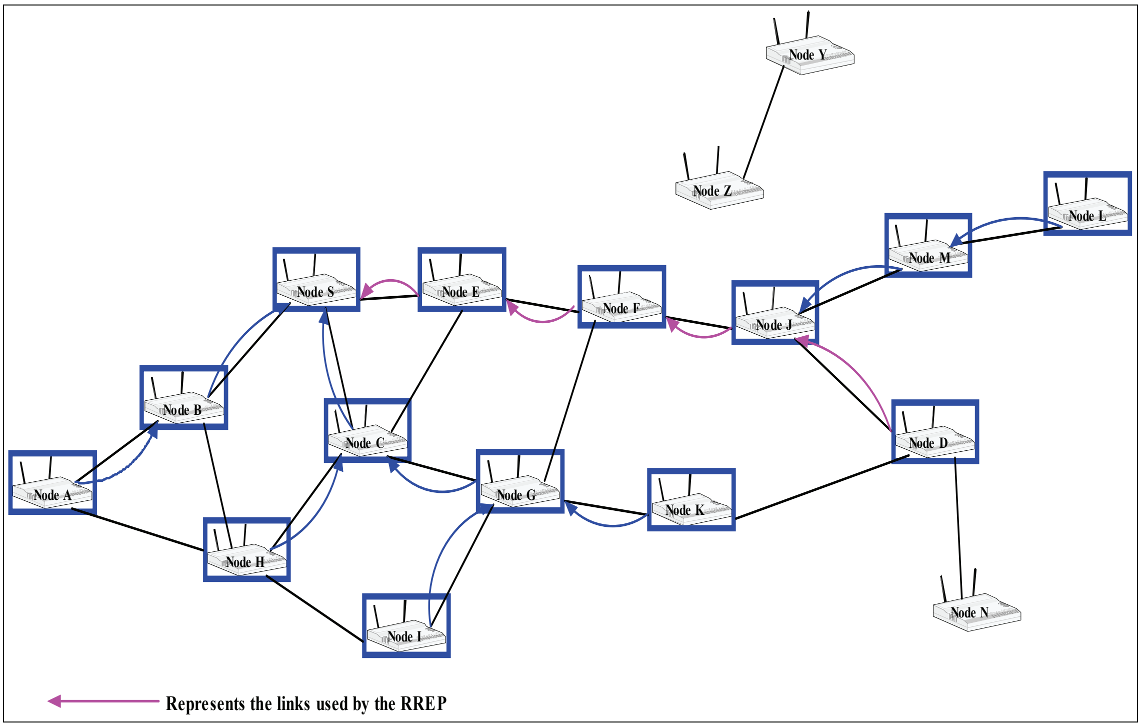


Figure 3.17: Route Discovery (6)

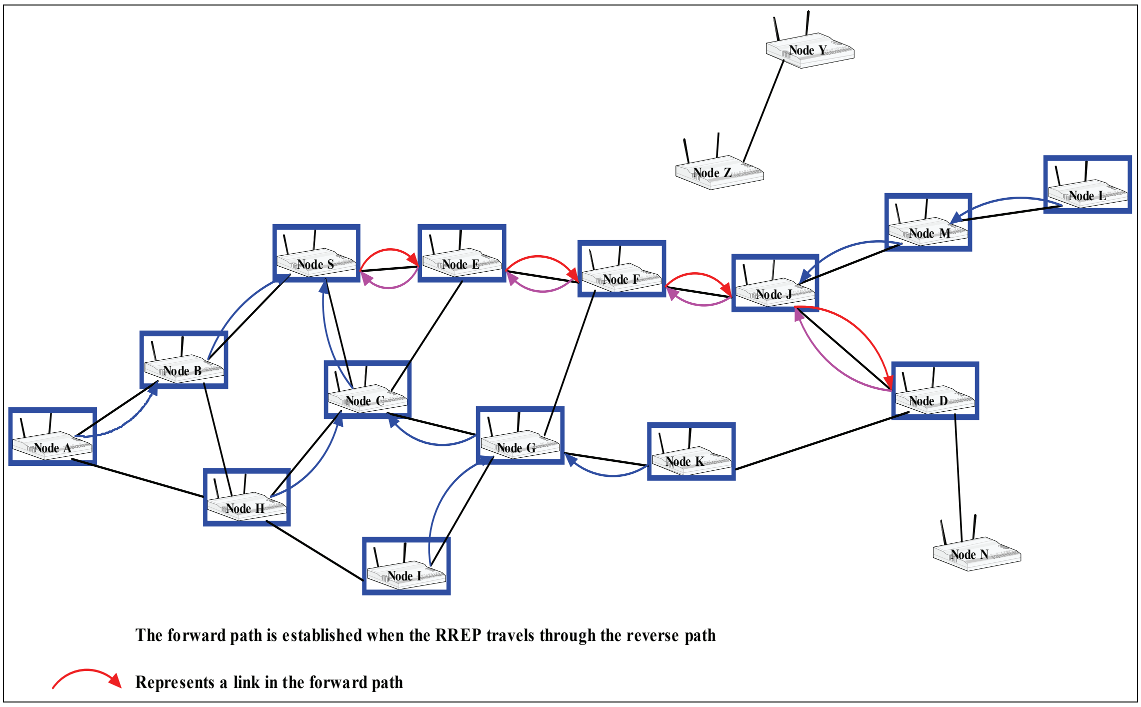
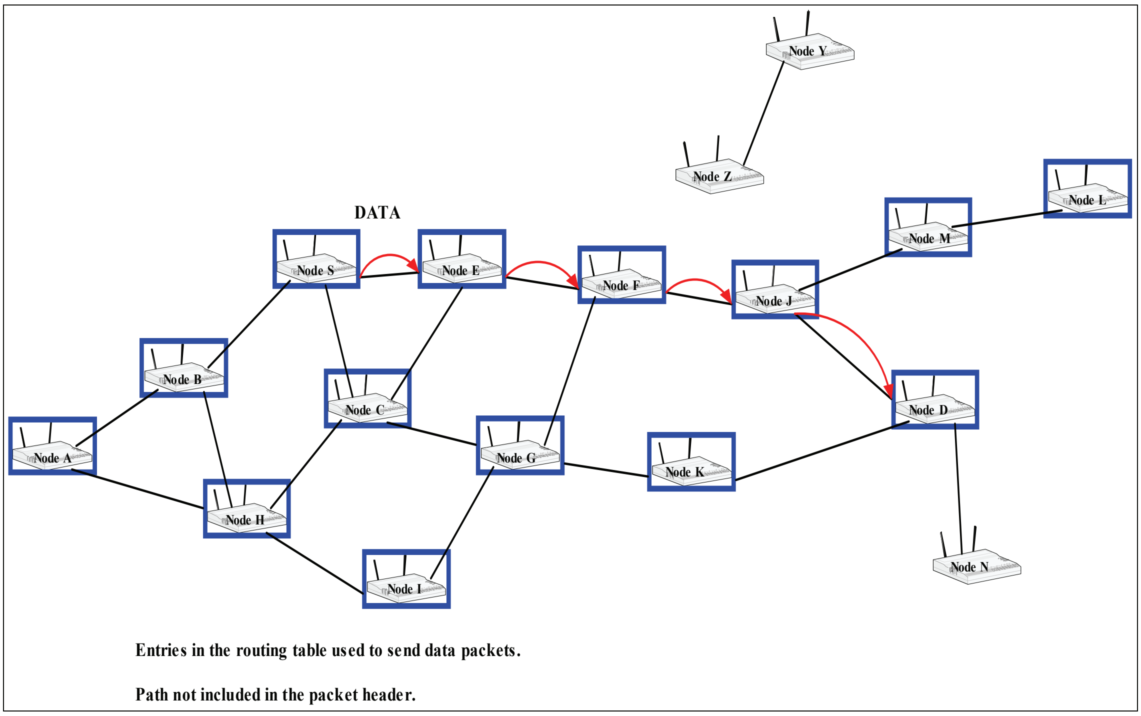

Node Y
Node Z
Node L
Node M
Node S
Node E
Node F
Node J
Node B
Node C
Node D
Node A
Node G
Node K
Node H
Node I
Node N
The forward path is established when the RREP travels through the reverse path
Represents a link in the forward path

Figure 3.18: Route Discovery (7)



Entries in the routing table used to send data packets.

Path not included in the packet header.

Figure 3.19: Route Discovery (8)

3.2.3.5. ROUTE REQUEST (RREQ) MESSAGE FORMAT

0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1
Type								J	R	G	D	U	Reserved											Hop Count							
RREQ ID																															
Destination IP Address																															
Destination Sequence Number																															
Originator IP Address																															
Originator Sequence Number																															

Figure 3.20: Route Request message format

The format of the RREQ message is showed above, and contains the following fields:

Type: 1.

J: Join flag; reserved for multicast.

R: Repair flag; reserved for multicast.

G: Gratuitous RREP flag; indicates whether a gratuitous RREP should be unicast to the node specified in the Destination IP address field (see sections 6.3 and 6.6.3 of [AODV_03]).

D: Destination only flag; indicates only the destination may respond to this RREQ (see section 6.5 of [AODV_03].

U: Unknown sequence number; indicates the destination sequence number is unknown (see section 6.3 of [AODV_03]).

Reserved: Sent as 0; ignored on reception.

Hop Count: The number of hops from the Originator IP Address.

RREQ ID: A sequence number uniquely identifying the particular RREQ when taken in conjunction with the originating node's IP address.

Destination IP Address: The IP address of the destination for which a route is desired.

Destination Sequence Number: The latest sequence number received in the past by the originator for any route towards the destination.

Originator IP Address: The IP address of the node which originated the Route Request.

Originator Sequence Number: The current sequence number to be used in the route entry pointing towards the originator of the route request.

3.2.3.6. ROUTE REPLY (RREP) MESSAGE FORMAT

<table>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td></tr>
<tr><td colspan="8">Type</td><td>R</td><td>A</td><td colspan="9">Reserved</td><td colspan="5">Prefix Sz</td><td colspan="8">Hop Count</td></tr>
<tr><td colspan="32">Destination IP Address</td></tr>
<tr><td colspan="32">Destination Sequence Number</td></tr>
<tr><td colspan="32">Originator IP address</td></tr>
<tr><td colspan="32">Lifetime</td></tr>
</table>

Figure 3.21: Route Reply message format

The format of the Route Reply message is illustrated above, and contains the following fields:

Type: 2

R: Repair flag; used for multicast.

A: Acknowledgment required; see sections 5.4 and 6.7 of [AODV_03].

Reserved: Sent as 0; ignored on reception.

Prefix Size: If non zero, the 5-bit Prefix Size specifies that the indicated next hop may be used for any nodes with the same routing prefix (as defined by the Prefix Size) as the requested destination.

Hop Count: The number of hops from the Originator IP Address to the Destination IP *Address*: For multicast route requests this indicates the number of hops to the multicast tree member sending the RREP.

Destination IP Address: The IP address of the destination for which a route is supplied.

Destination Sequence Number: The destination sequence number associated to the route.

Originator IP Address: The IP address of the node which originated the RREQ for which the route is supplied.

Lifetime: The time in milliseconds for which nodes receiving the RREP consider the route to be valid.

3.2.3.7. ROUTE ERROR (RERR) MESSAGE FORMAT

<table>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td></tr>
<tr><td colspan="8">Type</td><td>N</td><td colspan="15">Reserved</td><td colspan="8">Hop Count</td></tr>
<tr><td colspan="32">Unreachable Destination IP Address (1)</td></tr>
<tr><td colspan="32">Unreachable Destination Sequence Number (1)</td></tr>
<tr><td colspan="32">Additional Unreachable Destination IP Addresses (if needed)</td></tr>
<tr><td colspan="32">Additional Unreachable Destination Sequence Numbers (if needed)</td></tr>
</table>

3.22: Route Error message format

The format of the Route Error message is illustrated above, and contains the following fields:

Type: 3

N: No delete flag; set when a node has performed a local repair of a link, and upstream nodes should not delete the route.

Reserved: Sent as 0; ignored on reception.

DestCount: The number of unreachable destinations included in the message; MUST be at least 1.

Unreachable Destination IP Address: The IP address of the destination that has become unreachable due to a link break.

Unreachable Destination Sequence Number: The sequence number in the route table entry for the destination listed in the previous Unreachable Destination IP Address field.

3.2.4. REACTIVE 2 STATE

All the nodes within an area must work in proactive mode, understanding the control messages necessary to get the routing tables (as TC and Hello messages). But also the nodes outside the area with connectivity with an area border router (that have connectivity with an area) need to understand these messages. These nodes are working in the Reactive 2 mode. Figure 3.23 illustrates an example where there are nodes working in proactive mode and nodes working in reactive mode. The Node b is working in the Reactive 2 mode and it must understand the proactive routing protocol control messages, like Nodes c and d (that are within the area). The Node b answers to these messages only if necessary, but only to Node c. That means that Node b does not propagate the proactive control messages to nodes outside the area.

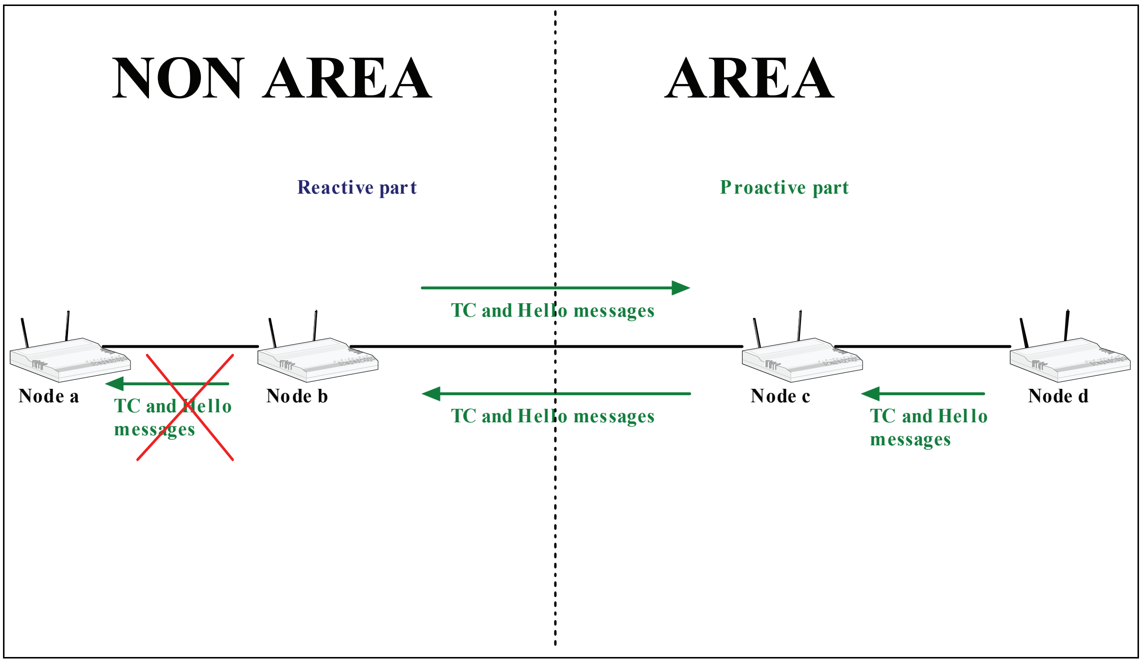


Figure 3.23: Example of R2 node: The node b is a R2 node. It has in its routing table all the nodes within the proactive area (c and d). Also, the nodes inside the area have in its routing table the node b and select him as a gateway to communicate with node a.

As said earlier, in Figure 3.23 Node b is working using the AODV features, but also uses the proactive information coming from the area to create its routing table since it needs to know how to route to Node c.

Under normal circumstances, an OLSR node (P1 node) that does not have a routing table entry for the destination of the data packet will simply drop the packet. However, the destination may be an AODV node. Hence, the OLSR node will send the data packets to the nearest ABR and this one to a Reactive 2 node. Then, the Reactive 2 node will need to initiate route discovery on behalf of the OLSR node. In order to advertise the Reactive 2 nodes connectivity to other AODV nodes as well as OLSR nodes which are separated by AODV networks, the Reactive 2 node will broadcast HNA messages indicating that it is the default “gateway” for the OLSR nodes in the network. In the case of an OLSR node receiving HNA messages from multiple Reactive 2 nodes, the nearest Reactive 2 node will be selected. These HNA messages also enable a Reactive 2 node to discover other Reactive 2 nodes connected to the same OLSR area so that any RREQ can be unicast to these Reactive 2 nodes.

3.2.5. REACTIVE 3 STATE

In this state the node has decided to work in proactive mode, but it knows no areas yet. It makes no sense to have an area with just one node. Due to this, this node continues working using the AODV features while it is sending the OLSR control messages searching for another router in the same situation, or an area to join. When the node working in the Reactive 3 state listens to the proactive control messages it will work in the Proactive 1 state, becoming part of the area (if N<Y). As we can see, the minimum number of nodes in an area is two.

If the node working in reactive 3 state listens to the proactive control messages of an area, it must check how many nodes there are in that area. If N<Y, then our node will join the area and will work in the Proactive 1 state. But if N≥Y, then the node must continue to search another area that it can join. The new node knows the number of N, because Penaguila introduces a field in the Hello messages with that information (see section 3.2.2.3.1).

3.2.6. PROACTIVE 2 STATE

The nodes working in this mode are the area border routers (ABRs), that is, the nodes that have connectivity with any router outside the area. All the nodes outside the area must understand the reactive routing protocol messages (RREQ and RREP), but also the area border routers must do so.

When an ABR receives a RREQ, it first determines whether it has a path to the requested node (can be a known AODV node, or an OLSR node that is in its routing table because it is in the same proactive area) or whether it has to forward the RREQ to other AODV nodes crossing the area. If it has a path to the destination node, it will send a RREP to the sender. If the destination in the RREQ is a pure OLSR node, the ABR will have to keep track of the destination sequence number on behalf of the OLSR node, to ensure correct operation of the AODV protocol. If the ABR does not have a path to the destination, it will have to rebroadcast the RREQ to other AODV nodes and unicast the RREQ to other ABRs, enabling it to traverse OLSR networks. When an ABR receives a RREP, it will create a forward route to the source of the RREP, and forward the RREP to the next hop of the reverse route. Similarly, the broadcasting of RERR messages will have to be modified. In AODV networks, RERR messages are unicast if there is only one predecessor or broadcast if there is more than one. However, if the node is an ABR, RERR messages will be unicast to every predecessor node that is an ABR.

An ABR receiving an AODV routing message (RREQ, RREP or RERR) that has to pass through the proactive area sets the flags to the correspondent value explained in the section 3.2.8. The exit ABRs will change the flags's value again. Then, these ABRs must broadcast the routing messages to all the routers with connectivity to them. In Figure 3.24 we can see an example:

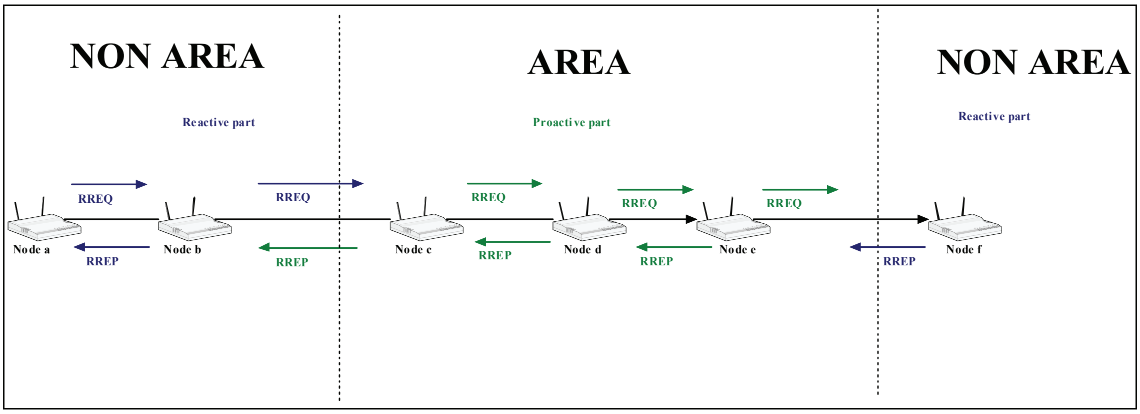


Figure 3.24: Route Discovery process throwing an area. Nodes c and e are ABRs. Node c understands the RREQ and it changes the flags of these message. Then, it sends the message to the Node e whom change the flags again. The same happens with the RREP but this time from Node e to Node c.

The same happens with the RREP: The destination answers with a RREP, and the entry area border router (Node e) forwards it to the same area border router which first sent the RREQ (Node c). Then, the exit area border router forwards it using the reverse path.

3.2.7. PROCESSING OF DATA PACKETS

For a data packet received for a Proactive 2 node via its Reactive 2 node which is connected to from a Reactive 1, Reactive 2 or Reactive 3 node, and the destination is not found, a RERR will be sent back to the sender. For a data packet received by a Reactive 2 node via its Proactive 2 node which is connected to an OLSR node, and the destination is not found, it will buffer the data packets and send a RREQ to initiate a route discovery process on behalf of the OLSR node. If no RREP is received after RREQ_RETRIES, then it will send an ICMP (Internet Control Message Protocol) Destination Unreachable message back to the OLSR node.

3.2.8. MESSAGE TYPES

When a message is sent through the networks, 4 different cases can be distinguished:

1. A proactive message through a proactive area: Proactive 1 messages.
2. A proactive message through a reactive zone: Proactive 2 messages.
3. A reactive message through a reactive zone: Reactive 1 messages.
4. A reactive message through a proactive area: Reactive 2 messages.

In all the messages sent, there are 2 bits (i.e., flags): P and N describing which kind of message it is:

00: Proactive 1 message.
01: Proactive 2 message.
10: Reactive 1 message.
11: Reactive 2 message.

Therefore, if P=0, then the packet is an OLSR packet. If P=1 the packet is an AODV packet.

If N=0, the packet is working in a region with nodes working in the same mode than the packet is. If N=1, the opposite happens.

Depending of which kind of message it is, the node knows how it has to process it. For example, since an OLSR node has no route entry to other nodes (AODV nodes or OLSR nodes separated by AODV networks) other than the OLSR nodes in its own network, data packets have to be routed between two Reactive 2 nodes. The original flags of the data packet ("00") will be replaced by the new flags ("01") by the source Reactive 2 node. When the data packet reaches the destination Reactive 2 node, the original flags of the data packet will be restored. The packet has the OLSR packet format as it was explained in the section 3.2.2.2, but when the intermediate Reactive 1 nodes see the flags set to "01" they know how to forward it.

Also, when a reactive message has to go through a proactive area, these flags are used to indicate to the nodes within the area that they have to process this packet as an AODV packet.

When the flags are set to "00" or "10", the working is like pure OLSR or pure AODV respectively.

To summarize, the Penaguila routing protocol defines 6 different states each one with a different behaviour. However, all the nodes have to know how to forward both kinds of packets, proactive and reactive. The way that the nodes have to know which kind of packet it is, is by using the flags described in this section.

3.2.9. A NODE CHANGES TO PROACTIVE 2 OR REACTIVE 2 STATE

When there is communication between a reactive node and a proactive one, both have to have each other in their routing tables. In the example of Figure 3.25, the Node d has in its routing table node r. It knows that it is working in reactive (Reactive 2 mode) mode and, besides it has to know the node r is working in proactive mode (Proactive 2 mode). The same happens to the node r: it knows it is working in proactive mode, but it needs to know that Node d is working in reactive mode. The key is that the only nodes within an area connected with the reactive routers are the area border routers. It is easy to define a mechanism to achieve that a node knows when it is a border router and when it is not.

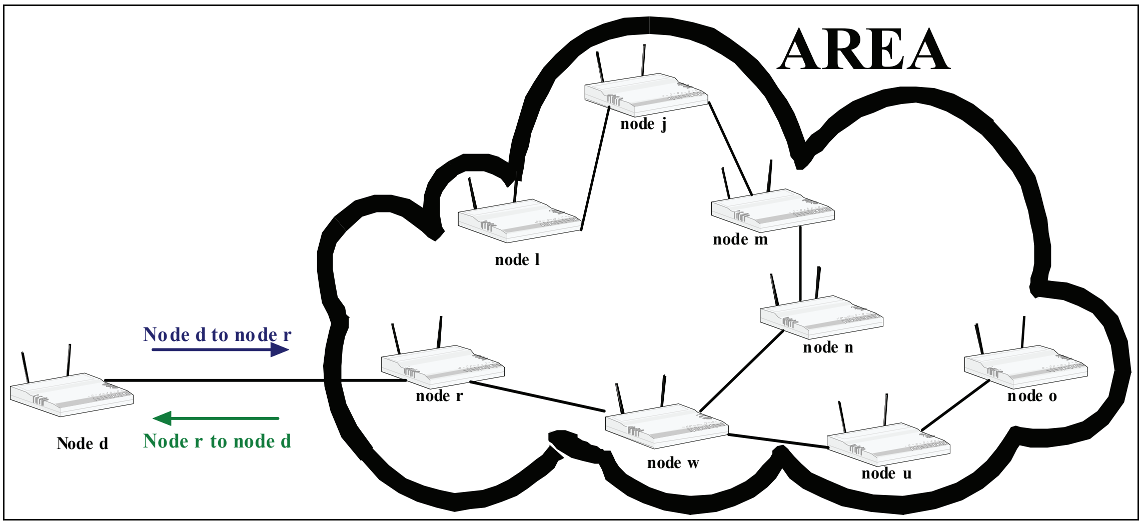


Figure 3.25: Communication between a reactive node and a proactive one. Node d is a R2 node. Node r is an ABR.

In the 'State Transition of a Mobile Node' of Figure 3.2, we can see that just as a node only can change to the reactive 2 state from the reactive 1 state (when there is connectivity with a node in the proactive area), also a node only can change to the proactive 2 state from the proactive 1 state (when there is connectivity with a reactive node). Both reactive 2 state and proactive 2 state nodes, work together doing the job of a "gateway" that communicates the reactive and the proactive areas. The objective of this chapter is to explain how a node knows if a node has to work in one of these two states.

Penaguila protocol introduces in all packets an additional bit (in addition to the flags described in the last section). If the last node sending the packet was outside the area, the bit=1. If the last node was in an area, the bit=0. Hence, we can have:

(a) The node receiving a packet sees the bit at 1. Then, there are two subcases:

(a1) The node receiving the packet has been working in reactive mode (reactive 1 state) until now. When this node sees the bit at 1, it knows that it must not change to the reactive 2 state. Then, the node will continue working as described in section 3.2.3. (using the reactive protocol features: Reactive 1 mode).

(a2) The router receiving the packet has been working in proactive mode (proactive 1 state) until now. Seeing that the packet comes from a node outside the proactive area it now knows that it is an area border router. Thus, it has to change to Proactive 2 state.

(b) The node receiving a packet sees the bit at 0. Then, there are two subcases:

(b1) The node receiving the packet has been working in proactive mode (proactive 1 state) until now. Then, the node will continue working as described in section 3.2.2 (using the proactive 1 state features).

(b2) The node receiving the packet has been working in reactive mode (reactive 1 state) until now. Then, it knows that it is a reactive router working with an area border router (hence, it has to change to Reactive 2 state).

Both Reactive 2 and Proactive 2 states understand every control message (AODV as OLSR control messages). Hence, there is no problem to know when they loose connectivity between them because both understand the mechanisms of AODV and OLSR routing protocols to discover the loss of connectivity.

Once this has been defined, the operation of the protocol is very simple. The reactive routers ignore the control packets of the proactive routing protocol, with the exception of those reactive routers that have connectivity with the area border routers. They have to use these packets to store in their routing tables how to reach the nodes inside the area. The proactive routers do not process the control packets of the reactive routing protocol with the exception of the area border routers. They have to use these packets to store in their routing tables how to reach the nodes outside the area.

3.2.10. IDENTIFICATION OF THE ABRs AND THE R2 NODES

The nodes working in any of the reactive states do not need to know which nodes are working in Reactive 2 state. The Reactive 2 state nodes are part of the reactive zone, and the other reactive nodes see them as normal reactive nodes. The communication between a Reactive 2 state and an area is transparent to the other reactive nodes.

However, the nodes working in any of the proactive states need to know which nodes are working in Proactive 2 and Reactive 2 state with connectivity to their area. That is because when a packet has to be sent outside the proactive area, it is necessary to know to which nodes the packet must be sent (that is, to the ABRs and R2 nodes).

All the Proactive 1 nodes know which are the ABRs and R2 nodes because in the Hello messages there are two lists, enumerating the neighbours that are proactive 2 nodes and the neighbours that are reactive 2 nodes. The "link code" field in the Hello message specifies what kind of neighbour is each node as was explained in the section 3.2.2.3.1.

In Penaguila routing protocol every node that is within an area has, in its routing table, the routers with connectivity with the area (all the Reactive 2 nodes connected to the area) even though they do not pertain to the area. Also, every node in the area can find in its routing table which nodes are the ABRs. For example, in the proactive area of Figure 3.26 the router u has the routing table illustrated in Figure 3.27 (where 1=true and 0=false):

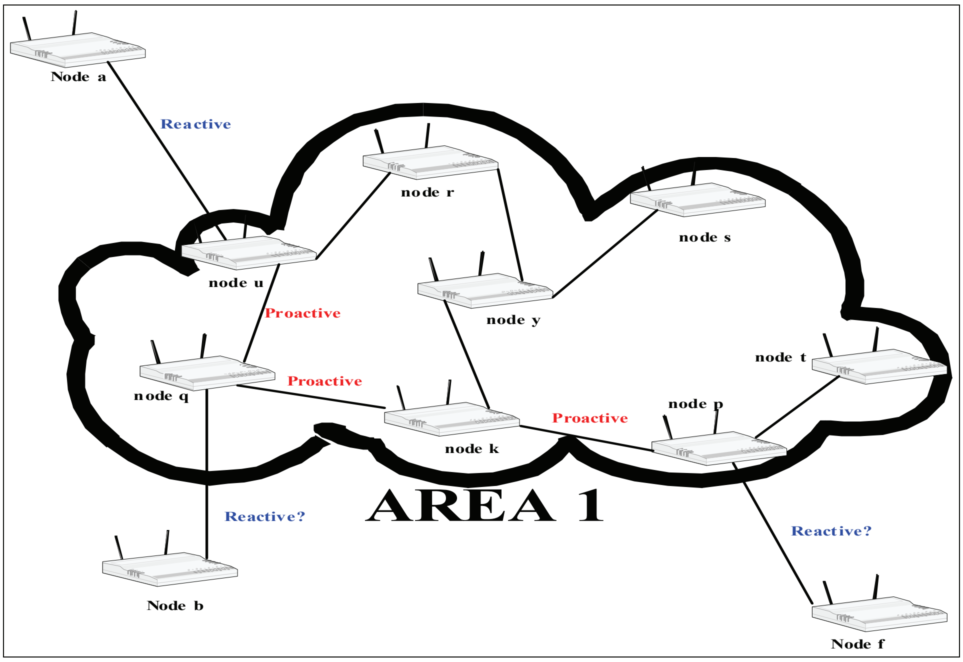


Figure 3.26: Example of a proactive area connected with Reactive 2 nodes. In the area 1 nodes u, q and p are ABRs. The nodes r, y, k, s and t are Proactive 1 nodes, and the nodes a, b and f are Reactive 2 nodes.

Destination	Next Hop	Hops	Destination into the area	Destination has any destination in its table outside the area
q	q	1	1	1
r	r	1	1	0
y	r	2	1	0
k	q	2	1	0
s	r	3	1	0
p	q	3	1	1
t	q	4	1	0
a	a	1	0	(no valid)
b	q	2	0	(no valid)
f	q	4	0	(no valid)

Figure 3.27: Node u routing table. Looking at its routing table, the node u knows that the destinations with the field ”Destination into the area” set to ’0’ are Reactive 2 nodes. Also, looking the field ”destination has any destination in its routing table outside the area” to ’1’ the node u knows that the nodes q and p are ABRs.

The routing table for the nodes within an area must have two fields called “Destination into the area” and “Destination has any destination in its table outside the area”. If the destination is not inside the area, the node knows that this destination is a Reactive 2 node. On the other hand, if the destination has any destination in its routing table outside the area, the node knows that this destination is an ABR.

3.3. OPERATION EXAMPLE

We can see the area as another “node” in the global network. In Figure 3.28 we can see an example of Penáguila Network where the Area 1 and Area 3 are shown in Figures 3.37 and 3.38 respectively:

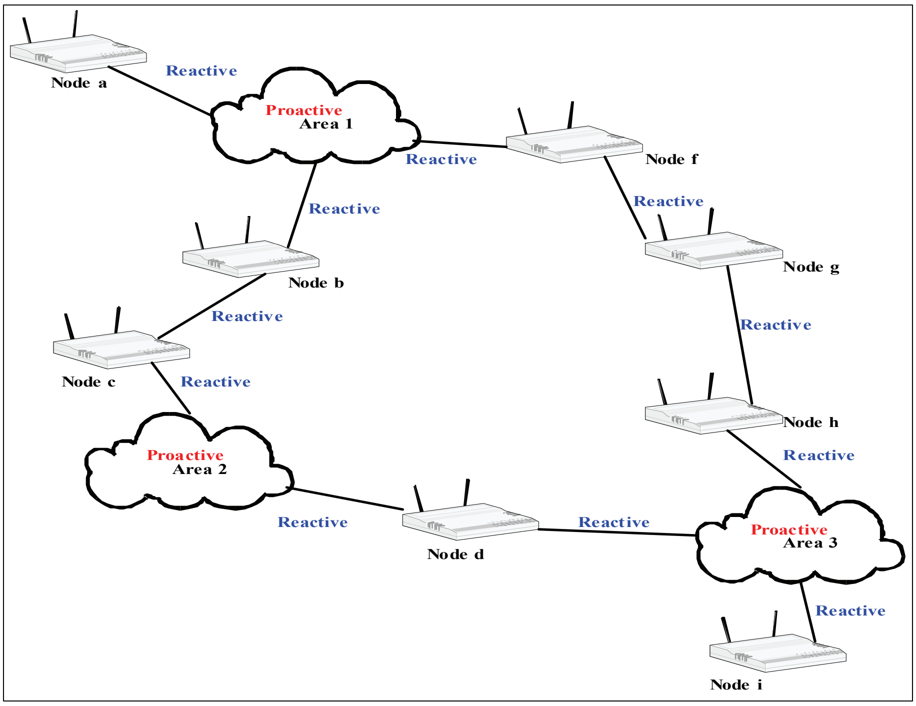


Figure 3.28: Example of Penaguila network. There are nodes working into areas and nodes working reactively. All the links are bidirectional and represent connectivity between nodes

We are going to study the case in which "node a" wants to send a packet to "node i" before the route is created. Nodes a, b, c , d, f, g, h and i are working in the reactive mode. Nodes a, b, c, f, h and i are in the reactive 2 state, and Node g in reactive 1 state. Since the communication begins in Node a, the reactive protocol must discover the path using the Route Discovery process.

The RREQ is travelling through the network as was described before in the chapter 3.2.3. But now, we not only have nodes, but also areas. Within these areas, all the routers know how to reach a destination. In case that the RREQ has to reach a destination node crossing an area, the internal routers of the area have to forward the RREQ to the area border routers. If the case of Figures 3.28, 3.29, 3.30, 3.31, 3.32, 3.33, 3.34 and 3.35, we are calling brij to the border router of 'area i', called 'node j'. For example, the inside of 'Area 1' is shown in Figure 3.37, and we see the area border routers called 'node q' and 'node p'. Hence, we will call them br1a and br1p respectively.

As shown in Figure 3.29, Node a sends the RREQ to the next node that is inside the Area 1. In Figure 3.30 we can see that the RREQ follows two different ways. To simplify we are going to explain only the right path. In Figure 3.37 Area 1 is illustrated and we can see that nodes u, p and q are proactive 2 state routers (ABRs), while nodes r, y, k, s and t are proactive 1 states routers. The RREQ arrives to u and it forwards it to the other ABRs, that is to nodes q and p. Node p sends the RREQ to the Node f (Figure 3.30), and this one to Node g (Figure 3.31). Node g to the Node h (Figure 3.32), and this one to the node j (Figure 3.33) that is within the Area 3 (showed in the Figure 3.38). Node j send the RREQ to the others ABRs, nodes r and w. The node w finally finds the destination Node i (Figure 3.34). Thus, it is possible to answer with a RREP to the source (Node a) as shown in Figure 3.35.

Another issue to have into account is, in the case of Figures 3.33 and 3.34, when two RREQ with the same Broadcast ID and the same source IP address enter to the 'Area 3' but through different border routers (if both enter by the same border router there is no problem, because AODV mechanism achieves that only one RREQ is considered). To understand the problem we need to see the inside of Area 3, shown in Figure 3.38.

When the RREQ arrives to 'node j' from 'Node h', 'node j' stores 'Node h' as a reverse path (Figure 3.33). The same happens with 'node r' to 'Node d'. But only one of them arrives first to 'node w'. In this case, we are going to suppose that the first RREQ is the one arriving via 'node j' and 'Node h'. When the second RREQ arrives through the path 'Node d' and 'node r', then 'node w' ignores it because it detects that is a RREQ with the same source node IP address and the same broadcast ID than before.

When the 'Node i' answers with a RREP, the 'node w' has as a reverse path the 'node j'. Hence, it is through that way where the RREP is going to travel.

Therefore, when the first RREQ reaches the destination ('node i'), this node answers with the RREP as has been explained in the chapter 3.2.3 following the reverse path. Figure 3.36 shows the reverse path used by the RREP.

Each ABR performs as an AODV node. Hence, there are no loop problems or inconsistencies. The issue is that all the area border routers know about this concrete RREQ. Hence, the RREQ cannot enter again inside the area for any other place and create loops. Both entrance area border routers (nodes j and r) establish a reverse path to the source node (Figures 3.34, 3.35 and 3.36) but the exit area border router (node w) will select only one of them to send the RREP to the Node a (Figure 3.36). This one will be the one from which it first receives the RREQ.

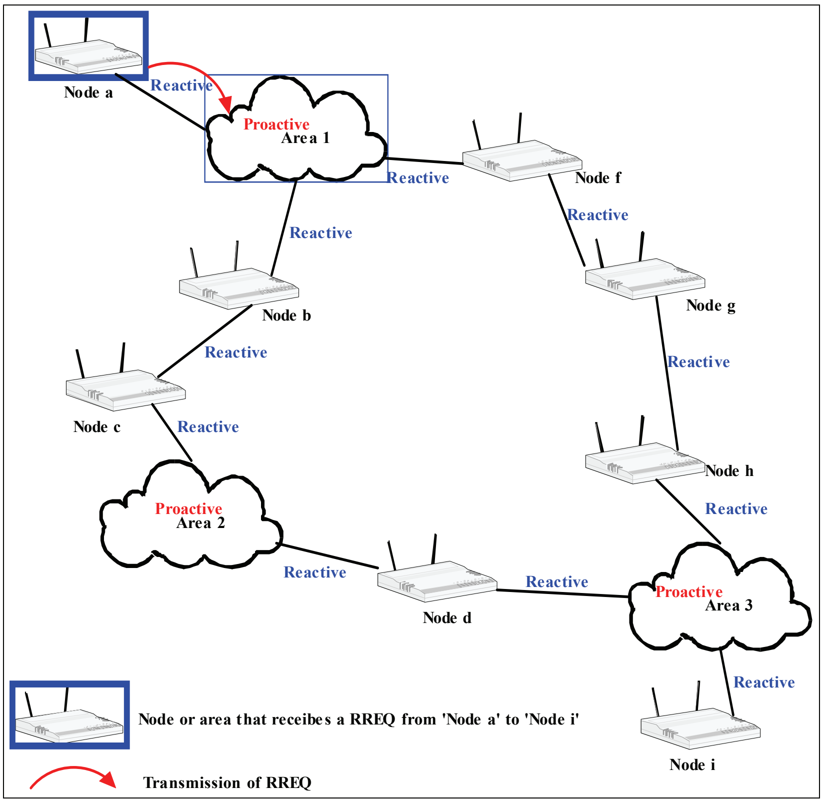


Figure 3.29: Operation example (1). The ‘node a’ sends a RREQ to the Area 1

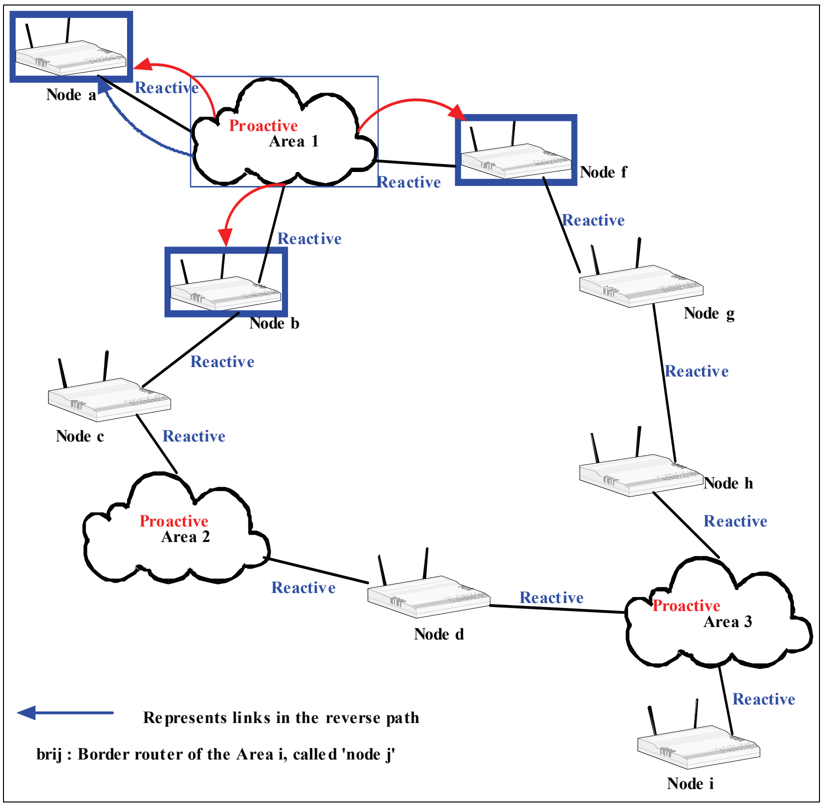


Figure 3.30: Operation example (2). The area 1 has two exits. Hence, it propagates the RREQ by both. The exit routers from area 1 propagate the RREQ to all the routers that have connectivity with them (in this case, node border router of Area 1 called 'node q' (br1q) to node b, and node border router of the Area 1 called 'node p' (br1p) to node f).

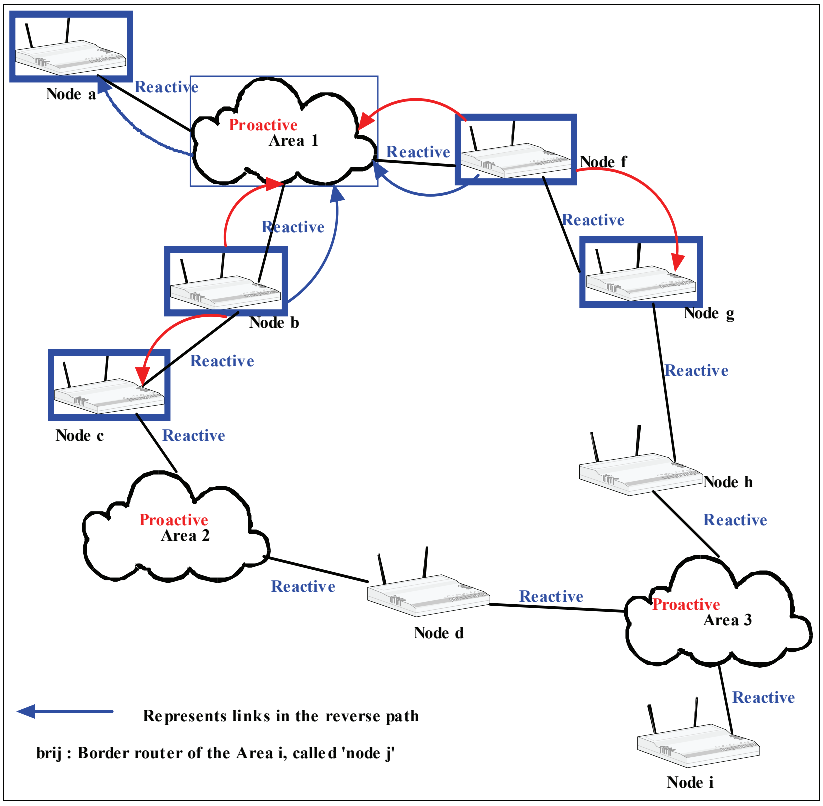


Figure 3.31: Operation example (3). The nodes b and f propagate the RREQ to nodes c and g respectively.

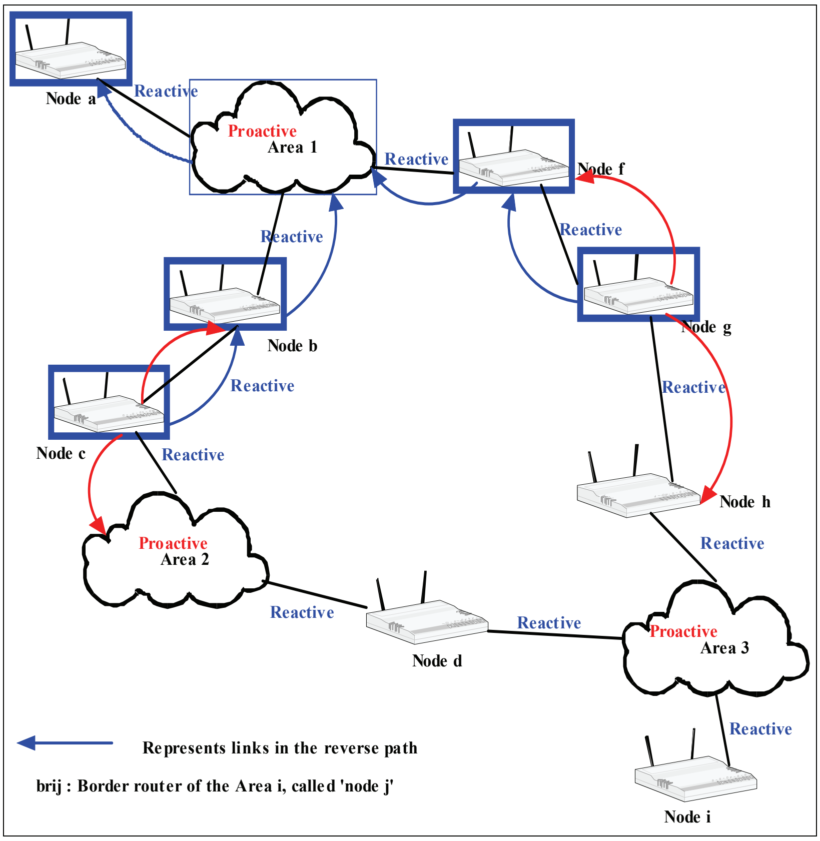


Figure 3.32: Operation example (4). The nodes c and g propagates the RREQ to the area 2 and node h respectively.

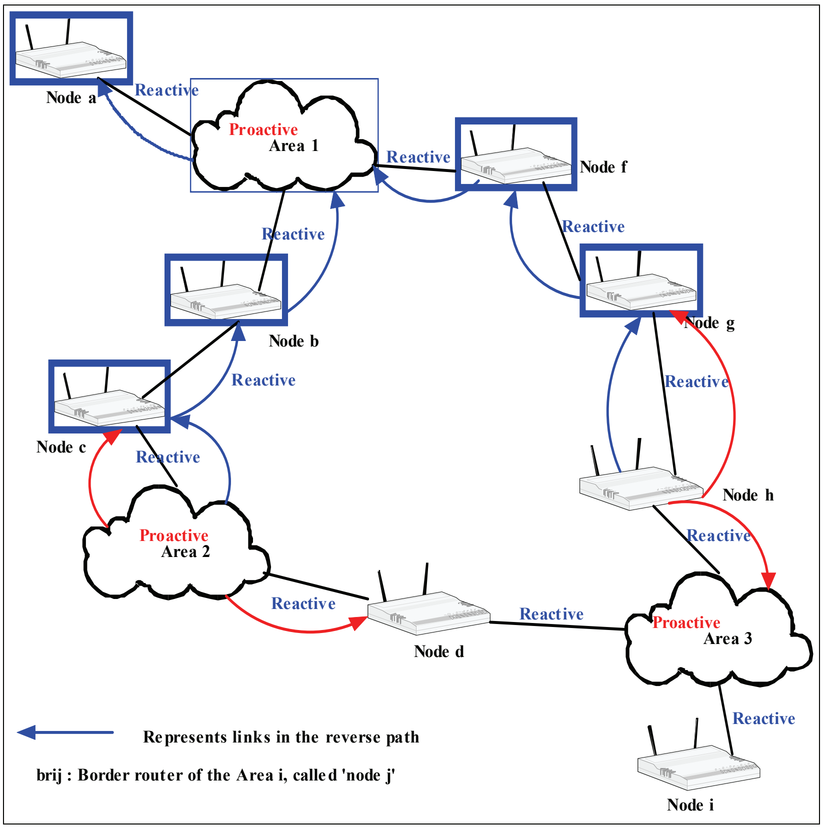


Figure 3.33: Operation example (5). The RREQ reaches node d and Area 3.

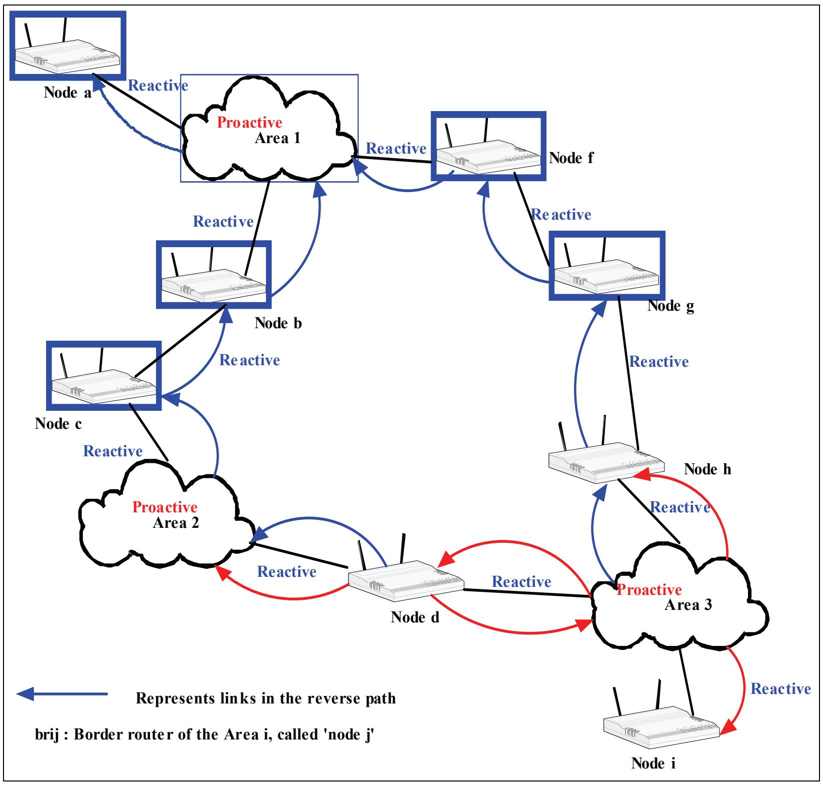


Figure 3.34: Operation example (6). The RREQ has arrived to Area 3 before by the right path, but now it arrives to Area 3 another RREQ from the same source and with the same broadcast ID. As the second RREQ enters by other border router, it will have two reverse paths.

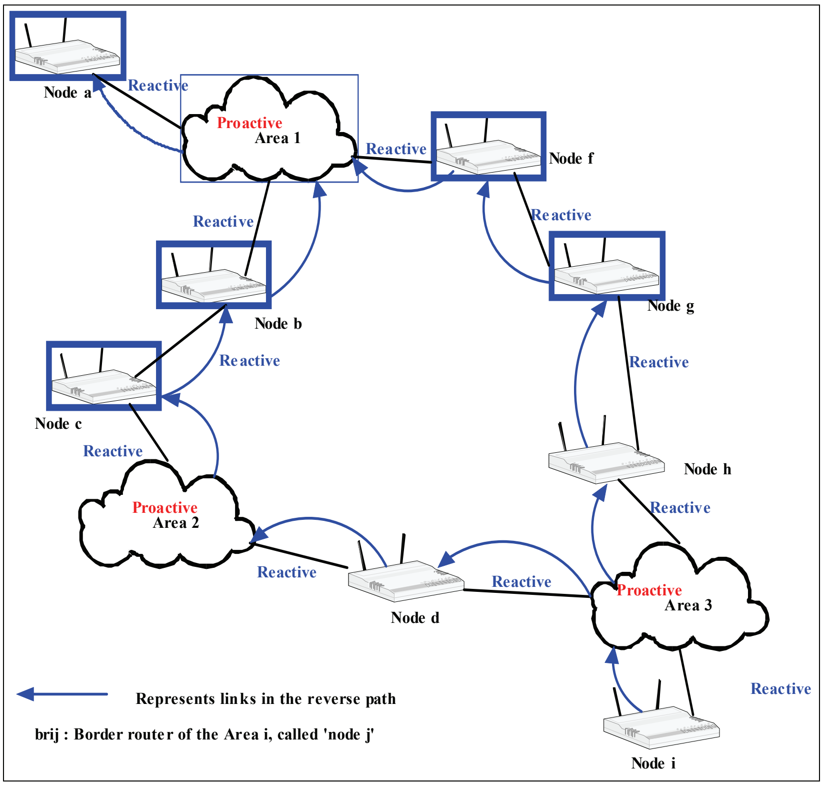


Figure 3.35: Operation example (7). The node i does not broadcast the RREQ because is the destination node in the Route Discovery process. The down border router of the Area 3 does not broadcast again the RREQ because it knows that it already did it.

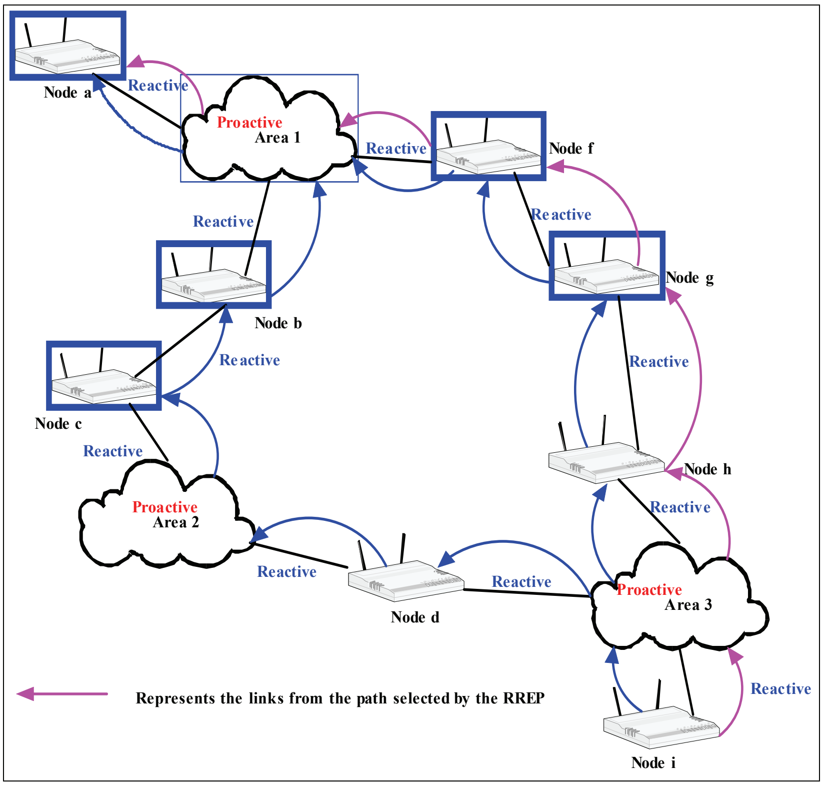


Figure 3.36: Operation example (8). The RREQ arriving for the right path was first. Hence, the reverse path is established in by this way.

As we can see, the working is the same than with AODV. AODV is very good protocol when the topology changes quickly. Here, the only nodes who use this reactive protocol are the ones who don't belong to any area, in other words, the nodes who move frequently. When the traffic is high and the changes in the topology are not very frequents, is better to use a proactive protocol. With Penaguila protocol the nodes that are moving slowly and have a lot of traffic work with routing tables.

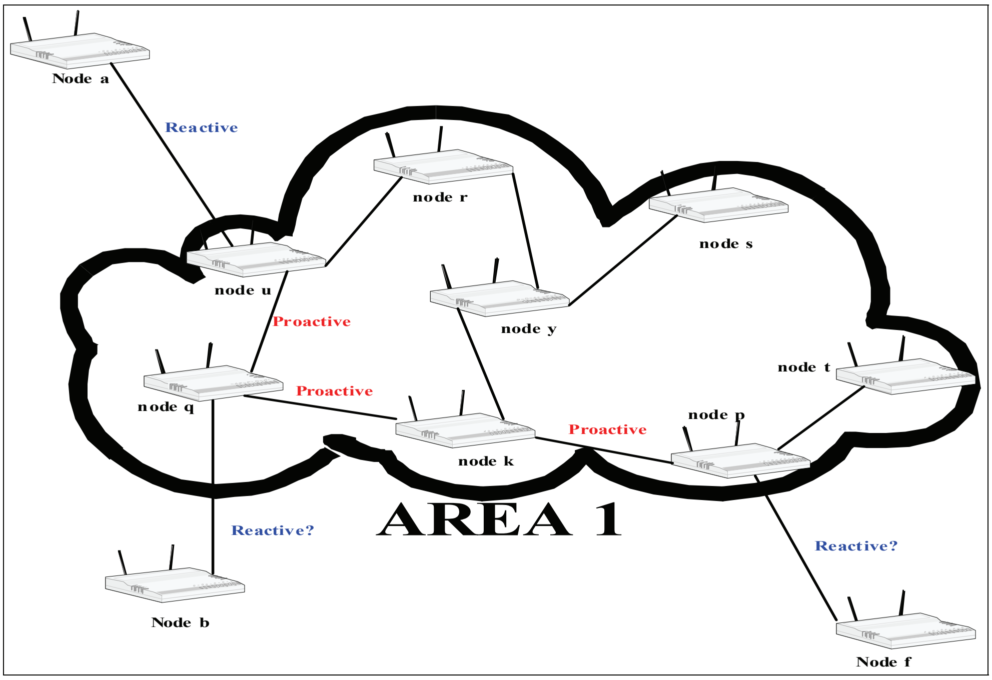


Figure 3.37: Internal structure of the area 1. The lines represents connectivity.

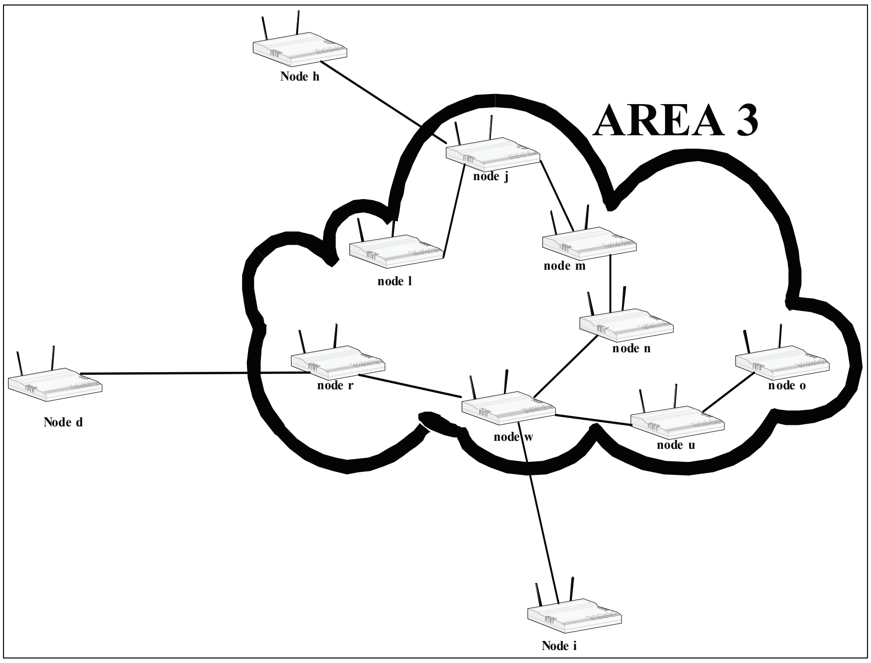


Figure 3.38: The Area 3 internal structure

4. EVALUATION

In this chapter firstly there is an introduction about how the evaluation of a routing protocol for MANETs is done, and why it was not possible to do that here. Secondly, there is a theoretical comparison between Penaguila and some representative protocols (AODV, DSR, OLSR and ZRP). And finally, there is a quantitative study of these existing protocols using results of other documents and references, followed by a qualitative study discussing these results and trying, with this reasoning, to guess how the Penaguila routing protocol could perform.

4.1. INTRODUCTION

In this chapter it is going to be discussed the new protocol in comparison with others already existing. It is important to know if in deed, to mix both AODV and OLSR improves each one separately. Besides AODV and OLSR, DSR and ZRP are going to be studied hereafter. DSR is a very important and typical routing protocol for MANETs, and ZRP is hybrid as well as Penaguila.

When comparing a new routing protocol with the current protocols making a simulation study is very usual. There are several different simulation programs that can be used for the simulation, like for example: ns2 [NS_2], GloMoSim [GloMoSim], QualNet [QualNet] and OPNET [OPNET]. The most commonly used software of the four is the ns2 [NS-2]. However, this is a Master Thesis of five months, and the largest part of it has been to study the theory in order to firstly understand how the different routing protocols for MANETS operate under different settings, and secondly to define a new protocol. To programme a routing protocol of these characteristics may involve writing thousands of code lines. Therefore, it was impossible to do this kind of evaluation in so short time.

However, in this chapter we will discuss in as much detail as possible how the Penaguila protocol performs under different network settings, and how good it is in comparison with other protocols.

Since it has been impossible to simulate the Penaguila routing protocol, there is no quantitative study of it in this Master Thesis. However, for AODV, DSR, OLSR and ZRP there are a lot of quantitative studies. Some results of these reports are included in the following to understand and demonstrate the performance of these protocols. Understanding why they perform in such a way and explaining qualitatively as much as possible all the features, it is possible intuitively to predict how good the Penaguila routing protocol can be.

4.2. THEORETICAL COMPARISON

4.2.1 PARAMETERS

To compare the different routing protocols for MANETS there are a group of important parameters to take into account. These parameters are common in the quantitative analysis but it is necessary to know and to understand them to do a theoretical study. In the next points some of the most representatives are explained :

- Throughput: packets delivered per second (TCP traffic only). Examining throughput, especially when it is considered relative to different network scenarios, helps to determine how well the routing protocols permit applications to optimize the use of the available bandwidth.
- Packet delivery ratio: packets successfully delivered to destinations over total number of packets sent. Packet delivery ratio is calculated by dividing the number of packets received by the destination by the number of packets originated by the application layer of the source. It specifies the packet loss rate, which limits the maximum throughput of the network. The better the delivery ratio, the more complete and correct is the routing protocol.
- Control packet overhead: The routing overhead describes how many routing packets for route discovery and route maintenance need to be sent in order to propagate the data packets. It is an important measure for the scalability of a protocol. It, for instance determines if a protocol will function in congested or low-bandwidth situations, or how much node battery power it consumes. If a

protocol requires sending many routing packets, it will most likely cause congestion, collision and data delay in larger networks.

- Control byte overhead: The total number of control bytes used in the control packets.
- Delay: End-to-end packet delay, from source to destination. The end-to-end delay measures the delay a packet suffers after leaving the sender and then arriving at the receiver application. This includes delays due to route discovery, queuing at Internet protocol (IP) and medium access control (MAC) layers, and propagation in the channel.
- Hop Count: It represents the number of hops that a packet has taken before it has been correctly delivered.

4.2.2 COMPARING THE PROTOCOLS

In this section the five protocols described are compared. In section 4.2.2.1 a comparison overview is provided, and in sections 4.2.2.2 through 4.2.2.5 the protocols are compared with respect to resource usage, mobility, route discovery delay, and scalability, respectively.

4.2.2.1. OVERWIEW

As a proactive routing protocol, OLSR inserts high control traffic overhead on the network. To maintain and to update the routing table for the entire network it needs a lot of communication between the nodes, as well as periodic updates flooding the network. The use of MPR's reduces this control traffic overhead, but for small networks the improvement is minimal. The traffic overhead also consumes bandwidth.

The behaviour of reactive protocols AODV and DSR is different. The main part of control traffic is emitted during route discovery. Therefore, a lot of the resource and bandwidth consumption is related to actual data traffic.

ZRP and Penaguila are hybrid routing protocols. Both combine proactive and reactive approaches to achieve a better performance. Within the proactive areas, the behaviour is proactive. Hence, inside an area each node maintains and stores the information of the entire area. To communicate these areas a reactive protocol is used. Hence, route discovery will be needed. The advantage of these protocols is that they have significantly reduced the amount of communication overhead when compared to pure proactive protocols. They also have reduced the delays associated with pure reactive protocols such as DSR or AODV, by allowing routes to be discovered faster. This is because the nodes only store and maintain routing information of the nodes that are in the same proactive area. Also, the route discovery process is faster when looking for a node outside the same proactive zone, since a border router that has proactively the path to the destination can answer with a RREP on behalf of the destination.

The disadvantage of these protocols is that for large values of routing zone they can behave like a pure proactive protocol, while for small values they behave like a reactive protocol.

Table 4.1 shows the different characteristics of the protocols under study in this chapter.

Protocol Property	OLSR	AODV	DSR	ZRP	Penaguila
Routing structure	Flat	Flat	Flat	Flat	Hierarchical
Loop free	Yes	Yes	Yes	Yes	Yes
Multiple routes	No	No	Yes	No	No
Distributed	Yes	Yes	Yes	Yes	No
Reactive	No	Yes	Yes	Hibryd	Hibryd
Unidirectional link support	No	No	Yes	No	No
QoS support	Yes	No	No	No	Only inside the proactive areas
Multicast	Possible	Yes	No	No	Possible inside the areas
Security	No	No	No	No	No

Power efficiency	No	No	No	No	No
Periodic broadcasts	Yes	Yes	No	Yes	Yes

Table 4.1: Comparison between the protocols

4.2.2.2. RESOURCE USAGE

The storage complexity of the OLSR protocol is related to the number of nodes in the network. With OLSR all the nodes need to maintain information about the entire network constantly. OLSR must keep the topology information in the topology set, the MPR information in the MPR selector set and also update the state information about the links and neighbours. It also maintains information about routes that may never be used. Besides, the control overhead increases the necessary processing in each node. For all that, the energy consumption is greater than in the reactive routing protocols.

The storage complexity of AODV and DSR is related to the number of communication pairs. Both only have to store information about active routes. That simplifies the storage complexity and reduces energy consumption in regard to the proactive protocols. Also, the control overhead is less than in OLSR since little or no routing information is maintained.

There is no periodic maintainability of the routes in DSR. In the OLSR it is done by TC messages and in the AODV by periodic Hello messages. OLSR tries to minimize this traffic with the usage of MPR. Only these nodes broadcast the TC messages. In addition to this, OLSR also uses Hello messages to maintain the neighbour's status. On the other hand with AODV only the nodes that participate in the communication periodically send Hello messages with the hop limitation to one hop. Besides, the size of the Hello messages in AODV is smaller than those used in OLSR.

OLSR needs more bandwidth and energy resources that AODV and DSR.

The byte overhead of DRS deserves a special mention. Since DSR is a source routing protocol, the path from the source to the destination is attached in the header of every packet. The larger the number of hops is, the higher the byte overhead is. If the byte overhead is high, the bandwidth efficiency is smaller and the processing quantity is greater in every node.

ZRP is supposed to reduce the table maintenance inherent to proactive protocols, but realistically has a higher overhead than proactive and reactive protocols. If the zones overlap greatly, there are redundant Route Requests flooding the network. Besides, the intermediate nodes have high stress when there is a link breakage.

Penaguila routing protocol limits the size of the proactive areas, working inside them almost in the same way than OLSR. Therefore, the nodes working within an area have a routing table with a limited size because they do not store in their table the destinations outside the area. Therefore, the nodes working in Proactive 1 state in Penaguila do not have the problem of a large number of nodes to store in their routing table like in the pure OLSR. The nodes working in Reactive 1 mode have almost the same behaviour than AODV. Hence, the resource usage is similar. However, as a drawback, the nodes working as ABR or in Reactive 2 mode have to process all the control messages (reactive and proactive) and to store destinations within and outside the proactive areas. This means that for a lot of nodes Penaguila achieves a small resource usage, but for the ABR and R2 nodes the stress is high.

4.2.2.3. MOBILITY

Each routing protocol has different strengths and weaknesses when there is node mobility in MANETs. The dynamic topology in MANETs causes path breaks. When this happens, the routing protocol needs to find new routes. OLSR periodically updates the topology information, so the new routes are calculated immediately when a path breakage is reported. AODV and DSR are reactive protocols, so this immediate new route calculation is not done. A route discovery must be initiated. When the network traffic is distributed, OLSR has less overhead than both reactive protocols due to having found the routes proactively. On the other hand, AODV and DSR first have to discover a route before the information can be transmitted. Therefore, in case the network has sporadic traffic, they have great control overhead per packet. Otherwise, when the traffic has a long duration (i.e., the traffic is more or less static), they may perform better, since the amount of control overhead per packet decreases.

In Penaguila routing protocol, we can have different cases:

A node moves inside an area: Same case than in OLSR. All the nodes within the area have to update their routing tables. The nodes outside the area do not need to know anything about that change. Only the R2 and ABR must.

A node moves from an area to outside the area: All the nodes within the area have to update their routing tables. For the nodes in the reactive zone, there is a new node that can be reached by means of route discovery if necessary.

A node moves in the reactive zone: Same case than in AODV. The nodes in the reactive zone that want to communicate with this node have to begin a new route discovery process. The nodes within the proactive areas do not need to update their routing tables. Only the R2 and ABR must initiate a route discovery process if necessary.

A node moves from reactive zone to a proactive area: All the nodes within the area have to update their routing tables. For the nodes in the reactive zone, if they want to reach that node, they need to establish communication with the correspondent ABR and R2 node.

An ABR moves: All the nodes within the area have to update their routing tables. If, due to that movement its correspondent R2 node looses connectivity with the area, the R2 node must change to pure reactive mode. For communications between an area and the exterior, new route discoveries will be needed.

A R2 node moves: All the nodes within the area have to update their routing tables. For communications between an area and the exterior, new route discoveries will be needed. If, due to that movement its correspondent ABR looses connectivity with the reactive region, the ABR must change to pure proactive mode.

The critical point is the mobility of an R2 node or an ABR, because they communicate an area with the outside and a lot of data packets can be lost. When there is a change in the topology in the proactive area, the nodes within the area in the next periodic update will change their routing table. When the change is outside the area it is the same case than in AODV.

As it happens with Penaguila, in ZRP the effect of the node mobility depends in great manner on the size of the proactive regions. In ZRP each node stores information of the nodes at distance smaller or equal that the zone radius and determines the border routers. When there is a topology change, all the nodes involved have to rebuild its routing table and decide their new border routers using IARP and BRP. The difference with the Penaguila protocol, is that Penaguila nodes take into account the mobility speed when deciding in which mode to work. Besides, unlike ZRP, Penaguila proactive zones do not overlap. Hence, the node mobility can only affect one area while in ZRP just one node position change can affect multiple areas.

4.2.2.4. ROUTE DISCOVERY DELAY

In OLSR a node which wants to find a route to a destination, only has to look at its routing table. In AODV and DSR the node needs to initialize a route discovery process unless a valid route is stored. Since looking up the routing table takes less time than flooding the network, the OLSR performs better in delay sensitive networks.

With Penaguila a reactive node that needs to communicate with a reactive node initiates a route discovery process from the source to the destination. If the destination is inside the area, the route discovery process arrives until the ABR which knows how to reach the destination. When a proactive node needs to communicate with a reactive node it initiates a route discovery process from its correspondent R2 node (gateway) to the destination. If the destination is inside the area, the origin only needs to look up its routing table for the destination. Therefore, in the pure reactive or pure proactive cases the performance is the same that in AODV and OLSR respectively. In the hybrid case, the delay is worse than in pure proactive, but better than in pure reactive.

ZRP theoretically reduces route determination delay inherent to reactive protocols but still performs worse that the proactive protocols as OLSR. This parameter can change depending of the zone radius.

4.2.2.5. SCALABILITY

AODV and DSR protocols perform better in networks with static traffic. On the other hand, OLSR has advantage in networks with high density and highly sporadic traffic. But their scalability is limited when the network size increases. In the case of the AODV and DSR protocols there is a huge flooding of packets to search the routes. In the case of the OLSR protocol the routing table size grows nonlinearly and the control messages can block the actual data packets.

In ZRP the critic parameter since the point of view of the scalability is the node density. That is because if zones greatly overlap, redundant Route Request messages are flooded through the network.

Penaguila protocol presents a better scalability than reactive protocols, because the areas cut down the distances in the route discovery process. Also, it has better scalability than OLSR since the proactive nodes only have to store and maintain routes within their area, that is of a limited size. Besides, the use of R2 nodes and ABR as gateways introduces a hierarchy of two levels that allows them to increase the scalability.

4.3. QUANTITATIVE STUDY OF THE EXISTING PROTOCOLS

As said before, in this Master Thesis there is no simulation of the Penaguila routing protocol. However, it is possible to find a lot of papers studying by means of simulation the performance of AODV, DSR, OLSR and ZRP. In the following, these protocols are going to be analyzed using graphs with the parameters explained before, taken of some papers.

To make a performance study it is necessary to define a scenario which is going to be object of the analysis. As it has been said in this master thesis, each routing protocol for MANETs performs better or worse depending on the environment (number of nodes, traffic, nodes velocity, etc.).

In this chapter results of the [Perf_MIL04] paper (quantitative) are used to discuss (qualitatively) the characteristics of each protocol. This study was made using QualNet. The control values for parameters in experimental groups (or the scenarios defined for the next simulations) are:

Parameters/ Group	**Size**	**Density**	**Hops**	**Load**	**Mobility**	**Sources**
Size (nodes)	Varies	50	Varies	50	50	50
Density (m/nodes)	253	Varies	253	253	253	253
Max. Hops	10	10	Varies	10	10	10
Load (bytes/s)	1460/src	23360	17520	Varies	23360	17520
Mobility (m/s)	0	0	0	0	Varies	0
Sources	1/3' rd	16	12	16	16	varies

Table 4.2: Control Values for parameters for each experimental group

In all the experiments of [Perf_MIL04] Constant Bit Rate (CBR) application traffic was used with UDP as the transport layer. At the MAC layer, the IEEE 802.11 DCF (Distributed Coordination Function) protocol was utilized, with the IEEE 802.11b radio device with a maximum data rate of 2 Mbps. The radio range was approximately 375.

Each experiment occurred within a square terrain dimension. Node placement within the topology is always random and uniform. The node density was chosen to be

253 meters square per node, with the exception of the node density group of experiments for which this value varies. The control values for all parameters in all experimental groups are summarized in table 1.

The set of CBR applications for each set of experiments were chosen by randomly selecting the set of sources and destinations from the available nodes using 3 different random seeds. Each point of the graphs of the results is the result of 9 separate simulation runs.

The main part of this study is based on [Perf_MIL04], but in those cases that the results of [Perf_MIL04] are not clear enough or when it is necessary to use some additional information, there are extra results from other papers (more references) in order to study in depth the performance of the routing protocols.

4.3.1. ROUTING PROTOCOL SCALABILITY. NETWORK SIZE

The next table summarizes the network size experiment parameter settings:

Network Sizes	10, 100, 225, 529, and 1024 nodes
Node Placement	Uniform density (avg. 1 node/ 253 m^2), random placement
Mobility	None
Traffic	1/3rd of the network randomly selected sources with randomly selected destinations
Simulation Time	Proportional to the number of nodes, varied from 160 seconds to 1850 seconds
Stabilized Application Load	Proportional to the number of nodes, varied from 4380 bytes/sec

Table 4.3: Network size experiment parameter settings

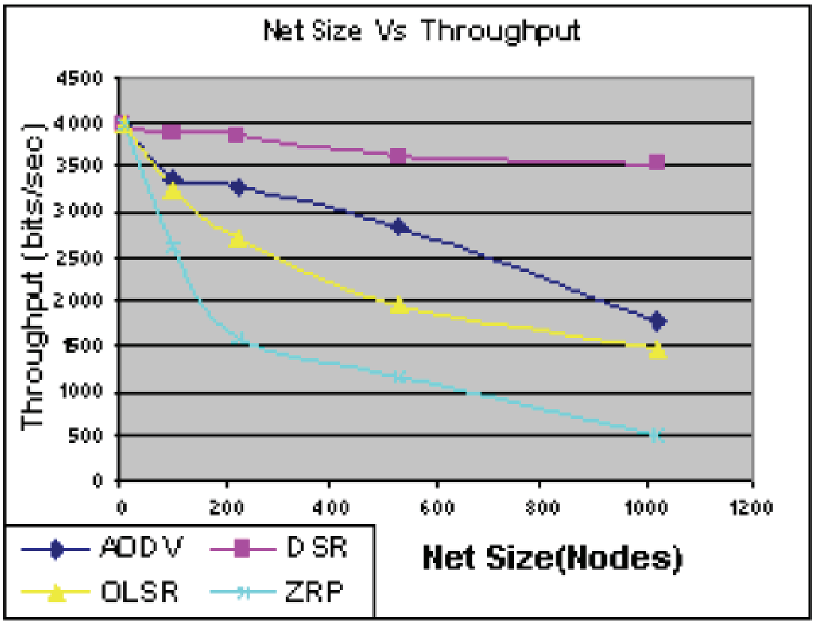


Figure 4.1: Throughput for Network Size

The network size vs. throughput graph in Figure 4.1 plots the per-node average of application level observations of bps data received.

According with these results, DSR is the best routing protocol when the network grows with this particular configuration.

OLSR and AODV perform similar in the range of 0-100 nodes, but when the number of nodes is greater, AODV performs better. One of the possible problems of OLSR is that as a link state protocol, if it is unable to converge, there are large disconnects in the known network topology and many packets are dropped due to the lack of sufficient routing information. On the other hand, AODV is supposed to perform as well as DSR, but here AODV shows a steeper decline than DSR. This behaviour is attributed in [Perf_MIL04] by one or more of the optimization differences between both protocols.

Finally, ZRP which was implemented in this study with the default protocol designer's recommendations for the timer and zone size variables performs much worse.

4.3.2. NODE DENSITY

The next table summarizes the node density experiment parameter settings:

Network Size	50 nodes
Node placements	Grid placement: • Sparse: 1 node/347 m^2 • Moderate: 1 node/185 m^2 • Dense: 1 node/132 m^2
Mobility	None
Traffic	1/3rd of the network randomly selected sources with randomly selected destinations
Simulation Time	225 seconds
Stabilized Application Load	23360 bytes/sec

Table 4.4: Node density experiment parameter settings

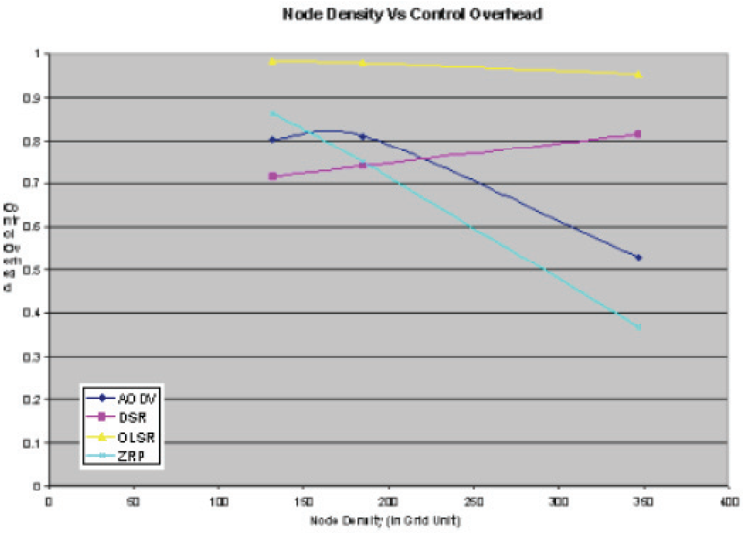


Figure 4.2: Node Density Vs Control Overhead

In Figure 4.2, the Control Overhead curve for the Node Density experiments is shown. The control overhead measurements are normalized. The horizontal axis represents the distance between neighbouring nodes in the grid.

The sparse networks have higher paths lengths. Thus, in these networks there are more rebroadcasts of route requests, and more route reply packets. For that reason DSR increases its control overhead when the density is smaller. However, AODV begins with a high overload when the node density is high, but uses fewer control packets as the density is smaller.

ZRP performs similar to AODV. When the density is high, it performs better and this is because the original route acquisition process depends on neighbouring nodes overhearing and rebroadcasting route requests, and if router requests are lost, the entire process stalls. The hidden terminal problem can contribute to route request losses, and is more prevalent in sparse networks. These protocols have difficulty dealing with a network with few neighbours.

On the other hand, OLSR as a proactive protocol has an almost uniform control overhead. It trends downwards with sparse networks because there are less links to report. But since there are fewer links, route convergence takes longer, which was seen in the latency graphs for this experiment set.

4.3.3. NUMBER OF HOPS

The following table (table 4.5) summarizes the number of hops experiment parameter settings:

Network Size	121 nodes
Node Placements	Grid placement: 1 node/252 m^2
Mobility	None
Traffic	Approx 1/4^{rth} nodes randomly selected as sources with randomly selected

	destinations at a given hop count.
Simulation Time	225 seconds
Stabilized Application Load	17520 bytes/sec (802.11b)

Table 4.5: Number of hops experiment parameter settings

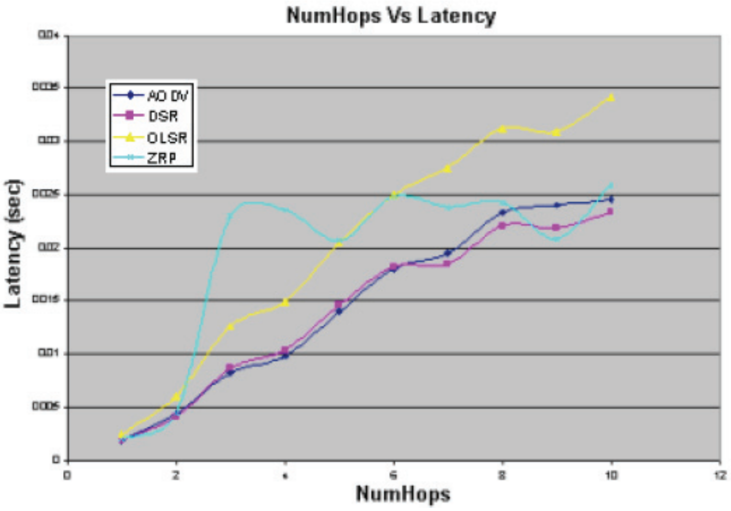


Figure 4.3: Number of Hops Vs Latency

The strangest result is to see that the latency for OLSR has the highest values from 1 to 10 hops, and generally the highest slope. For OLSR to lose its innate advantage in latency, network route convergence would have to be slower than route acquisition, and given the high control overhead data that was collected for this experiment set, it is easy to see that this is the case. However, under normal circumstances the OLSR is supposed to be the best of the analyzed protocols since the point of view of the latency.

For ZRP at the 1 and 2 hop has better latency than OLSR. This is because the proactive zone of interest is much smaller. At 3 hops and beyond, this result is indicative of the interzone routing, and it shows a fairly flat graph from 3 to 10 hops, with some oscillation caused by random number seeds not being completely filtered out.

AODV and DSR perform similar. In this experiment both have lower average latency than OLSR, but as said before, this is not normal.

4.3.4. MOBILITY

The following table (table 4.6) summarizes the mobility experiment parameter settings:

Network Size	50 nodes
Node Placements	Uniform density (avg. 1 node/253 m^2), random placement
Mobility	Random waypoint mobility, constant speed of 2 m/s, 9 m/s, 16 m/s, 20 m/s, 30 m/s with a 30 second pause when it reaches each randomly selected destination before choosing the next one.
Traffic	1/3rd nodes randomly selected as sources with randomly selected destinations at a given distance in hop count from the source
Simulation Time	225 seconds
Stabilized Application Load	17520 bytes/sec

Table 4.6: Mobility experiment parameter settings

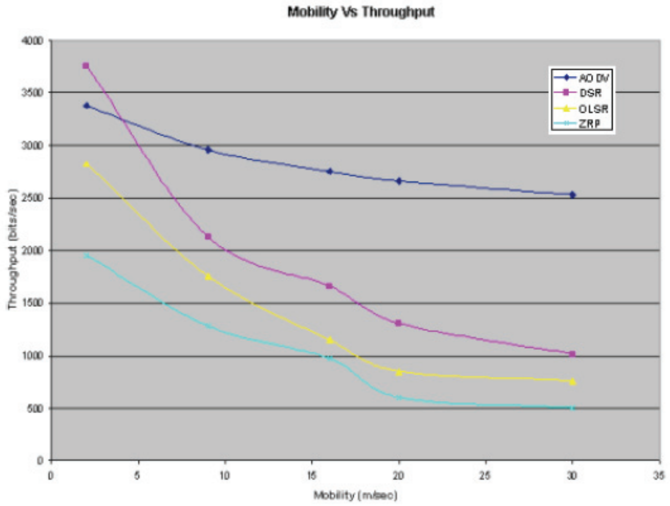


Figure 4.4: Mobility Vs Throughput

Figure 4.4 represents the mobility versus throughput data that we collected for this experiment set.

AODV is the best here. DSR starts out with higher throughput in the lowest mobility case, but DSR optimizations seem less able to handle high mobility, but it still manages a second place finish. OLSR is the third place finisher. OLSR is somewhat less scalable than DSR, but follows a roughly similar curve of decline. ZRP is the worst in this roundup.

4.3.5. NUMBER OF SOURCES AND DESTINATIONS

In [Perf_MIL04] the graphs are not available but it has been mentioned that all protocols performed similarly.

4.3.6. NETWORK LOAD

In the experiment, the authors of [Perf_MIL04] increased the frequency of data packet transmissions from 0.2 packets per second (senders send 1 packet every 5 seconds), to 200 packets per second (senders send 1 packet every 5 milliseconds). Figure 4.5 is the network load versus PDR result for this experiment set.

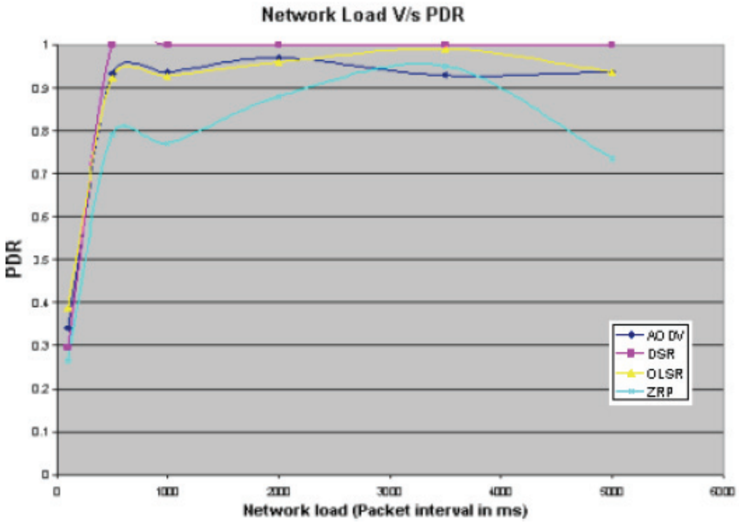


Figure 4.5: Network Load Vs PDR

The choice is between the perfect delivery of DSR in adequately provisioned cases, and the increased robustness to high traffic that OLSR provides. AODV manages to slightly outperform OLSR on three of the cases, but trail by a wider margin near the high and low ends of the packet interval spectrum. ZRP trails overall, and is not well suited to this particular experiment set.

4.4. CONCLUSION

In this chapter a qualitative analysis has been provided for AODV, OLSR, DSR, ZRP and Penaguila. The quantative analysis was not possible to be done for Penaguila, but it was necessary at least to show one for the rest of protocols. In this section, the conclusions of the study realized are explained.

The AODV and DSR protocols will perform better in the networks with static traffic and with a number of source and destination pairs relatively small for each host. In this case, AODV and DSR use fewer resources than OLSR, because the control overhead is small. Also, they require less bandwidth to maintain the routes. Besides, the routing table is kept small reducing the computational complexity. Both reactive protocols can be used in resource critical environments.

The OLSR protocol is more efficient in networks with high density and highly sporadic traffic. The quality metrics are easy to expand to the current protocol. Hence, it is possible for OLSR to offer QoS. However, OLSR requires that it continuously have some bandwidth in order to receive the topology updates messages.

The scalability of both classes of protocols is restricted due to their proactive or reactive characteristics. For reactive protocols, it is the flooding overhead in the high mobility and large networks. For OLSR protocol, it is the size of the routing table and topological updates messages.

ZRP is supposed to perform well in large networks with low area overlapping. But in any of the papers considered to write this thesis ZRP showed a better performance that the other protocols. Besides, and as a disadvantage, there is an optimum zone radius for each environment as was studied in [OPTZRP_99] that is difficult to determine.

Penaguila protocol is supposed to outperform the rest of the protocols under study in large networks with nodes having different traffic rates and different mobility degrees. Each node decides if it is better to work in proactive or in reactive mode. Hence, every node adjusts the control overhead and the resource usage to its necessities.

5. SUMMARY

In this master thesis there has been a description of what MANETs are and why they are so interesting. Because of its characteristics, the tradicional routing protocols for wired networks are not advisable for them. A specific routing protocol for MANETs is necessary. In this thesis the main groups of these protocols have been explained and some of the most commonly used of them were studied. We saw that each protocol is better in a specific environment. None of them are perfect for all the ranges of nodes mobility, traffic, number of nodes, etc.

The two main groups of protocols studied are the proactive and the reactive ones. The main characteristic of the proactive is that each node maintains a route to every node in the network. Besides, it periodically updates this information. No matter if there is communication between the nodes or not. As representative examples of proactive protocols, OLSR and DSDV were described here. On the other hand, in the reactive ones the nodes only calculate the routes between those nodes that want to communicate. This kind of protocols perform in a more efficient usage of the bandwith (which is very limited in the MANETs medium) and the resources of the nodes. However, as a drawback, when the route is not available yet, the delay to achieve it can be great. The reactive protocols choosen here to be studied were AODV and DSR.

In the reactive, the main problem is the delay to achieve a new route. In the proactive, it is the high usage of resources and bandwith when it is not necessary. Both, reactive and proactive also have the problem of the scalability. To solve these problems, a new kind of protocols appeared: the hybrid ones. A hybrid routing protocol combines both, the proactive and reactive to achieve better performance. The most popular of them is ZRP and its operation was described here too.

None of the existing protocols are suitable for a MANET with a large number of nodes, each one of them with a different velocity and traffic. ZRP solves in part the problem of the scalabilty, but under different patterns of traffic and nodes velocity performs worse than the OLSR, DSR and AODV. Understanding the strengths and weaknesses of each protocol, a new one was proposed. The objective of this new protocol was to be suitable to MANETs with nodes moving freely, with different ranges of speed and traffic. Also, another objective was to improve the scalability of the reactive and proactive protocols.

The protocol proposed here was called Penaguila. As ZRP and other hybrid routing protocols, it is based on having some nodes working in proactive mode creating areas, and comunicating this areas with other nodes working in reactive mode. The difference between Penaguila and ZRP, is that Penaguila takes into acount the speed and traffic of each node. Therefore, Penaguila tries to have each node working in the mode more suitable for itself.

Also, an evaluation of the OLSR, AODV, DSR, ZRP and Penaguila has been done. Since it was not possible to program Penaguila in NS-2 because of the short time to write the thesis, it was only feasible to do a qualitative study. In this study the advantages and disadvantages of each protocol were exposed and the concluision was that Penaguila can outperform the existing protocols when: A) The network is large, since it is a hierarchical routing protocol. B) The nodes have very different speeds and amount of traffic.

6. FUTURE WORK

This report has proposed a routing protocol for MANETs. The greatest part of the work during the five months of duration of the master thesis was to read papers and RFCs to understand what MANETs are, why they are so important, and which kind of routing protocols they use. Once the different existing routing protocols as well as their advantages and disadvantages were understood, the objective was to design a new protocol more suitable for networks with nodes moving freely. These networks should be able to be both large and small. Also the traffic pattern was taken into acount to decide the features of each node.

Since there was no time to make a quantitative study by means of simulation, only a qualitative analysis was done. Therefore, as future work, Penaguila should be programed for example in NS-2 to carry out a performance study in comparison with the other protocols already implemented.

Also in the evaluation the possible stress for the nodes working as ABRs and in R2 state was explained. In the case of ABRs and R2 nodes, the main problem can be the big storage and resources that are necessary. It should be advisable that those nodes are powerful because of the big complexity of calculus necessary (it has to run the features of two protocols at the same time). This subject has not been taken into acount here. Hence, as a future work it should be interesting to investigate a solution.

Another possible problem is also related with the R2 nodes. These nodes are working in reactive mode because they are moving fast. If there are a small quantity of them in a big area, a lot of traffic runs through them. That involves two problems:

1. If the velocity of the R2 node involves a topology change, a lot of traffic can be lost.
2. If the traffic through the R2 node is very high, the R2 node will try to change to proactive mode, causing the loss of a R2 node and maybe the loss of a big quantity of traffic.

Therefore, as future work these two possible problems should be checked out to know if they are really problems or not. In the case that were necessary, a solution should be proposed.

7. REFERENCES

[Abolhasan_04] M. Abolhasan, T. A. Wysocki, and E. Dutkiewicz, "A Review of Routing Protocols for Mobile Ad hoc Networks", In Elsevier Journal of Ad hoc Networks, 2 (2004), 1-22.

[ADHOC_01] C. E. Perkins, "Ad Hoc Networking", Addison-Wesley, Boston, 2001.

[AODV_03] C. Perkins, E.Belding-Royer, S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing", IETF RFC 3561, July 2003.

[AODV_N02] "Ad hoc On-Demand Distance Vector (AODV) Routing". November 2002 <draft-ietf-manet-aodv-12.txt>

[Benzaid et al., 2003] Benzaid M., Minet P., Al A. K. 2003. "Analysis and simulation of fast-OLSR". The 57th IEEE Semiannual Vehicular Technology Conference. 3. 1788 – 1792 pp

[Benzaid et al., 2002] Benzaid M., Minet P., Al A. K. 2002. "Integrating fast Mobility in the OLSR routing protocol". Proceedings of the 4th IEEE Conference on Mobile and Wireless Communications Networks, Stockholm, Sweden.

[CompManets] "Comparative Study of Routing Protocols for Mobile Ad-hoc NETworks". Thomas Heide Clausen, Philippe Jacquet and Laurent Viennot.

[DSDV_94] Charles E. Perkins, Pravin Bhagwat "Highly Dynamic Destination Sequenced Distance Vector (DSDV) for Mobile Computers", In Proceedings of the SIGCOMM'94 Conference on Communications Architectures, Protocols and Applications, August 1994.

[DSR_04] D. Johnson, D. Maltz, Y. Hu, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks(DSR)", IETF Internet-Draft, July 2004.

[DSR_F02] ”The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)”, February 2002. <draft-ietf-manet-dsr-07.txt>

[DYMO_06] I. Chakeres, C. Perkins, “Dynamic MANET On-demand (DYMO) Routing”, IETF Internet-Draft, March 2006.

[DYPRO_57] R. E. Bellman, “Dynamic Programming”, Princeton University Press, Princeton, NJ 1957.

[EffTraffic] “Effects of Small Transfers and Traffic Patterns on Performance and Cache Efficacy of Ad Hoc Routing” Shao-Cheng Wang, Ahmed Helmy. Department of Electrical Engineering University of Southern California {shaochew, helmy}@usc.edu.

[EncapsulationIP] C. Perkins, “Minimal Encapsulation within IP”, *RFC2004*, October 1996.

[EV_AODV] E. M. Belding-Royer and C.E Perkins, “Evolution and Future Directions of the Ad hoc On-Demand Distance Vector Routing Protocol”, Ad hoc Networks Journal, 1(1), pp. 125-150, July 2003.

[FLONET_62] L. R. Ford, D. R Fulkerson, “Flows in Networks”, Princeton University Press, Princeton, NJ, 1962.

[GloMoSim] GloMosim. http://pcl.cs.ucla.edu/projects/glomosim/

[HIGHDYN_06] Michael Sirivianos and Athanasios Leontaris “Comparative Evaluation of Ad-Hoc Routing Protocols in Highly Dynamic Environments”, 30-May-2006.

[LSR_95] J. Moy, “Link-state routing”, In Martha E. Steenstrup, editor, Routing in Communications Networks, pages 135-157. Prentice Hall, 1995.

[MANET] http://en.wikipedia.org/wiki/Mobile_ad-hoc_network

[MANET_02] C-k Toh, “Ad Hoc Mobile Wireless Networks”, Prentice Hall, 2002

[MANET_03] I. Chalamtac, M. Conti and J. Liu, “Mobile Ad hoc Networking. Imperatives and Challenges”, Ad Hoc Network Journal, Vol. 1 N. 1, January-Feburary-March, 2003.

[MANET_04] S. Basagni, M. Conti, S. Giordando, and I. Stojmenovic, “Mobile Ad Hoc Networking”, IEEE Press & Wiley Inter-Science, 2004.

[manet_99] S. Corson, J. Macker, “Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations”, IETF RFC 2501, January 1999.

[mobility04] “Scenario-oriented *ad* hoc networks simulation
with mobility models” Kevin M. McNeill, Yun Zhao, Natalia Gaviria Ralph Martinez & John Deal. MlLCOM 2004 - 2004 IEEE Military Communications Conference

[MOLSR] Philippe Jacquet et altres. Multicast Optimized Link State Routing. draft-ietf-manet-olsr-molsr-01.txt, april 2002

[MPLS] “MPLS implementation in mobile ad-hoc networks (manets)” Sasan Adibi. ECE dept, university of Waterloo

[MultiAdHoc] C. Mantelis, ”Multicasting in Ad Hoc Networks”, Unpublished document. Columbia University, USA. [Electronic] Available:
< http://comet.ctr.columbia.edu/~jaekwon/E6768/HarryMantelis-E6768Final.pdf >.
2003-10-29.

[NS_2] The Network Simulator ns2. http://www.isi.edu/nsnam/ns/

[ODM_01] K. Wu and J. Harms, “On-Demand Multipath Routing for Mobile Ad Hoc Networks”, Proceedings of European Personal and Mobile Communications Conference (EPMCC), Vienna, Austria, February 2001, Paper 21.1.

[ODM_99] A. Nasipuri and S. R. Das, “On-Demand Multipath Routing for Mobile Ad Hoc Networks”, Proceedings of the 8th Int. Conf. On Computer Communications and Networks (IC3N), Boston, October 1999.

[OPNET] Making Networks and Applications Perform. OPNET. http://www.opnet.com/

[OLSR_01] P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouti, A. Qayyum, L. Viennot, “Optimized link state routing protocol for ad hoc networks”, IEEE INMIC, Pakistan, 2001.

[OLSR_03] T. Clausen, P. Jacquet, “Optimized Link State Routing Protocol (OLSR)”, IETF RFC 3626, Octubre 2003.

[OLSR_N98] P. Jacquet, P. Muhlethaler, A. Qayyum, “Optimized Link State Routing Protocol”, Internet Draft, draft-ietf-manetolsr-00.txt, November 1998.

[OLSR_QoS] Ying Ge, Thomas Kunz and Louise Lamont “Quality of Service Routing in Ad-Hoc Networks Using OLSR.” Proceeding of the 36th Hawaii International Conference on System Science (HICSS’03).

[OPTZRP_99] M. R. Pearlman and Z. J. Haas, “Determining the Optimal Configuration for the Zone Routing Protocol”, IEEE Journal on Selected Areas in Communication, 17 (8). pp. 1395-1414, 1999.

[Perf_MIL04] MILCOM 2004 - 2004 IEEE Military Communications Conference “Performance of mobile ad hoc networking routing protocols in large scale scenarios” Julian Hsu, Sameer Bhatia, Ken Tang, Rajive Bagrodia. Scalable Network Technologies, Inc., Culver City, CA.

[PERFROUT_01] A. Boukerche, "A performance comparison of routing protocols for ad hoc networks", Apr. 2001, pp. 1940-1946.

[PERFROUT_98] J. Broch, D.A. Maltz, D. B. Johnson, Y. C. Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols", Oct. 1998, pp. 85-97.

[PERFROUT_99] P. Hohansson, T. Larsson, N. Hedman, B. Mielczarek, and M. Degermark, "Scenario-based performance analysis of routing protocols for mobile ad-hoc networks", Aug. 1999, pp.195-206.

[Qayyum, et al., 2002] Qayyum A. Laouiti A. Viennot L. 2002. Multipoint Relaying Technique for flooding broadcast messages in mobile wireless Networks in HICSS: Hawaii International Conferece on System Sciences. Hawaii, USA.

[QualNet] Scalable Network Technologies. QualNet. http://www.scalable-networks.com/

[Resilience] http://en.wikipedia.org/wiki/Resilience

[REVROUT_99]...E. M. Royer and C. K. Toh, "A review of current routing protocols for ad hoc mobile wireless networks", Apr. 1999, pp. 46-55

[rfc2018] http://www.ietf.org/rfc/rfc2018.txt

[rfc3135] http://www.ietf.org/rfc/rfc3135.txt

[ScalOLSR] "LANMAR+OLSR: A Scalable, Group Oriented Extension of OLSR", Mario Gerla, XiaoYan Hong Kaixin Xu, Yeng Lee WAM. http://www.cs.ucla.edu/NRL/wireless/. August 7, 2004, Dan Diego

[TBRPF_04] R. Ogier, F. Templin, M. Lewis, “Topology Dissemination Based on Reverse-Path Forwarding (TBRPF) ”, IETF RFC 3684, Febrero 2004.

[TrafficO] “Sustaining Performance Under Traffic Overload”. Saman DeSilva Rajendra V. Boppana. Computer Science Department The Univ. of Texas at San Antonio, San Antonio, TX 78249.

[TrafficP] “Effects of Small Transfers and Traffic Patterns on Performance and Cache Efficacy of Ad Hoc Routing” Shao-Cheng Wang, Ahmed Helmy, Department of Electrical Engineering, University of Southern California.

[YUMING_04] Shengming Jiang, Yaoda Liu, Yuming Jiang and Qinghe Yin, “Provisioning of Adaptability to Variable Topologies for Routing Schemes in MANETs”, March 12, 2004.

[ZRP_02] Z. J. Hass, R. Pearlman and P. Samar. “Zone routing protocol for ad-hoc Networks”. Internet Draft, draft-ietf-manet-zrp-04.txt, work in progress, 2002.