UNIVERSITÀ DI ROMA LA SAPIENZA

Facoltà di Ingegneria

Corso di Laurea in Telecomunicazioni

Tesi di Laurea Specialistica

Passive Clustering 2.0:

energy based and topology driven

Relatore:

Prof. M-G. Di Benedetto

Correlatore:

Dr. L. De Nardis

Paolo Greco

Matr. 794514

Anno accademico 2006-2007

Introduction

"The term wireless is normally used to refer to any type of electrical or electronic operation which is accomplished without the use of a "hard wired" connection. Wireless communication is the transfer of information over a distance without the use of electrical conductors or "wires".The distances involved may be short (a few meters as in television remote control) or very long (thousands or even millions of kilometers for radio communications). When the context is clear the term is often simply shortened to "wireless" [1]

"Clustering is the classification of objects into different groups, or more precisely, the partitioning of a data set into subsets (clusters), so that the data in each subset (ideally) share some common trait - often proximity according to some defined distance measure."[2]

Wikipedia

The above definitions are at the foundation of this work. Our goal is to define a clustering approach for a wireless network of mobile nodes aiming at performance improvement in terms of energy, overhead and complexity.
Wireless communications can be based on:

1. radio frequency communications,

2. infrared (IR) short-range communications, for example from remote controls or via IRDA.

Applications include point-to-point communication, point-to-multipoint communication, broadcasting, cellular networks and other wireless networks. In the last decades, wireless communications industry experienced drastic changes driven by many technology innovations. Figure **??** shows a picture of the evolution of wireless access technologies in the last 25 years.

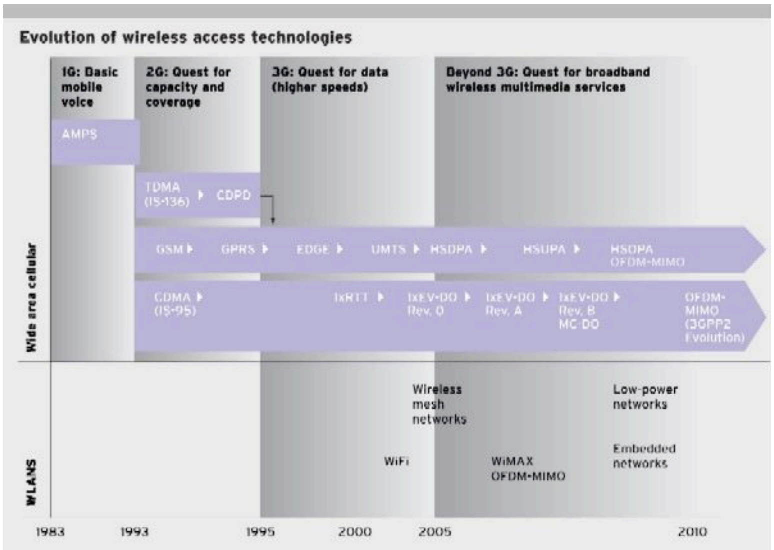

Figure 1. Evolution of networks

In the following we will focus on ad-hoc and sensor networks of mobile nodes, as this promises to be a key scenario in the evolution of the mobile wireless networks. Two aspects in the design of ad hoc ad sensor networks gathered a significant attention by the research comunity: routing and network organization. Previous work on routing led to the definition of two dominant strategies in ad hoc and sensor networks for routing: proactive and reactive routing protocols. Research work on network organization which is a key aspect in the deployment of ad hoc and sensor networks led to the definition of several approaches based on partition of nodes in sub-networks, also known as Clustering. Clustering has been proposed as solution in a lot of different fields like medicine, economics, informatics due to the fact that this method can lead to a faster and simpler solutions to a wide range of problems. Clustering has been adopted in the past for wireless networks. As an example, the partion of terminals adopted in cellular networks, enabling frequency reuse in cellular networks can be considered a simple clustering approach. Clustering promises to be a key tool in the deployment of ad-hoc and sensor networks as well. An intelligent partition of the network in sub-groups can lead to a longer network lifetimes and better network performance. The analysis of the impact of clustering on adhoc/sensor networks performance, with particular focus on multihop routing efficency, is the main topic of this thesis. Existing approaches for clustering and routing are reviewed, and a novel clustering scheme taking into account energy efficiency and network topology is proposed, built on previous work presented in [3]. The effectiveness of the proposed scheme is analyzed by extensive computer simulation.

Organization of the Work

The routing and clustering techniques, proposed in the literature for ad hoc and sensor networks are introducted in Chapter 1 and 2. Chapter 3 presents selected routing and clustering schemes and also discusses the new implementation based on energy related parameter. The proposed idea is analyzed and validated in Chapter 4, presenting simulation results. Finally, conclusions and future applications are reported in Chapter 5.

Acknowledgements

Desidero ringraziare la mia famiglia e tutte le persone che mi sono state vicine durante il mio percorso universitario.
Un ringraziamento speciale va alla "famiglia ACTS" e in particolare alla Prof. Maria Gabriella Di Benedetto, Luca De Nardis e Daniele "Dome" Domenicali.

Contents

List of Tables

List of Figures

Chapter 1

Mobile Networks

1.1 History

A wireless ad hoc sensor network consists of a number of sensors spread across a geographical area. Each sensor has wireless communication capability and some level of intelligence for signal processing and networking of the data. Some examples of wireless ad hoc sensor networks are the following:

1. Military sensor networks to detect and gain as much information as possible about enemy movements, explosions, and other phenomena of interest.

2. Sensor networks to detect and characterize Chemical, Biological, Radiological, Nuclear, and Explosive (CBRNE) attacks and material.

3. Sensor networks to detect and monitor environmental changes in plains, forests, oceans, etc.

4. Wireless traffic sensor networks to monitor vehicle traffic on highways or in congested parts of a city.

5. Wireless surveillance sensor networks for providing security in shopping malls, parking garages, and other facilities.

6. Wireless parking lot sensor networks to determine which spots are occupied and which are free.

The above list suggests that wireless ad hoc sensor networks offer certain capabilities and enhancements in operational efficiency in civilian applications as well as assist in the efforts to increase alertness to potential terrorist threats.

Two ways to classify wireless ad hoc sensor networks are whether or not the nodes are individually addressable, and whether the data in the network is aggregated. The sensor nodes in a parking lot network should be individually addressable, so that one can determine the locations of all the free spaces. This application shows that it may be necessary to broadcast a message to all the nodes in the network. If one wants to determine the temperature in a corner of a room, then addressability may not be so important. Any node in the given region can respond. The ability of the sensor network to aggregate the data collected can greatly reduce the number of messages that need to be transmitted across the network. This function of data fusion is discussed more below.

The basic goals of a wireless ad hoc sensor network generally depend upon the application, but the following tasks are common to many networks:

- Determine the value of some parameter at a given location: In an environmental network, one might went to know the temperature, atmospheric pressure, amount of sunlight, and the relative humidity at a number of locations. This example shows that a given sensor node may be connected to different types of sensors, each with a different sampling rate and range of allowed values.

- Detect the occurrence of events of interest and estimate parameters of the detected event or events: in the traffic sensor network, one would like to detect a vehicle moving through an intersection and estimate the speed and direction of the vehicle.

- Classify a detected object: is a vehicle in a traffic sensor network a car, a mini-van, a light truck, a bus, etc.

- Track an object: in a military sensor network, track an enemy tank as it moves through the geographic area covered by the network.

In these four tasks, an important requirement of the sensor network is that the required data be disseminated to the proper end users. In some cases, there are fairly strict time requirements on this communication. For example, the detection of an intruder in a surveillance network should be immediately communicated to the police so that action can be taken.

Wireless ad hoc sensor network requirements include the following:

- Large number of (mostly stationary) sensors: Aside from the deployment of sensors on the ocean surface or the use of mobile, unmanned, robotic sensors in military operations, most nodes in a smart sensor network are stationary. Networks of 10,000 or even 100,000 nodes are envisioned, so scalability is a major issue.

- Low energy use: Since in many applications the sensor nodes will be placed in a remote area, service of a node may not be possible. In this case, the lifetime of a node may be determined by the battery life, thereby requiring the minimization of energy expenditure.

- Network self-organization: Given the large number of nodes and their potential placement in hostile locations, it is essential that the network be able to self-organize; manual configuration is not feasible. Moreover, nodes may fail (either from lack of energy or from physical destruction), and new nodes may join the network. Therefore, the network must be able to periodically reconfigure itself so that it can continue to function. Individual nodes may become disconnected from the rest of the network, but a high degree of connectivity must be maintained.

- Collaborative signal processing: Yet another factor that distinguishes these networks from MANETs is that the end goal is detection/estimation of some events of interest, and not just communications. To improve the detection/estimation performance, it is often quite useful to fuse data from multiple sensors. This data fusion requires the transmission of data and control messages, and so it may put constraints on the network architecture.

- Querying ability: A user may want to query an individual node or a group of nodes for information collected in the region. Depending on the amount of data fusion performed, it may not be feasible to transmit a large amount of the data across the network. Instead, various local sink nodes will collect the data from a given area and create summary messages. A query may be directed to the sink node nearest to the desired location.

Sensor types and system architecture: With the coming availability of low cost, short range radios along with advances in wireless networking, it is expected that wireless ad hoc sensor networks will become commonly deployed. In these networks, each node may be equipped with a variety of sensors, such as acoustic, seismic, infrared, still/motion videocamera, etc. These nodes may be organized in clusters such that a locally occurring event can be detected by most of, if not all, the nodes in a cluster. Each node may have sufficient processing power to make a decision, and it will be able to broadcast this decision to the other nodes in the cluster. One node may act as the cluster master, and it may also contain a longer range radio using a protocol such as IEEE 802.11 or Bluetooth.

1.1.1 Origins and First Generation

The origin of Mobile Ad Hoc Networks can be dated back in the 1973 with DARPA PRNET project [4]. The project initiated research focusing on the feasibility of using packet-switched radio communications as the basis for reliable computer communications. Through the years the Darpa PRnet has envolved to be a robust, reliable, operational experimental network (1973-1987). A key aspect of the Darpa PRnet was the definition and the implementation of routing and automatic network management. The Packet radio network components:

- Firmware, that can be loaded into a PR either locally (via serial interface) or from the PRNET. The firmware in each PR gathers information about bidirectional link quality, nodal capacity and route characteristics and provides this knowledge to debugging and monitoring;

- Communications, that use radio frequency technology to transmit and receive data. The implemented packet radios support omni-directional, spreadspectrum, half-duplex transmission and reception at 400kbit/s and 100kbit/s rates.

The main characteristics of the DARPA PRNET acn be listed as follows :

1. Measuring from Firmware

2. Error detection using Cylinc Redundancy Checksum (32bit)

3. The devices implement chip modulation at a 12.8 mchip/s rate which produce a direct sequence, spread-spectrum waveform using pseudo-noise sequence

4. Radio frequencies capabilities are between 1718.4 Mhz and 1840.0 Mhz

5. Routing that in PRnet is designed to enable reliability, speed and correctness.

1.1.2 Second Generation

The beginning second generation can be mapped onto survivor adaptive radio networks (1980-1993). The main goal of the survivor project was to provide packet switched netwoking to the mobile battlefield elements in infrastructure less environments. Another important project of this second generation is the Global Mobile Information System [5]. GloMo initiatives include: self organizing/self healing networks, both flat and hierachical multihop routing algorithms, ATM over wireless, Georouting, Satellite comunications networks, heterogeneous networking with IP overlays, end-to-end network enhancement and security and survivability for ad-hoc networks.
GloMo technologies are applicable to Wide Area Information System, Information Systems for Dismounted Forces, and Information System for Rapid Deployment Forces. A third program dealing with second generation wireless network is Near Term Digital Radio Systems (NTDR) [6]. The NTDR program provided a prototype mobile ad-hoc network (MANET) radio system to the United States Army. The MANET protocols were provided by Bolt, Beranek and Newman; the radio hardware was supplied by ITT. These systems have been used on the field by the United Kingdom as the High Capacity Data Radio (HCDR) and by the Israelis as the Israeli Data Radio. They have also been purchased by a number of other countries for experimentation. The NTDR protocols consist of two components: clustering, and routing. The clustering algorithms dynamically organize a given network into cluster heads and cluster members. The cluster heads create a backbone; the cluster members use the services of this backbone to send and receive packets. The cluster heads use a link state routing algorithm to maintain the integrity of their backbone and to track the locations of cluster members. The NTDR routers also utilize a variant of Open Shortest Path First (OSPF) that is called ROSPF (R stands for Radio) . ROSPF does not use the OSPF hello protocol for link discovery. Instead,

OSPF adjacencies are created and destroyed as a function of MANET information that is distributed by the NTDR routers, both cluster heads and cluster members. Wireless communication systems can satisfy the need for rapid deployment of a network of independent mobile users. Significant examples include establishing survivable, efficient, dynamic communication for emergency/rescue operations, disaster relief efforts, and military networks. Such network scenarios cannot rely on centralized and organized connectivity, and can be identified as applications of Mobile Ad Hoc Networks (MANET). A MANET is an autonomous collection of mobile users that communicate over relatively bandwidth constrained wireless links. Since the nodes are mobile, the network topology may change rapidly and unpredictably over time. The network is decentralized, and all network activity including discovering the topology and delivering messages must be executed by the nodes themselves, i.e., routing functionality will be incorporated into mobile nodes.

1.1.3 Evolution and future works

As consequences of technical advancemnet Mobile Ad-Hoc Networks is revived as a potential technology. Their importance has been enforced by the recent release of the IEEE 802.15.4 [7] standard for the physical and MAC layers. The IEEE 802.15.4 standard defines the characteristics of the physical and MAC layers for Low-Rate Wireless Personal Area Networks (LR-WPAN). The advantages of an LR-WPAN are ease of installation, reliable data transfer, short-range operation, extremely low cost, and a reasonable battery life, while maintaining a simple and flexible protocol stack. 802.15.4 forms the basis for the ZigBee industrial standard [8]. The ZigBee Alliance is an association of companies working together to develop standards (and products) for reliable, cost-effective, low-power wireless networking and it is foreseen that ZigBee technology will be embedded in a wide range of products and applications across consumer, commercial, industrial and government markets worldwide. As previously mentioned ZigBee builds upon the IEEE 802.15.4 standard which defines the physical and MAC layers. It defines the network layer specifications, handling star and peer-to-peer network topologies, and provides a framework for application programming in the application layer. It is worth mentioning that the IEEE 80.15.4 standard was recently updated by introducing ranging and positionign capabilities, by means of novel UWB physical layer, leading to the release of 802.15.4a version [9].

1.2 Routing in MANETs

The limited computing and communications resources avaiable in MANETs make the design of an efficient and reliable routing strategy a challenging problem.
An intelligent routing strategy is required to efficiently use the limited resources while at the same time being adaptable to the changing network conditions such as: network size, traffic density and network partitioning.
At the same time with this, the routing protocol may need to provide different levels of QoS to different types of applications and users. Prior to the increased interest in wireless networking, two main algorithms were used in wired networks. These algorithms are commonly referred to as the link-state and distance vector algorithms.
In link-state routing, each node maintains an up-to-date view of the network by periodically broadcasting the link-state costs of its neighbouring nodes to all other nodes using a flooding strategy. When a node receives an update packet, it updates its view of the network and its link-state information by applying a shortest-path algorithm in order to choose the next hop node for each destination.
In distance-vector routing, each node i maintains a set of distances D^x_{ij} towards each destination x,where index j varies over the neighbours of node i. Node i selects a neighbour, k, to be the next hop for x if $D^x_{ik} = minD^x_{ij}$. This allows each node to select the shortest path to each destination. The distance-vector information is updated at each node by a periodical dissemination of the current estimate of the shortest distance to every node.
The traditional link-state and distance-vector algorithms presented above do not adapt efficiently to large MANETs.
This is because periodic or frequent route updates in large networks may consume significant part of the available bandwidth, increase channel contention and may lead to an excessive energy consumption in nodes requiring thus to frequently recharge their power supply.

In order to overcome the problems associated with the link-state and distance-vector algorithms, a number of routing protocols have been proposed for MANETs.
These protocols can be classified into three different groups as shown in the Figure 1.1.

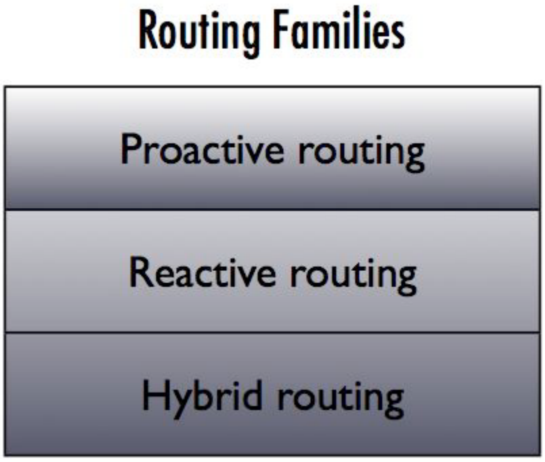

Figure 1.1. Routing Families

In proactive routing protocols, the routes to all the destinations (or parts of the network) are determined at network start up, and then maintained by using a periodic route update process.
In reactive protocols, routes are determined when they are required by the source, and are obtained by using a route discovery process.
Hybrid routing protocols combine the basic properties of the above two classes of protocols into one. That is, they are both reactive and proactive in nature.
Each class includes a number of different routing strategies, which employ either a flat or a hierarchical routing structure. The most important protocols in each class are described in the following section.

1.2.1 Proactive routing protocols

In proactive routing protocols, each node maintains routing information to every other node (or nodes located in a specific part) in the network. The routing information is usually kept in a number of different tables. These tables are updated on a periodic basis and/or if the network topology changes. Proactive protocols mainly differ in the way the routing information is updated, detected and in the type of information kept in each routing table and in the number and size of the tables. Two fundamental proactive routing protocols are:

- DSDV Destination Sequenced Distance Vector
- OLSR Optimized Link State Routing

Several other protcolos were defined as a variation of the two above.

DSDV: *Destination Sequenced Distance Vector*

The DSDV [10] algorithm guarantees loop free routes. It provides a single path to a destination, which is selected using the distance vector shortest path routing algorithm. In order to reduce the amount of overhead transmitted through the network, two types of update packets are used. These are referred to as a "full dump" and "incremental" packets,respectively. A full dump packet carries all the available routing information, while an incremental packet carries only the information changed since the last full dump. The incremental update messages are sent more frequently than the full dump packets. However, DSDV introduces large amounts of overhead in network operation due to the requirement of the periodic update messages, with overhead growing according to a $O(N^2)$ law. Therefore the protocol does not scale well in large networks since a large portion of the network bandwidth is used in the updating procedures.

OLSR: *Optimized Link State Routing*

OLSR [11] is a point-to-point routing protocol based on the traditional link-state algorithm. In this protocol, each node maintains topology information about the network by periodically exchanging link-state messages. The novelty of OLSR is that it minimises the size of each control message and the number of rebroadcasting nodes during each route update by employing a multipoint replaying (MPR) strategy. Tge strategy works as follows: during each topology update, each node in the network selects a set of neighbouring nodes to retransmit its packets. Nodes belonging to the set are called the multipoint relays of that node. Any node which is not in the set can read and process each packet but does not retransmit. In order to select the MPRs, each node periodically broadcasts a list of its one hop neighbours using hello messages. From the list of nodes in the hello messages, each node selects a subset of one hop neighbours, which covers all of its two hop neighbours. For example, in Figure 1.2, node A can select nodes B, C, K and N to be the MPR nodes, since these nodes cover all the nodes which are two hops away. Each node determines an optimal route (in terms of hops) to every known destination using its topology information (from the topology table and neighbouring table), and stores this information in a routing table. Therefore, routes to every destination are immediately available when data transmission begins.

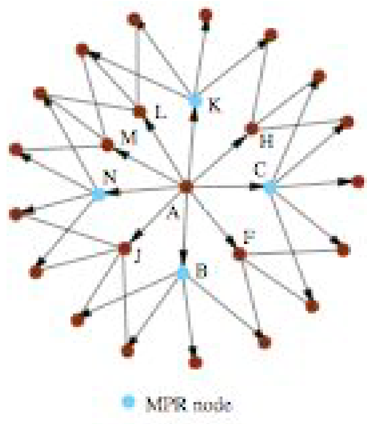

Figure 1.2. OLSR example

1.2.2 Reactive routing protocols

On-demand routing protocols were designed to reduce the overhead obeserved in proactive protocols by maintaining information for active routes only. This means that routes are determined and maintained for nodes that require to send data to a particular destination. Route discovery usually occurs by flooding route request packets through the network. When a node with a route to the destination (or the destination itself) is reached, a route reply is sent back to the source node using link reversal if the route request has travelled through bi-directional links or by piggy-backing the route in a route reply packet via flooding. Therefore, the route discovery overhead (in the worst case scenario) will grow by $O(N + M)$ when link reversal is possible and $O(2N)$ for uni-directional links. Reactive protocols can be classified into two categories:

- source routing;
- table driven.

Source Routing

In source routed on-demand protocols, each data packets carries the complete source to destination address. Therefore, each intermediate node forwards these packets according to the information kept in the header of each packet. This means that the intermediate nodes do not need to maintain update routing information for each active route in order to forward the packet towards the destination. Furthermore, nodes do not need to maintain neighbour connectivity through periodic beaconing messages. The major drawback with source routing protocols is that in large networks they do not perform well. This is due to two main reasons. First as the number of intermediate nodes in each route grows, then so does the probability of route failure. To show this let $P(f) = a \cdot n$, where $P(f)$ is the probability of route failure, a is the probability of a link failure and n is the number of intermediate nodes in a route. From this, it can be seen that as $n- > \infty$, then $P(f)- > \infty$. Second, as the number of intermediate nodes in each route grows, the amount of overhead carried in each header of each data packet grows as well. Therefore, in large networks with significant levels of multihopping and high levels of mobility, these protocols may not scale well. An example of this group is the Dynamic Source Routing protocol (DSR) presented below.

DSR: Dynamic Source Routing The DSR protocol [12] requires each packet to carry the full path (every hop in the route), from source to the destination. This means that the protocol will not be very effective in large networks, as the amount of overhead carried in the packet will increase as the network diameter increases. Therefore in highly dynamic and large networks the overhead may consume most of the bandwidth. However, this protocol provides a simple and effective solution for small to average size networks (perhaps up to a few hundred nodes). An advantage of DSR is that nodes can store multiple routes in their route cache, which means that the source node can check its route cache for a valid route before initiating route discovery, and if a valid route is found there is no need for route discovery. This is very beneficial in networks with low mobility, as routes stored in the route cache will be valid for longer time. Another advantage of DSR is that it does not require any periodic beaconing (or hello message exchanges), therefore nodes can enter sleep node to preserve their power. This also saves a considerable amount of bandwidth in the network.

Table Driven

In table driven routing (also known as point-to-point routing), each data packet only carries the destination address and the next hop address. Therefore, each intermediate node in the path to the destination uses its routing table to forward each data packet towards the destination. The advantage of this strategy is that routes are adaptable to the dynamically changing environment of MANETs, since each node can update its routing table when it receives fresher topology information and hence forward the data packets over fresher and better routes. Using fresher routes also means that fewer route recalculations are required during data transmission. The disadvantage of this strategy is that each intermediate node must store and maintain routing information for each active route and each node is required to be aware of their surrounding neighbours through the use of beaconing messages. In the following the most important table driven reactive protocols are reviewed.

LMR: *Light Weight Mobile Routing*

The LMR [13] protocol is an on-demand routing protocol, which uses a flooding technique to determine its routes. The nodes in LMR maintain multiple routes to each required destination. This increases the reliability of the protocol in case of route failure, by allowing nodes to select the next available route to a particular destination without initiating a route discovery procedure. Another advantage of this protocol is that each node only maintains routing information to its neighbours, thus avoiding extra delays and storage overheads associated with maintaining complete routes. However, LMR may produce occasionally temporary invalid routes, introducing extra delays due to the presence of loops.

TORA: *Temporally Order Routing Algorithm*

The TORA [14] routing protocol is based on the LMR protocol. It uses similar link reversal and route repair procedures, and also the creation of Directed acyclic graph (DAG), which is similar to the query/reply process used in LMR. Therefore, it also has the same benefits as LMR. The advantage of TORA is that it reduces the far-reaching control messages to a set of neighbouring nodes where the topology change has occurred. Another advantage of TORA is that it also supports multicasting, although this is not incorporated into its basic operation. TORA can be used in conjunction with lightweight adaptive multicast algorithm (LAM) to provide multicasting. The disadvantage of TORA is that the algorithm may also produce temporary invalid routes as LMR does.

AODV: *Ad-hoc On Demand Distance Vector*

The AODV [15] routing protocol is based on DSDV and DSR algorithms seen before. It uses the periodic beaconing and sequence numbering procedure of DSDV and a similar route discovery procedure as in DSR. However, there are two major differences between DSR and AODV. The most distinguishing difference is that in DSR each packet carries full routing information, whereas in AODV the packets only carry the destination address. This means that AODV has potentially less routing overheads than DSR. The other difference is that the route replies in DSR carry the address of every node along the route, whereas in AODV the route replies only carry the address of the destination and the sequence number. The advantage of AODV is that it is adaptable to highly dynamic networks. However, nodes may experience large delays during route construction, and link failure may initiate another route discovery, which introduces extra delays and consumes more and more bandwidth as the size of the network increases.

1.2.3 Hybrid routing protocols

Hybrid routing protocols combine properties and characteristics of proactive and reactive protocols. These protocols increase scalability by allowing nodes within close proximity to work together to form a sort of backbone thus reducing the route discovery overhead. This is mostly achieved by proactively maintaining routes to nearby nodes and determining routes to far away nodes by means of a route discovery-based strategy. Most hybrid protocols proposed to date are zone-based, meaning that the network is partitioned or seen as a number of zones by each node.

ZRP: *Zone Routing Protocol*

The ZRP [16] introduces the concept of routing zone. The routing zone of a node x defines a range (in hops) within which each node is required to maintain proactively network connectivity. Therefore, routes towards nodes within the routing zone, are immediately available at node x, routes towards nodes that lie outside the routing zone, are determined on-demand (i.e. reactively): any on-demand routing protocol can be used to determine a route to the required destination. The advantage of this protocol is that it significantly reduces the amount of communication overhead when compared to pure proactive protocols. It also reduces the delays associated with pure reactive protocols such as DSR, by allowing faster route discovery. This is because, in order to determine a route to a node outside the routing zone, the routing only has to travel to a node which lies on the boundaries (edge of the routing zone) of the required destination. Since the boundary node would proactively maintain routes to the destination it could complete the route from the source to the destination by sending a reply back to the source with the required routing address. The behaviour of the ZRP can be modified by changing the size of the routing zone: for large size of the routing zone the protocol resembles pure proactive protocol, while for small sizes it behaves like a reactive protocol.

1.3 Mobility Models

In order to carry out an accurate performance evaluation of the proposed protocols It is paramount to use a mobility model that accurately represents the behaviour of the mobile nodes (MNs) that will eventually run the protocols. Two types of mobility models are currently used in the simulation of networks of mobile nodes: trace-based and synthetic models. Trace-based models reproducs mobility patterns that are recorded in real life systems. Traces provide accurate information, especially when they involve a large number of participants and an appropriately long observation period. However, new network environments (e.g. ad hoc networks) are not easily modeled if traces have not yet been acquireed. In this cases use synthetic models are the only possible solution. Synthetic models attempt to realistically represent the behavior of MNs without the use of traces. In the following we will present several synthetic mobility models that have been proposed for (or used in) the performance evaluation of ad hoc network protocols. A mobility model should attempt to mimic the movements of real MNs. Changes in speed and direction must occur and they must occur in reasonable time slots. For example, we would not want MNs to travel in straight lines at constant speeds throughout the course of the entire simulation because real MNs would not travel in such a restricted manner.

1.3.1 Entity mobility models

Entity mobility models determine the movement pattern of each mobile node indipendently from the behaviour of other nodes in the network. The most important synthetic entity mobility models for ad hoc networks are the following ones:

1. Random Walk Mobility Model (including its many derivatives): A simple mobility model based on random directions and speeds.

2. Random Waypoint Mobility Model: A model that includes pause times between changes in destination and speed.

3. Mass Mobility: In this model each mobile host moves within a room according to the following pattern; the node moves along a straight line for a certain period of time before it makes a turn.

4. Random Direction Mobility Model: this model forces MNs to travel to the edge of the simulation area before changing direction and speed.

5. Boundless Simulation Area Mobility Model: A model that converts a 2D rectangular simulation area into a torus-shaped simulation area; allowing nodes to disappear on one edge and reappear immediately on from the opposite one.

6. Gauss-Markov Mobility Model: A model that uses one tuning parameter to vary the degree of randomness in the mobility pattern.

7. A Probabilistic Version of the Random Walk Mobility Model: A model that utilizes a set of probabilities to determine the next position of an MN.

8. City Section Mobility Model: The model forces MNs to more within a simulation area that represents streets within a city.

There are other synthetic entity mobility models available for the performance evaluation of a protocol in a cellular network or personal communication system (PCS). Although some of these mobility models could be adapted to an ad hoc network, this work focuses on those models that have been proposed for (or used in) the performance evaluation of an ad hoc network.

1.3.2 Group Mobility

Group mobility models allow the simulation of scenarios where the MNs decisions on movement depend upon the other MNs in group. The most important models belonging to this family are the following ones:

1. Exponential Correlated Random Mobility Model: A group mobility model that uses a motion function to create movements.

2. Column Mobility Model: A group mobility model where the set of MNs form a line and are uniformly moving forward in a particular direction.

3. Nomadic Community Mobility Model: A group mobility model where a set of MNs move together from one location to another.

4. Pursue Mobility Model: A group mobility model where a set of MNs follow a given target.

5. Reference Point Group Mobility Model: A group mobility model where group movements are based upon the path traveled by a logical center.

Group mobility models are most suitable when one needs to introduces relationships between nodes mobility patterns, such as in the simulation of security/police teams movement. In this work we decided to consider a more general scenario, where nodes move indipendently one from the other: as a consequence an entity mobility model was selected. In particular, the Mass Mobility models was selected due to its case of implementation and flexibility.

Chapter 2

Clustering

2.1 What is Clustering?

A wireless ad hoc network consists of nodes that can move freely and communicate with each other using wireless links. Ad-hoc networks do not use specialized routers for path discovery and traffic routing.

One way to support efficient communication between nodes is to develop a wireless backbone architecture; this means that certain nodes must be selected to form the backbone.

Over time, the backbone must change to reflect the changes in the network topology as nodes move around. The algorithm that selects the members of the backbone should naturally be fast, but should also require as little communication between nodes as possible, since mobile nodes are often powered by batteries. One way to solve this problem is to group the nodes into clusters, where one node in each cluster functions as clusterhead, responsible for intra-cluster routing and one or more nodes are elected as gateway responsibile for inter-clusters routing. The operation of partitioning the network into smaller groups called clusters is referred to as clustering. This solution grants several beneficial effects:

- it enables the spatial reuse of resources in order to increase the system capacity. In a non-overlapping multicluster structure, two clusters may deploy the same frequency or code set if they are not neighboring clusters.
  Also, a cluster can better coordinate its transmission events with the help of a special mobile node, such as a clusterhead, residing in it. This can save much resources used for retransmission thanks to reduced packet collisions.

- it increases routing efficiency, because the set of clusterheads and clustergateways can form a virtual backbone for inter-cluster routing: the generation and spreading of routing information can be restricted in this set of nodes.

- a cluster structure makes an ad hoc network appear smaller and more stable in the view of each mobile terminal. When a mobile node changes its attaching cluster, only mobile nodes residing in the old and new clusters need to update the information. Thus, local changes do not lead to an update involving the entire network, and information processed and stored by each mobile node is greatly reduced.

Clustering costs

The adoption of a cluster-based network organization in a MANET has its side effects and drawbacks, because the costruction and maintainance of a cluster structure usually requires additional cost compared with a flat-based network organization. The correct evaluation of the clustering cost is a key issue to validate the effectiveness and scalability enhancement of a cluster structure. By analyzing the cost of a clustering scheme under different aspects, both qualitatively and quantitatively, its usefulness and drawbacks can be clearly specified.
The aspects to be taken into account in evaluating the clustering cost terms can be described as follows:

- When the network changes dinamically the clustering structure requires explicit message exchange: this exchange can consume considerable bandwidth and drain mobile nodes' energy quickly.

- Some clustering schemes require to rebuild the whole cluster structure when a local event takes place (for example when a node dies). This event is called Ripple effect.

- Some clustering schemes separate the clustering initialization from the maintenance with a "freeze" period in nodes mobility for the initialization. This approach can be unsuitable for mobile ad hoc networks where high mobility is present.

- The number of rounds in which a cluster formation procedure can be completed is another relevant metric. In many schemes, not all mobile nodes can decide their status at the same time (within one round); furthermore variable number of rounds can be required to finish the cluster construction. Thus, the time required for these algorithms cannot be bounded and may vary noticeably for different network topologies.

2.2 History of algorithms for clustering

2.2.1 k-Means

The K-means algorithm (proposed by MacQueen in 1967) is one of the simplest unsupervised learning algorithms that solves the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori. The main idea is to define k centroids, one for each cluster. These centroids shoud be placed in a cunning way because different centroid location leads to different results. It can be shown that the best choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early groupage is done. The procedure is repeted iterateively. At each iteration point we proceed to re-calculate k new centroids as barycenters of the clusters resulting from the previous step. Once the k new centroids are determined, a new binding is carried out between the same data set points and the nearest new centroid. As a result of the iterative process the k centroids change their location at each iteration until no more changes are done. In other words centroids do not move any more. The clustering is then completed.
This algorithm aims at minimizing an objective function, in this case a squared error function. The objective function is:

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} ||x_i^j - c_j||$$

where $||x_i^j - c_j||$ is a chosen distance measure between a data point x_i^j and the cluster centre c_j,and it is an indicator of the distance of the n data points from their respective cluster centres. The algorithm is composed of the following steps:

1. Place K points into the space represented by the objects that are being clustered. These points represent initial group centroids.

2. Assign each object to the group that has the closest centroid.

3. When all objects have been assigned, recalculate the positions of the K centroids.

4. Repeat Steps 2 and 3 until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized can be calculated.

Although it can be proved that the procedure will always terminate, the k-means algorithm does not necessarily find the most optimal configuration, corresponding to the global objective function minimum. The algorithm is also significantly sensitive to the initial randomly selected cluster centres. The k-means algorithm can be run multiple times to reduce this effect.

2.2.2 QT-CLustering

QT (quality threshold) clustering (proposed by Heyer et al in 1999) is an alternative method of partitioning data, originally proposed for gene clustering. It requires more computing power than k-means, but it does not require to specify the number of clusters a priori, and always returns the same result when run several times. The algorithm can be described as follows:

1. The user chooses a maximum diameter (threshold) for clusters.

2. Build a candidate cluster for each point by including the closest point, the next closest, and so on, until the diameter of the cluster surpasses the threshold.

3. Save the candidate cluster with the most points as the first true cluster, and remove all points in the cluster from further consideration.

4. Recurse with the remaining set of points.

The distance between a point and a group of points is computed using complete linkage, i.e. as the maximum distance from the point to any member of the group.

2.3 Clustering Techniques

A large number of techniques have been proposed to define the cluster structure. We can divide the most important clustering schemes into six big families presented in Table 2.1

Clustering Schemes	Purposes
Ds-Based	Finding a (weakly) connected dominating set to reduce the number of nodes participating in route search or routing table maintenance.
Low maintainance	Providing a cluster infrastructure for upper layer applications with minimized clustering-related maintainance cost.
Mobility aware	Utilizing mobile nodes' mobility behavior for cluster construction and maintenance and assigning mobile nodes with low relative speed to the same cluster to tighten the connection in such a cluster.
Energy efficient	Avoiding unnecessary energy consumption or balancing energy consumption for mobile nodes in order to prolong the lifetime of mobile terminals and a network.
Load balancing	Distributing the workload of a network more evenly into clusters by limiting the number of mobile nodes in each cluster in a defined range
Combined metric based	Considering multiple metrics in cluster configuration, including node degree, mobility, battery energy, cluster size, etc., and adjusting their weighting factors for different application scenarios.

Table 2.1. Classification of clustering schemes

2.3.1 Dominating Set

Routing based on a set of dominating nodes [17, 18] , acting as the clusterheads to relay routing information and data packets, is a typical technique in MANETs. Such a set of nodes is called a Dominating Set (DS). Taking a MANET as an un-weighted graph G, a vertex (node) subset S of G is a DS if each vertex in G either belongs to S or is adjacent to at least one vertex in S. For example, in Figure 2.1 the black vertices form a DS. Each area surrounded by the dash line is the dominating area corrisponding to a specific dominating node. The vertices of a DS can be thus used as clusterheads, by assigning each vertex to a cluster corresponding to the vertex that dominates it. A DS is called a connected DS (CDS) if all the dominating nodes are directly connected with each other. As shown in the Figure 2.1 below the black vertices form a CDS and the black lines indicate the corresponding induced subgraph of the CDS. The idea of finding a CDS for a MANET comes from the fact that the routing process is only aggregated on mobile nodes in the DS [16]. Hence, when table-driven routing is applied, only nodes in the CDS are required to construct and maintain the routing tables. When on-demand routing is adopted, the route search space is limited to the CDS.

Wu's CDS

The distributed algortihm proposed by Wu [17] provides a way to find the connected dominant set. At start time every node v exchanges its neighbor list with all its neighbours. A mobile node sets itself as a dominating node if it has at least two unconnected neighbours. This is called a marking-process. Some rules are implemented to reduce the size of a CDS generated from the marking process las follows: a node deletes itself from the CDS when its close (open) neighbour set is completely included in the neighbor set(s) of a (two connected) neighboring dominating node(s) and it has smaller ID than the neighboring dominating node(s). A CDS with small

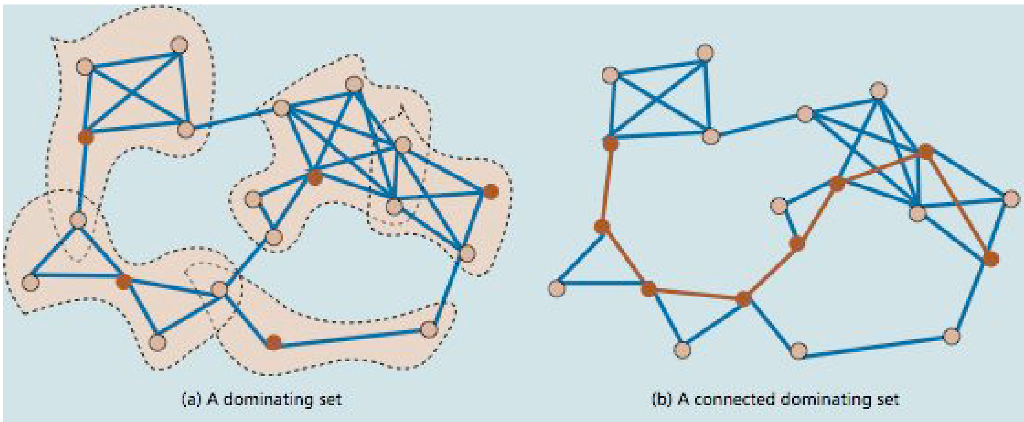

Figure 2.1. Example of Dominating Set

size reduces the number of nodes involved in routing-related tasks. The extension rules proposed in Wu's algorithm are effective for reducing the size of the DS. Also, Wu's algorithm allows the cluster construction can be completed in just two rounds, one round for the marking procedure and the second round for the application of the extension rules, so that the time complexity of this algorithm can be highly bounded. On the other hand, the maintenance events in Wu's CDS algorithm are expensive in terms of transmission efficiency and control overhead. Moreover, the actual efficiency improvement of this algorithm in presence of mobility is not verified.

2.3.2 Low Mainteinance

When network topology changes frequently, resulting in frequent cluster topology updates, the control overhead for cluster maintenance increases drastically. Thus, clustering behavior may consume a large portion of network bandwidth, drain mobile nodes energy quickly, and override its improvement on network scalability and performance. Hence, it is important to reduce the communication overhead caused by cluster maintenance. Most low-maintenance clustering protocols aim at providing stable cluster architecture by reducing the re-affiliation rate and especially minimizing re-clustering situations [3][19][20]. This approach is justified by the observation that re-clustering is more disturbing than re-affiliation since it causes more communication overhead, route invalidation, and even ripple effect. Here, re-affiliation refers to a non-clusterhead changing the attached cluster without affecting the corresponding clusterhead(s). By limiting re-affiliation and re-clustering events, the clustering-related control overhead can be reduced accordingly. A newly proposed scheme tries to eliminate completely the control overhead for clustering by constructing and maintaining cluster architecture.

Passive Clustering

In Passive Clustering (PC), a mobile node can be in one of the following four states: initial, clusterhead, gateway, and ordinary node [3]

. All mobile nodes are in "ordinary" state at the beginning of network operation. Only mobile nodes with "initial" state have the potential to be clusterheads. When a potential clusterhead in "initial" state has something to send, such as a flood search, it declares itself as a clusterhead by piggybacking its state in the packet. Neighbours can learn the clusterhead claim by monitoring the "cluster state" field in the packet, and then record the cluster ID and the packet receiving time. A mobile node that receives a claim from just one clusterhead becomes an ordinary node, while a mobile node that hears more claims can become a gateway. Since PC does not send any explicit clustering-related message to maintain the cluster structure, each node is responsible for updating its own cluster status by keeping a timer. Also, PC adopts a heuristic rule to control the number of gateways in a local area without breaking its passive nature. PC can form and maintain its cluster structure without explicitly exchanging any clustering control packet. Thus, it can completely eliminate the control overhead caused by active clustering. Furthermore PC does not require a "freeze" period to create the cluster structure.

2.3.3 Mobility Aware

Mobility is a prominent characteristic of MANETs, and is the main factor affecting topology change and route invalidation. Some believe that it is important to take the mobility metric [21, 22] into account in cluster construction in order to form a stable cluster structure and decrease its influence on cluster topology. Mobility-aware clustering indicates that the cluster architecture is determined by the mobility behavior of mobile nodes. The idea is that by grouping mobile terminals with similar speed into the same cluster, the intra-cluster links can become more tightly connected. Then the re-affiliation and re-clustering rate can be naturally decreased. An example is DDCA (Distributed Dynamic Clustering Algorithm) [23]. DDCA attempts to partition a number of mobile nodes into multi-hop clusters based on (α,t) criteria. The (α,t) criteria indicate that every mobile node in a cluster has a path to every other node that will be available over some time period t with a probability $\geq \alpha$ regardless of the hop distance between them. The purpose is to support robust and efficient routing, and adaptively adjusts its dominant routing scheme depending on the network mobility manner.

MOBIC

MOBIC suggests that cluster formation, especially clusterhead election, should take mobility into consideration. It points out that clusterhead election is a local activity so that a clusterhead should be determined only by its neighbours and itself. MOBIC [21] proposes an aggregate local mobility metric for the cluster formation process such that mobile nodes with low speed relative to their neighbors have the chance to become clusterheads. In MOBIC, the aggregate local speed of a mobile node is estimated by calculating the variance of a mobile node's speed relative to each of its neighbours. The problem is the node with low variance. Mobile nodes with low variance values in their neighborhoods take the clusterhead responsibility.

It is easy to see that MOBIC is feasible and effective for MANETs with group mobility behavior, in which a group of mobile nodes moves with similar speed and direction, as in highway traffic. Thus, a selected clusterhead can normally guarantee a low mobility with respect to its member nodes. However, if mobile nodes move randomly and change their speeds from time to time, the performance of MOBIC may be greatly degraded.

2.3.4 Energy Aware

Mobile nodes in a MANET normally depend on battery power supply during operation, hence the energy limitation poses a severe challenge for network performance [24, 25]. A MANET should strive to reduce its energy consumption greedily in order to prolong the network lifespan. Also, a clusterhead bears extra work compared with ordinary members, and it more likely "dies" early because of excessive energy consumption. The lack of mobile nodes due to energy depletion may cause network partition and communication interruption. Hence, it is also important to balance the energy consumption among mobile nodes to avoid node failure, especially when some mobile nodes bear special tasks or the network density is comparatively sparse.

IDLBC: Id Load Balancing Clustering

Each mobile node has a variable, virtual ID (VID), and the value of VID is set as its ID number at first. Initially, mobile nodes with the highest IDs in their local area win the clusterhead role. IDLBC limits the maximum time units that a node can serve as a clusterhead continuously, so when a clusterhead exhausts its duration budget (MaxCount), it resets its VID to 0 and becomes a non-clusterhead node. When two clusterheads move into the reach range, the one with higher VID wins the clusterhead role. Each non clusterhead node keeps a circular queue for its VID and shifts the VID value by one every time unit in one direction. Thus, when a clusterhead resigns, a non-clusterhead with the largest VID value in the neighborhood can resume the clusterhead function. IDLBC tries to avoid possible node failure due to energy depletion caused by excessively shouldering the clusterhead role. When a mobile node resigns its clusterhead status because of the expiration of its duration budget, another mobile node with the highest VID in the local area is chosen to

resume the clusterhead function. The newly chosen mobile node is the one whose previous total clusterhead serving time is the shortest in its neighborhood, and this should guarantee good energy level for being a new clusterhead.
This kind of new clusterhead selection may introduce ripple effect of re-clustering over the whole network without considering the network topology. In addition, the clusterhead re-election may require time synchronization of the VID value shift among different mobile nodes. Otherwise, the VID information may not be accurate enough to select the most suitable node to serve as new clusterhead. In addition, the clusterhead serving time alone may not be able to provide a good indication of energy consumption of a mobile node.

2.3.5 Load Balancing

Load balancing clustering [26] algorithms believe that there is an optimum number of mobile nodes that a cluster can handle, especially in a clusterhead-based MANET. A too-large cluster may put too heavy of a load on the clusterheads, causing clusterheads to become the bottleneck of a MANET and reduce system throughput. A too-small cluster, however, may produce a large number of clusters and thus increase the length of hierarchical routes, resulting in longer end-to-end delay. Load-balancing clustering schemes set upper and lower limits on the number of mobile nodes that a cluster can deal with. When a cluster size exceeds its predefined limit, re-clustering procedures are invoked to adjust the number of mobile nodes in that cluster.

DLBC: Degree Load Balancing Clustering

DLBC periodically runs the clustering scheme in order to keep the number of mobile nodes in each cluster around a system parameter, ED, which indicates the optimum number of mobile nodes that a clusterhead can handle. A clusterhead degrades to an ordinary member node if the difference between ED and the number of mobile nodes that it currently serves exceeds some value, MaxDelta. This mechanism tries to make all clusterheads almost serve the same and optimal number of member nodes. However, since the clusterhead change is still based on node degree, DLBC likely will cause frequent re-clustering because the movement of mobile nodes and consequent link setup/break results in dynamic variation of mobile node degree. In addition, how to select a clusterhead is not addressed in DLBC if in a local area no mobile nodes can satisfy the degree difference requirement between ED and its current node degree (2-hop).

2.3.6 Combined Metrics

Combined metrics based [27, 28] clustering takes a number of metrics into account for cluster configuration, including node degree, residual energy capacity, moving speed, and so on. This category aims at electing the most suitable clusterhead [29] in a local area, and does not give preference to mobile nodes with certain attributes, such as lowest ID or highest node degree. One advantage of this clustering scheme is that it can flexibly adjust the weighting factors for each metric to adjust to different scenarios. For example, in a system where battery energy is more important, the weighting factor associated with energy capacity can be set higher. However, not all of these parameters are always available and accurate, and the information inaccuracy may affect clustering performance [30].

WCA: Weighted Combinated Metrics

The proposed Weighted Clustering Algorithm (WCA) takes into consideration different paremeters:

- the ideal degree
- transmission power
- mobility and battery power of a mobile node.

The algorithm tries to keep the number of nodes in a cluster around a pre-defined threshold to facilitate the optimal operation of the medium access control (MAC) protocol. The clusterhead election procedure is not periodic but is based on the dynamism of the nodes. This on-demand execution of WCA aims to maintain the stability of the network, thus lowering the computation and communication costs associated with it.

Chapter 3

Motivation and used Strategies

The goal of this work is to analize and improve efficiency of clustering in wireless networks of mobile nodes. Based on the analogies carried out out in previos work it has been choose to improve and implement Passive Clustering strategy over the AODV routing protocol, in presence of mobility.
The reasons are:

- Passive Clustering seems to be a good solution in terms of overhead and complexity. It could be implemented without any specific signaling and it could safe energy reducing the traffic control.

- AODV is a reactive protocol that can be work well in presence of mobility.

In the next section both Aodv protocol and Passive Clustering solution are explicitated.

3.1 AODV

The Ad hoc On Demand Distance Vector (AODV)[15] routing algorithm is a routing protocol designed for ad hoc mobile networks.
AODV is capable of both unicast and multicast routing. It is an on demand reactive algorithm, meaning that it builds routes between nodes only as desired by source nodes.
It maintains these routes as long as they are needed by the sources. Additionally, AODV forms trees which connect multicast group members. The trees are composed of the group members and the nodes needed to connect the members. AODV uses sequence numbers to ensure the freshness of routes. It is loop-free, self-starting, and scales to large numbers of mobile nodes. AODV builds routes using a **route request / route reply** query cycle. When a source node desires a route to a destination for which it does not already have a route, it broadcasts a route request (RREQ) packet across the network. Nodes receiving this packet update their information for the source node and set up backwards pointers to the source node in the route tables. In addition to the source node's IP address, current sequence number, and broadcast ID, the RREQ also contains the most recent sequence number for the destination of which the source node is aware. A node receiving the RREQ may send a route reply (RREP) if it is either the destination or if it has a route to the destination with corresponding sequence number greater than or equal to that contained in the RREQ. If this is the case, it unicasts a RREP back to the source. Otherwise, it rebroadcasts the RREQ. Nodes keep track of the RREQ's source IP address and broadcast ID. If they receive a RREQ which they have already processed, they discard the RREQ and do not forward it. As the RREP propagates back to the source, nodes set up forward pointers to the destination. Once the source node receives the RREP, it may begin to forward data packets to the destination. If the source later receives a RREP containing a greater sequence number or contains

the same sequence number with a smaller hopcount, it may update its routing information for that destination and begin using the better route. As long as the route remains active, it will continue to be maintained. A route is considered active as long as there are data packets periodically travelling from the source to the destination along that path. Once the source stops sending data packets, the links will time out and eventually be deleted from the intermediate node routing tables. If a link break occurs while the route is active, the node upstream of the break propagates a route error (RERR) message to the source node to inform it of the now unreachable destination(s). After receiving the RERR, if the source node still desires the route, it can reinitiate route discovery. Multicast routes are set up in a

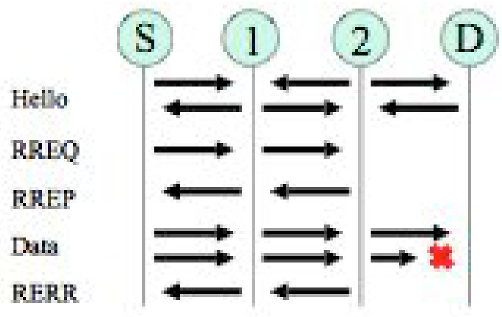

Figure 3.1. AODV messages

similar manner. A node wishing to join a multicast group broadcasts a RREQ with the destination IP address set to that of the multicast group and with the 'J'(join) flag set to indicate that it would like to join the group. Any node receiving this RREQ that is a member of the multicast tree that has a fresh enough sequence number for the multicast group may send a RREP. As the RREPs propagate back to the source, the nodes forwarding the message set up pointers in their multicast route tables. As the source node receives the RREPs, it keeps track of the route with the freshest sequence number, and beyond that the smallest hop count to the next multicast group member. After the specified discovery period, the source node will unicast a Multicast Activation (MACT) message to its selected next hop. This message serves the purpose of activating the route. A node that does not receive this message that had set up a multicast route pointer will timeout and delete the pointer. If the node receiving the MACT was not already a part of the multicast tree, it will also have been keeping track of the best route from the RREPs it received. Hence it must also unicast a MACT to its next hop, and so on until a node that was previously a member of the multicast tree is reached. AODV maintains routes for as long as the route is active. This includes maintaining a multicast tree for the life of the multicast group. Because the network nodes are mobile, it is likely that many link breakages along a route will occur during the lifetime of that route.

3.2 Passive Clustering

Passive Clustering [3] classifies ad-hoc and sensor nodes into critical and non crtitical nodes without any extra information. The proposed approach is to piggyback 2 bits of "cluster state" in MAC packet and monitorize the user traffic. Thus the cluster infrastructure can be constructed as a byproduct of user traffic, without any dependency on routing protocol. In Passive Clustering each node collects neighbour information from the MAC sender address carried by the incoming packets, and can construct clusters without collecting the complete neighbour list.

The possible state are four plus an internal state:

1. **Initial**
2. **Ordinary Node**
3. **ClusterHead**
4. **Gateway**

Passive clustering deploys the clustering structure fo free. Moreover, PC makes even the first flooding as efficient as all subsequent flooding and introduces many benefits including efficient flooding and density adaptation. As a result, PC reduces control overhead of ad hoc routing protocols significantly and, as a consequence, enables ad hoc routing in large, dense ad hoc/sensor networks. This kind of approach, virtually eliminates major source of cluster overhead like the time latency for initial clustering costruction as well as the comunication overhead for neighbour informations exchanges. It has seen that 2 bits of extra information are required. Such bits are stamped into a reserved field of the packet header. The sender ID carried by all the existing MAC protocols is also required, and it can be retrivied from the MAC header without any extra burden. Since the Mac packets are trasmitted in broadcast mode flooding, every node receives and reads the packets (in a promiscuos way) and, thus,partecipates in Passive Clustering.

To improve clustering stability, to speed up convergence, and most importantly, to avoid the stationarity requirement during the neighbour-learning and clustering phases, in (Gerla Work) a new clusterhead election rule is proposed which does not require any weight information [3].

This type of election is called "First Declaration Wins" and it works as follows:

a node which first claim to be the clusterhead remains the clusterhead and "rules" the rest of the node in its clusterd area (radio coverage). There is no waiting period to make sure all the neighbour have been checked unlike from all the weight-driven clustering mechanisms. Finally, a heuristic control for the gateway's selection is also defined.

Operational Description

Based on Aodv when a source send a RREQ all the neighbours are ready to be a CH. When a neighbour processes the route request and has packets to send, it declares itself as clusterhead by timestamping its clustering state claim in the packets. After a successful transmission from an aspiring CH, every node within the range of coverage learns the presence of the cluster-head by monitoring the "cluster" state of the received packets. At this point, the neighbours of the clusterhead record the clusterhead information and change their clustering state.
Cluster-head information is composed of CH ID and most recent transaction time timestamp. So the readiness of being a clusterhead is determined by network activities as well as by the node's clustering state.

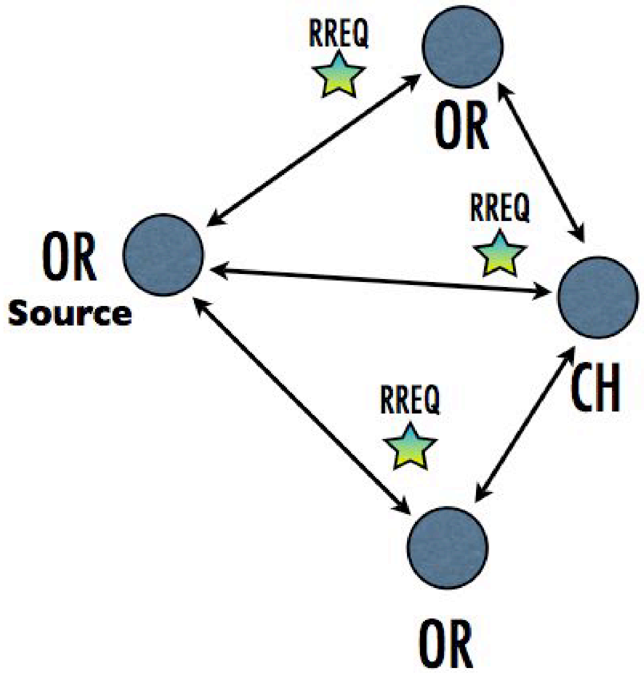

Figure 3.2. ClusterHead's Election

When a non-clusterhead node hears a packet from a clusterhead or a gateway it could be elegible as gateway according to the following rule:

$$\alpha * NC + \beta > NG$$

where:

- α is a coefficent properly chosen based on the desired degree of gateway redundancy
- β is a gateway redundancy factor
- NC is the number of ClusterHeads
- NG is the number of Gateways

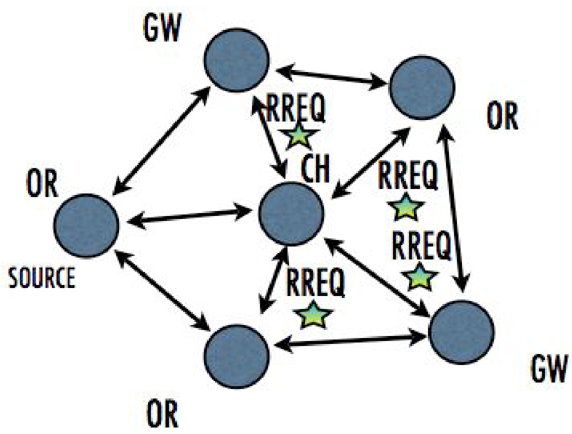

Figure 3.3. Gateway's Election

After a period of inactivity (no incoming or outgoing traffic for longer than the cluster timeout period) all the nodes revert to the initial state. Only nodes in initial state can be a clusterhead thus the minumum distance between any two clusterhead is two hops.
Clusterheads collect neighbour information by monitoring the network traffic. They

are responsibile for relaying intracluster packets. A node that hears more than clusterhead becomes a gateway. It reverts to ordinary node only if it does not hear from more than one clusterhead for a given period. Figure 3.4 presents the possible states of node.

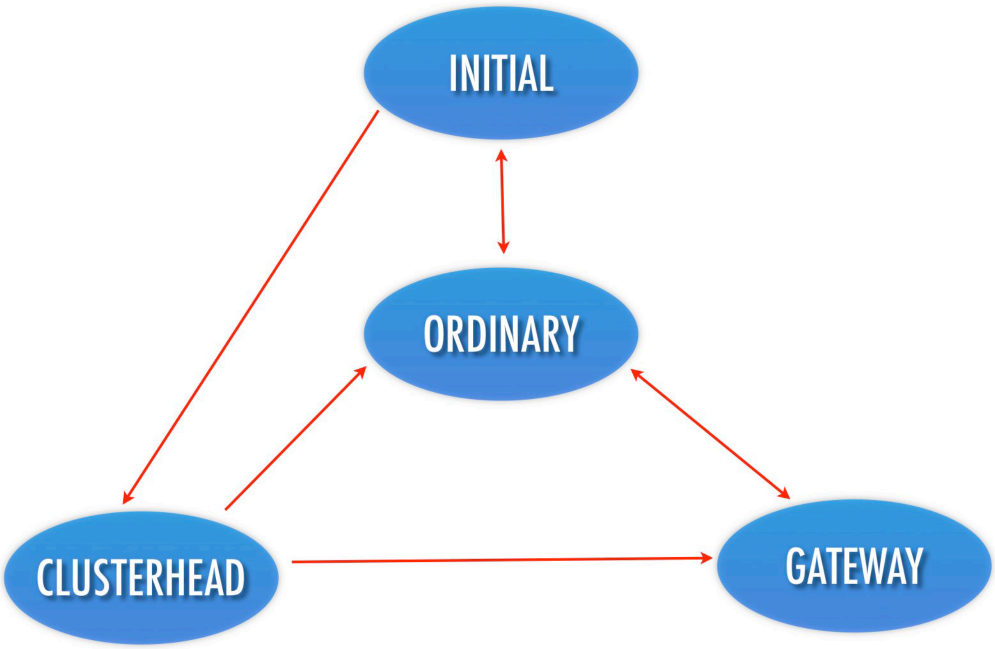

Figure 3.4. Cluster's possible state

3.3 Innovative contribution

In the original formulation of Passive Clustering the gateway's election procedure was not fully specified. Performance evaluation [3] was based on low traffic scenario,

and the simulations were carried out on short simulation time (ten minutes). Furthermore, selection criteria for the two parameters α and β were not provided.
In this thesis the above issues are investigated, leading to an improved version of the Passive Clustering protocol. Furthermore, an additional parameter is taken into account: the energy state of the node. This aspect was not covered in the original Pc, since this was only focusing on the idea of "Low Maintenance". The Passive Clustering is proposed as a modification of Aodv structure, (Aodv Pc) with the evaluation of energy of the nodes.
The "First Declaration Wins", originally proposed for clusterhead selection [3] was also introduced for gateway's election but with a key innovation: the introduction of energy.
The key idea behind this approach is the fact that an energy aware selection of clustehead and gateway can lead to a longer lifetime and improved network performance. Also a new rule for Passive Clustering was implemented: "the total coverage". This rule was introduced to avoid the chance of connectivty loss in the adhoc/sensor network, caused by the Passive Clustering standard criteria. Infact in small networks with few pairs of Source-Destination, the flooding search for the routes using as relay clusterheads and gateways, selected by Pc criteria, could not give cover the whole network. The new rule was based on following idea: if a node receive from a gateway and doesn't hear any clusterhead around it, it declares itself as a clusterhead. In table 3.1 were shown the possible states of the nodes in standard Passive Clustering. The "total coverage" has modified the original table as follows in 3.2

The energy parameter **E** was calculated taking in account the Friis formulation for free space transmission and the following model [31]:

$$E_{TX} = E_{Start} + L \cdot (E_{TxBitRate}(R_b) + E_{TxBitProp}(R_b) \cdot d^{\alpha})$$
$$E_{RX} = E_{Start} + L \cdot (E_{RxBitFixed} + E_{RxBitRate}(R_b)$$

where L is the length of the packet.

MY STATE	RECEIVED STATE	MY NEW STATE
OR	OR	IN
	-	-
	CH	GW o OR
	GW	OR
IN	OR	IN
	-	-
	CH	GW o OR
	GW	OR
CH	OR	CH
	-	-
	CH	GW o OR
	GW	CH
GW	OR	GW
	-	-
	CH	GW
	GW	GW o OR

Table 3.1. Clusters possible state

In the original formulation of Passive Clustering timers for the elections are chosen using a random value comprised into an interval [*Start Timer,Finish Timer*] (for example a random number between 1 second and 4 second), with average time equal to $MeanValue$ for all nodes.

The interval for the selection of the timer can be made different for each node by calculating a new Mean Value for the interval. In our approach energy was taken in account as follows. The actual random interval is defined on the basis of residual energy of the node, according to the following definitions:

$$f(E) = \frac{1}{E(\%)}$$

$$MeanValue = CONST \cdot f(E) - StartTimer$$

MY STATE	RECEIVED STATE	MY NEW STATE
OR	OR	IN
	–	–
	CH	GW o OR
	GW	CH o OR
IN	OR	IN
	–	–
	CH	GW o OR
	GW	OR
CH	OR	CH
	–	–
	CH	GW o OR
	GW	CH
GW	OR	GW
	–	–
	CH	GW
	GW	GW o OR

Table 3.2. New clusters state

$$FinishTimer = 2 \cdot MeanValue - StartTimer$$

♠ CONST is costant fixed at the value 1

Also a quadratic version of the formulation below has been proposed with:

$$f'(E) = \frac{1}{E(\%)^2}$$

$$MeanValue = CONST \cdot f'(E) - StartTimer$$

$$FinishTimer = 2 \cdot MeanValue - StartTimer$$

In figure 3.5 it is presented an example of the average seconds for clusterhead's or gateway's election with the above formulations. The second formulation is adopted

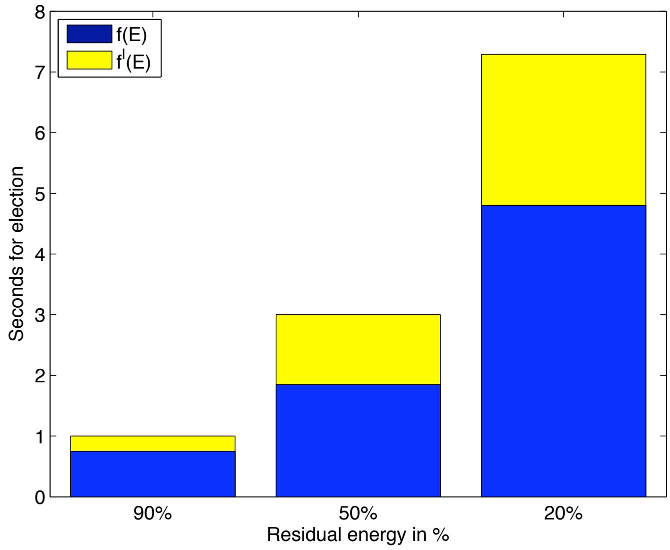

Figure 3.5. Average Seconds for Ch or Gw selection

for the simulation in chapter 4, because it perfoms better than the first one. In fact in this way the probability to select a node with low power during clustering procedure is reduced.

So if a node candidate to become a clustehead or a gateway has low energy, it sets its timer with a large value and thus it has less chances to be elected. Thus the topology of the entire adhoc/sensor network is driven and based on energy.

Chapter 4

Performance evaluation

The performance of the proposed clustering algorithm was carried out by computer simulation. Section 4.1 provides details on the simulation environment, while section 4.2 presents simulation results. Section 4.3 discussed and comments the results.

4.1 Simulaton Environment

Performance evalutation was carried out in the framework of Omnet++ version 3.4b2 simulator, in particular by using Mobility Framework package under Mac Os operating system. sssThe simulator describes an ad hoc network with a parameterizable number of hosts that move in a playground free of obstacles. Each host has a defined transmission power. Figure 4.1 shows a screenshot of the simulator graphical output. Each mobile host in the network is an Omnet++ compound module which encapsulates the following simple modules:

1. Application Module
2. Network Module
3. Route Module

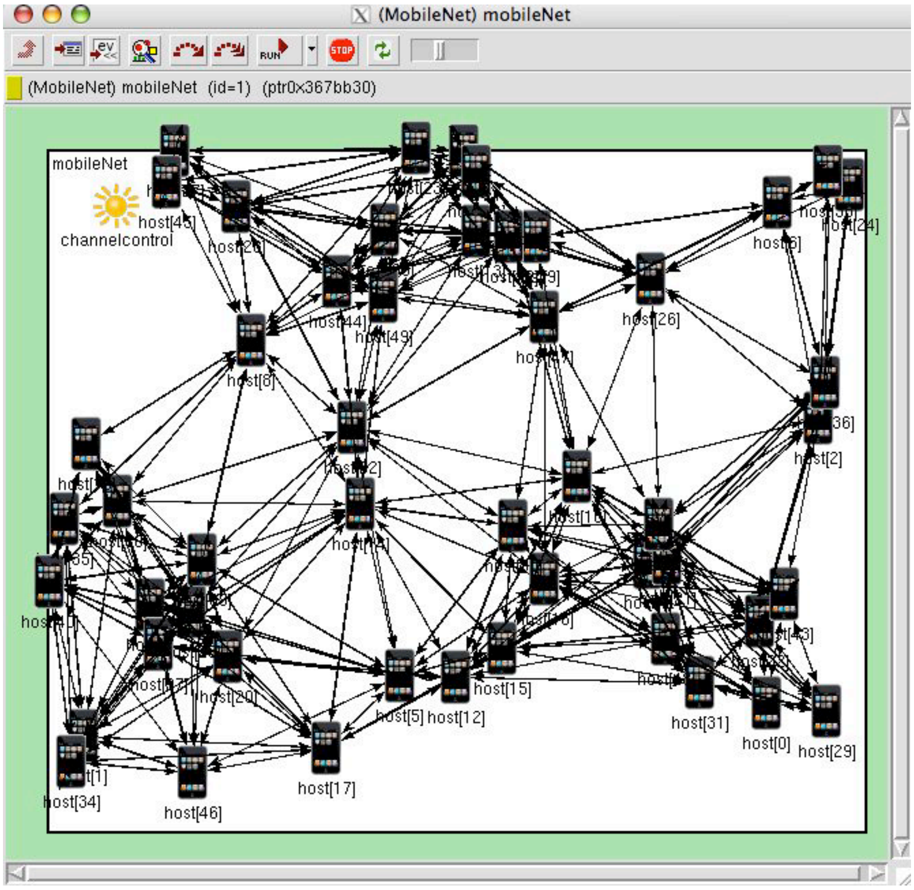

Figure 4.1. Simulator screenshot

4. Nic Module composed of Mac Module and Physical Module

Note that this structure is different from the original one (fig 4.2) proposed in the Mobility Framework since the routing module was not present and it was introduced from scratch in this work (fig 4.3). The next section provides key details on each module.

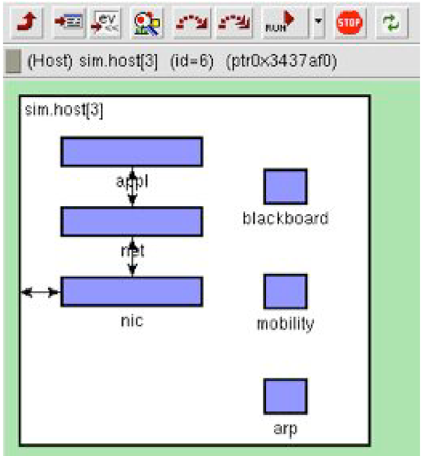

Figure 4.2. Orginal MF mobile node architecture

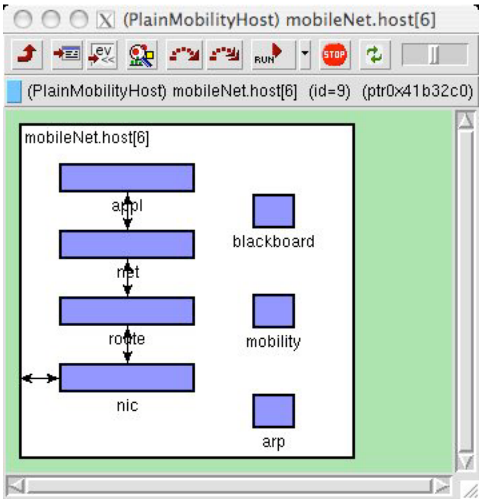

Figure 4.3. Implemented mobile node architecture

4.1.1 Modules specification

Main settings of the simulator are based on standard 802.15.4[7]. Infact both MAC layer and PHYSICAL layer of the simulator are based on it.

Let see in the specific layer by layer starting from the network scenario. For the network environment a $3km^2$ playground has been choosen, in which a variable number of host, from 10 to 50 nodes, exchanges traffic control and data for one hour. The values are in the table 4.1.

Network Environment

Playground size	600x500 meters
Number of Hosts	Variable 10 to 50
Simulation Time	1h

Table 4.1.

The physical module and mac module are carried on in the block called Nic (Network Inteface Card). The physical module uses some parameter described in the [7] like

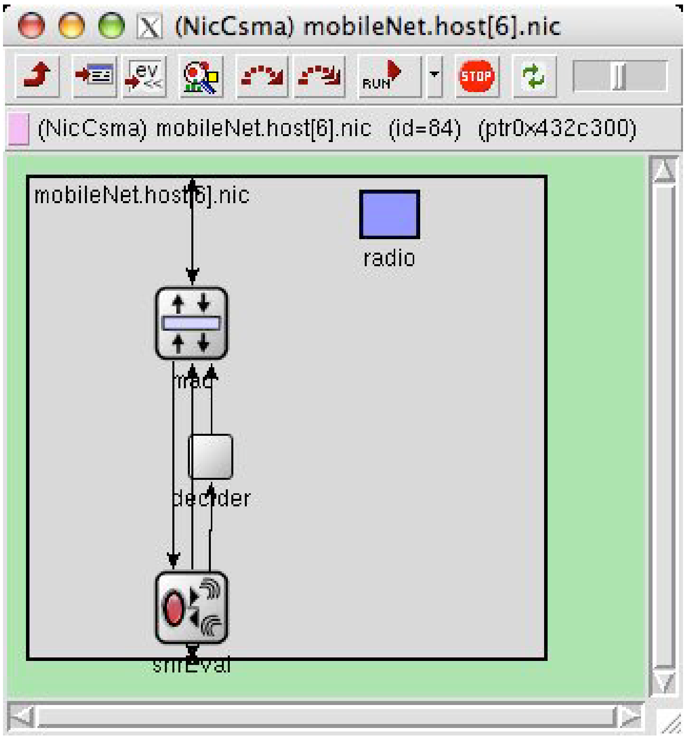

Figure 4.4. Nic Layers

the carrier frequency, and the header length, and it is created using two additional sub-modules:

SnrEval and Decider as shown in fig A.3. The SnrEval module simulates a transmission delay for all received messages and also calculates SNR information. Additionally the SNR information of all other messages in the receive buffer should be updated if desired. Afterwards the message is buffered (function bufferMsg()) to simulate a transmission delay. During this time other messages may arrive which would interfere with the buffered message and thus may result in additional SnrList entries to indicate a change in SNR for this message. In this submodule the energy evaluation in introduced. The Decider module only processes messages coming from the channel (for example from lower layers). Messages coming from upper layers bypass the Decider module and are directly handed to the SnrEval module. Decisions about bit error or lost messages only have to be made about messages coming from the channel. Consequently there is no need to process messages coming from upper layers in the Decider module. In table 4.2 are shown the fixed values for the Physical Module (SnrEval plus Decider).

Physical Module

Carrier Frequency	2.4 Ghz
Trasmitted Power	10 mW
Thermal Noise	-120 dBm
Threshold Level	10 dB
Header Length	64 bit

Table 4.2.

The MAC module is based on Carrier Sense Multiple Access (CSMA). Sensing is done using a detection fo radio states. So if the channel is "free" messages are sent, instead if the channel is "busy" the messages are bufferized and put in queue. The bitrate and header length used are the same seen for 802.15.4.

Mac Module

Bitrate	250 Kb/s
Header Length	104 bit
Inter-arrival Time	0.006 s
Queue Length	1 MB

Table 4.3.

The routing and Clustering Module is the core of entire project. It is based on Passive Clustering protocol [3] and Aodv Perkin's draft [15] and finally modified with energetic approach. It is based on the Adov implementation done by Nicola Concer [32], and adapted for the Mobility Framework. Thus, in table 4.4 are showed the most important parameter for the module.

Routing and Clustering Module

Control Message Size	512 bit
Hello Interval	11 sec
Allowed HELLO loss	2 sec
Delete Period	4 sec
ACTIVE ROUTE TIME OUT	30 sec
α	1
β	1
Clusterhead Life Period	for 5 m/s is 3 s
Gateway Life Period	for 5 m/s is 3 s

Table 4.4.

The network module is the same of traditional MF without any alteration. Only the header length has been changed as shown in table 4.5

Network Module

Header Length	64 bit

Table 4.5.

The application module is responsible for the traffic generation.
The traffic generation is done using an "exponential" function to select timers for new connections (connection interarrival mean period) and to create number of

packets (packet average). So, connections that enable a node to transmit are created. Also the remaining packets to send are taken in account; indeed for each connection the entire number of created meassages have to be sent before the establishment of a new connection.

Application Module

Data Payload	748 bit
Header Length	64 bit
Host enable to Transmit	All
Packet Average	variable 10,500,1000
Connection Interarrival Mean Period	10 sec

Table 4.6.

Mobility module is another important aspect. In Mass Mobility Model a mobile host moves within the room according to the following pattern. It moves along a straight line for a certain period of time before it makes a turn. This moving period is a random number, normally distributed with average of 5 seconds and standard deviation of 0.1 second. When it makes a turn, the new direction (angle) in which it will move is a normally distributed random number with average equal to the previous direction and standard deviation of 30 degrees. Its speed is also a normally distributed random number, with a controlled average, ranging from 0.1 to 0.45 (unit/sec), and standard deviation of 0.01 (unit/sec). A new such random number is picked as its speed when it makes a turn. This pattern of mobility is intended to model node movement during which the nodes have momentum, and thus do not start, stop, or turn abruptly. When it hits a wall, it reflects off the wall at the same angle.

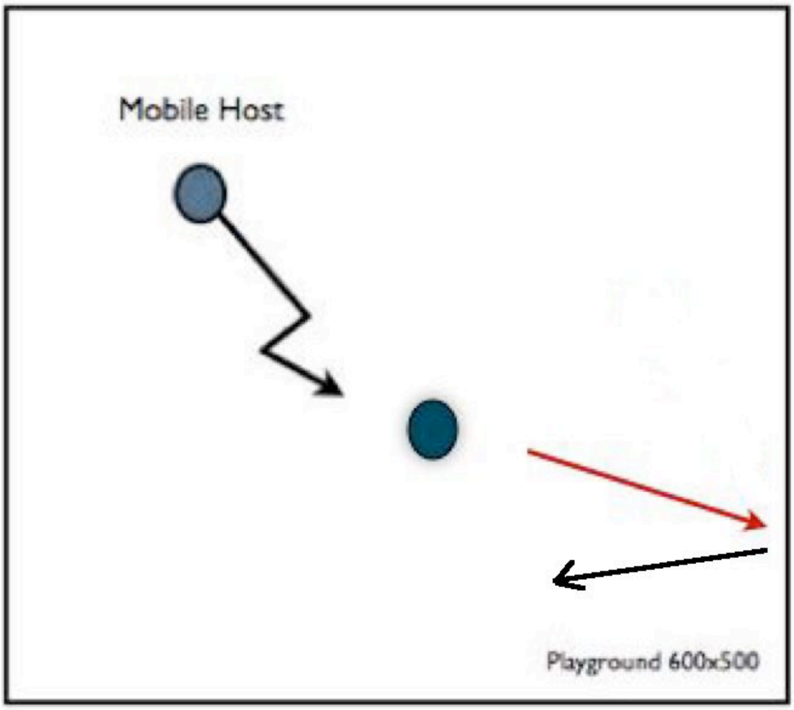

Figure 4.5. Mass Mobility Model

4.2 Simulation Results

As it has been seen, the work is focused at different levels of the modules, proposed in the Mobility Framework. The simulation is done with the fixed paremeters shown in the previous section, and two new variable key parameter:

1. Traffic
2. Mobility

4.2.1 Modellation of Traffic Classes

At first, simulation was on modelling the traffic classes in order to obtain:

- Low Traffic Scenario → Throughput about 90%

- Medium Traffic Scenario → Throughput about 60%
- High Traffic Scenario → Throughput about 30%

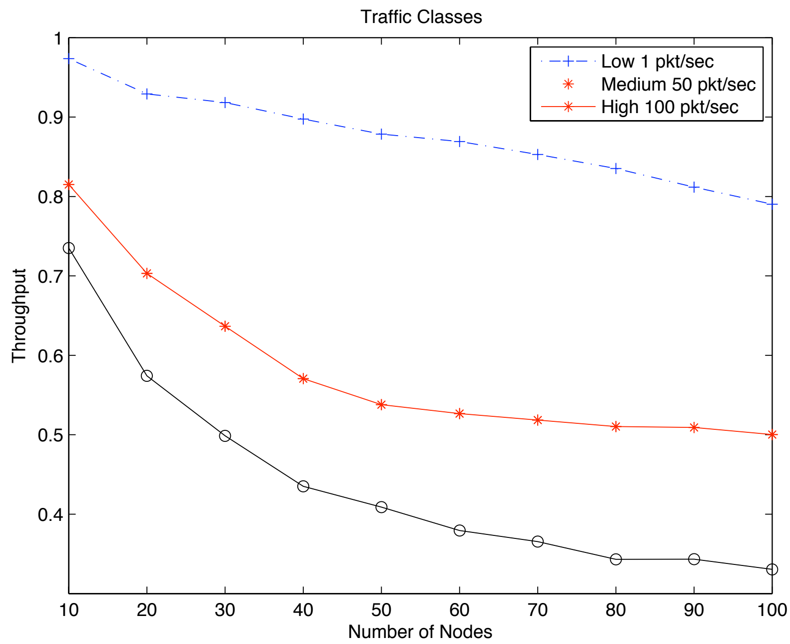

Figure 4.6. Traffic Classes

Thus simulations to find these values without introducing any type of mobility were investigated, and the results are in figure 4.6

Offered Network Traffic	Pkt/Sec
Low	1
Medium	50
High	100

Table 4.7.

Then the residual energy for the traffic classes has been plotted in figure 4.7. After those results, medium traffic class model has been selected and Aodv Pure (without

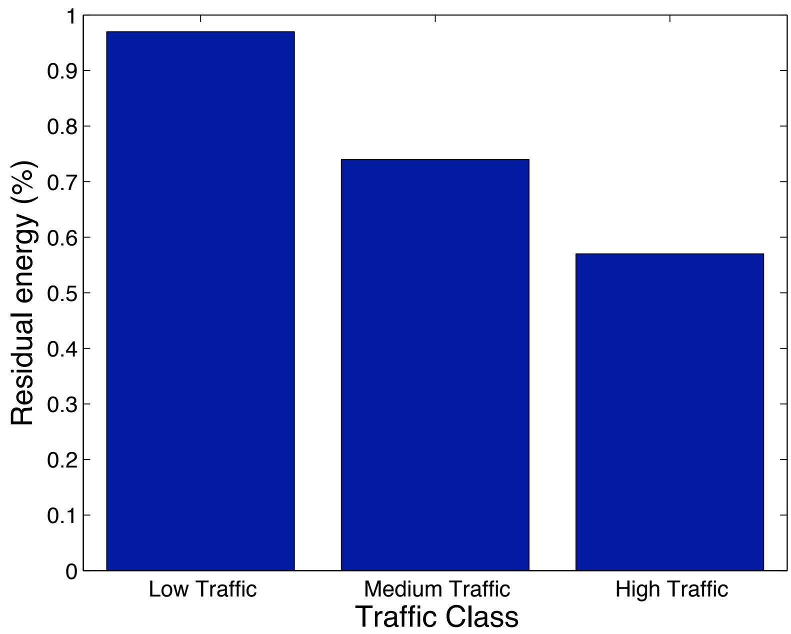

Figure 4.7. Residual Energy

any alteration) with the Mass Mobility model has been introducted.

4.2.2 Introduction of Mobility

In Mass Mobility Model a mobile host moves within the room according to the following pattern. It moves along a straight line for a certain period of time before it makes a turn. Utilizing the proposed model throughput and total sent control packets (RREQ, RREP, RERR, ACK) have been evaluated and plotted in figure 4.8 and 4.9, taking into account different speeds of:

- Low Speed → 2 m/s
- Medium Speed → 5 m/s
- High Speed → 10 m/s

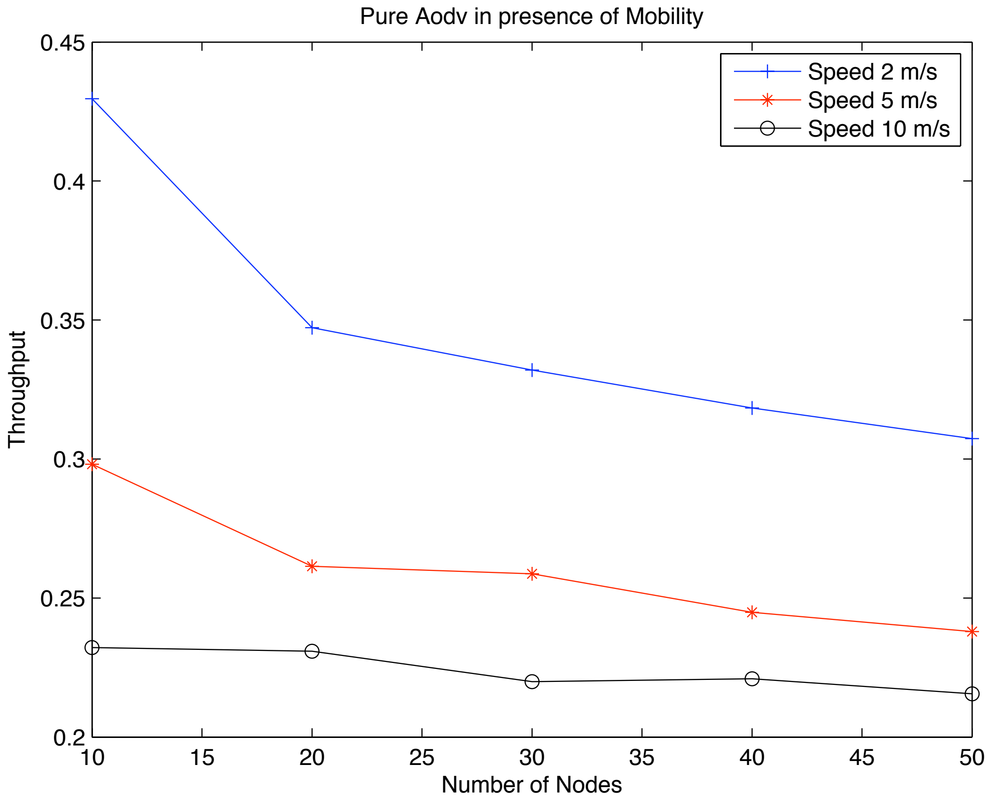

Figure 4.8. Throughput in presence of Mass Mobility

4.2.3 Clustering implementation

The Passive Clustering scheme was created inside the routing and clustering module of the Omnet++ stack. Thus, a new part of code that can select clusterhead and gateway based on the reception of a Route Request has been built-up. Each Route Request has a new field that carries on the "Cluster State" of the node. Using the table 4.8 all the possibles transition through the states have been discriminated and implemented on the simulator:

Both the election of clusterhead and gateway has been simulated according to the "First Declaration Wins" rule. Thus, to implement the "FDW" different timers

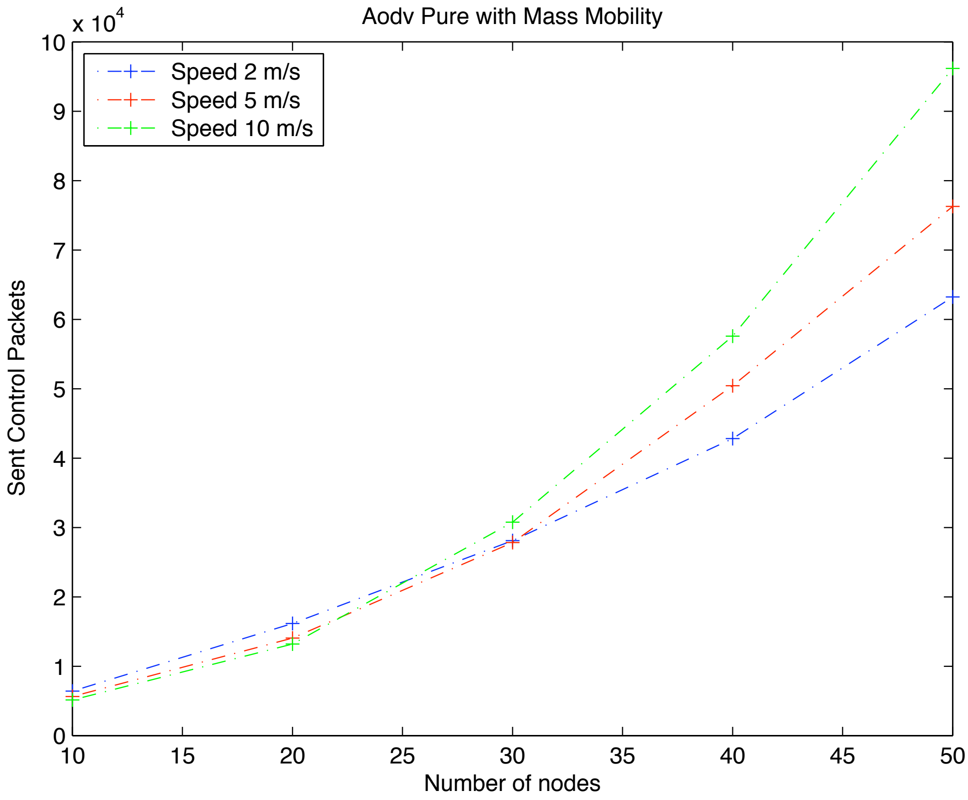

Figure 4.9. Sent Control Packets in presence of Mass Mobility

for the elections have been selected using the "uniform" function according to the Aodv time outs (for example the ACTIVE ROUTE time out). Also the gateway heuristic selection has been implemented setting α e β at values of 1 e 1. Finally the new rule for the "total coverage" of the network has been implemented. Infact in some simulation, without mobility and only one pair of Source-Destination, a coverage failure has been detected. For this reason the new "total coverage" rule has been added.

MY STATE	RECEIVED STATE	MY NEW STATE
OR	OR	IN
	-	-
	CH	GW o OR
	GW	CH o OR
IN	OR	IN
	-	-
	CH	GW o OR
	GW	OR
CH	OR	CH
	-	-
	CH	GW o OR
	GW	CH
GW	OR	GW
	-	-
	CH	GW
	GW	GW o OR

Table 4.8. New clusters possible state

4.2.4 Pure Aodv vs PC Aodv

The efficiency of Passive Clustering solution versus Pure Aodv has been compared in this scenario:

- Number of nodes between 10 and 50
- Medium traffic of 50 pkt/sec

- Average speed of 5 m/s

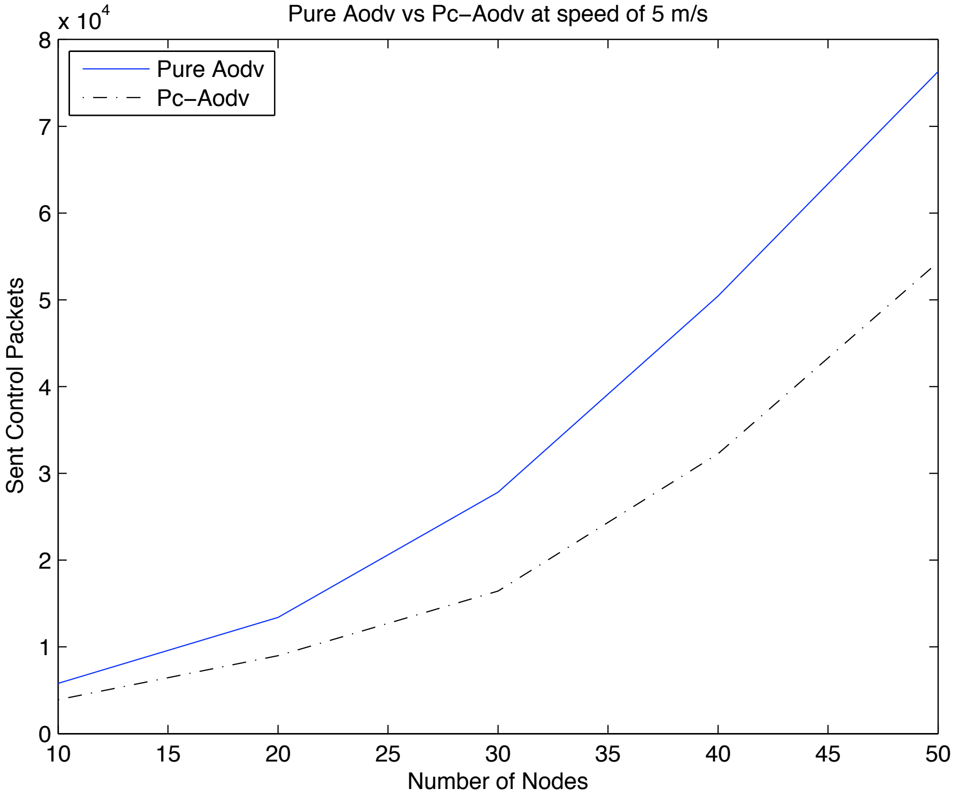

Figure 4.10. Sent Control Packets Pure vs Pc

The first comparision shown in figure 4.10 between Pure Aodv versus Passive Clustering Aodv has been done on the number of sent control packets. The second confrontation in figure 4.11 has been plotted analizing the throughput. In figures 4.12 and 4.13 energy utlization per node and energy consumption per data packet has been analized and plotted.

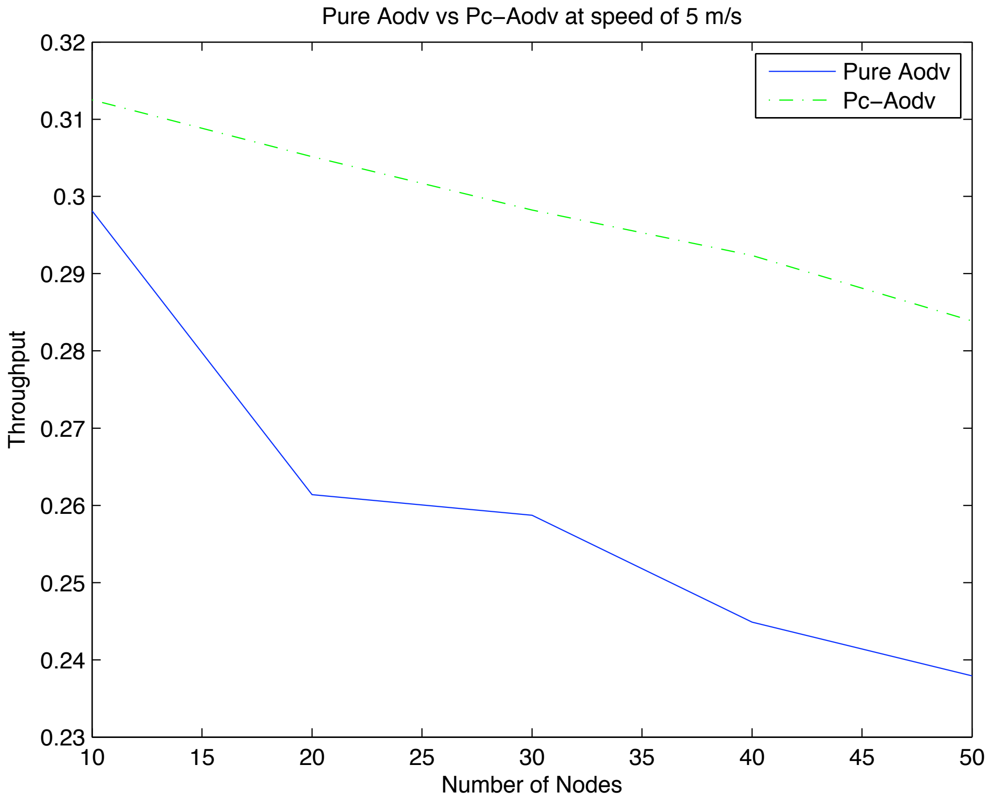

Figure 4.11. Throughput Pure vs Pc

4.2.5 Passive Clustering vs Energetic Passive Clustering

After Passive Clustering achievement, energy parameter has been introducted. The energy parameter **E** has been calculated taking in account the Friis formulation for free space transmission and the following model [31]:

$$E_{TX} = E_{Start} + L \cdot (E_{TxBitRate}(R_b) + E_{TxBitProp}(R_b) \cdot d^{\alpha})$$
$$E_{RX} = E_{Start} + L \cdot (E_{RxBitFixed} + E_{RxBitRate}(R_b))$$

where L is the length of the packet.
The energy has been utilized to set the clusterhead or gateway desirability. Infact the idea (illustrated in chapter 3) was to enlarge the interval for the clusterhead and

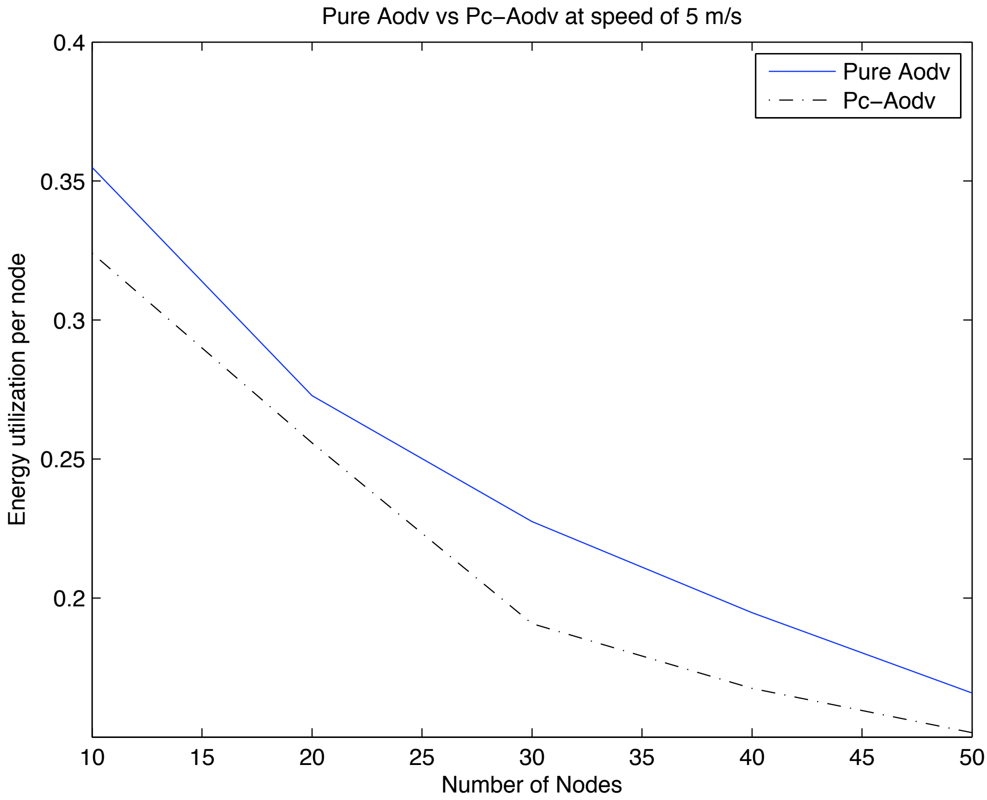

Figure 4.12. Energy utilization per node in Pure and PC

gateway election. In this way the nodes with low energy have less probability to win the election and became clusterhead or gateway. The implemented steps to enlarge the election interval [StartTimer , FinishTimer] are:

$$f'(E) = \frac{1}{E(\%)^2}$$

$$MeanValue = COST \cdot f'(E) - StartTimer$$

$$FinishTimer = 2 \cdot MeanValue - StartTimer$$

CONST is a fixed value (in this case is set to 1). StartTimer is a fixed value (for the simulation is set to 0.2 according to Aodv time outs).

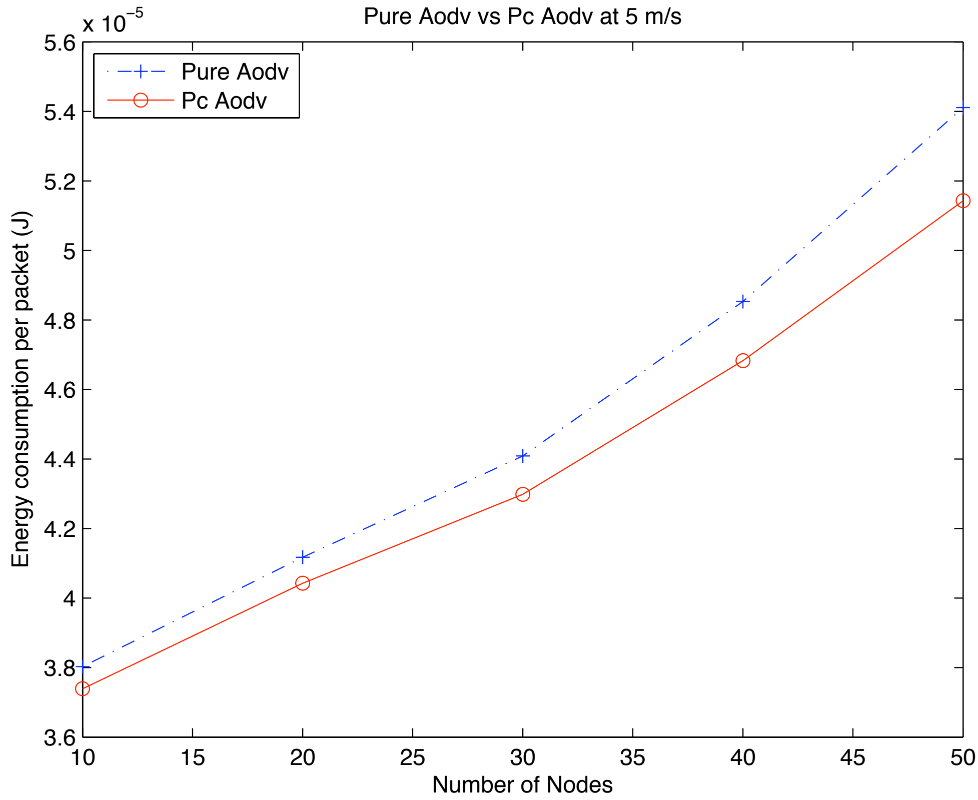

Figure 4.13. Energy consumption per packet in Pure and PC

Once the modified Passive Clustering (Passive Clustering 2.0) has been implemented, a comparison with the standard Passive Clustering has been done. The scenario utlized was the same of the Pure Aodv vs PC Aodv, with medium class traffic and speed of 5 m/s. In figure 4.14 and 4.15 throughput and sent control packets have been evaluated. Finally, energetic confrontation utilizing the standard deviation has been investigated and plotted. In the figure 4.16 the standard deviation of residual energy of the nodes has been evaluated with the increasing of the number of nodes. In the last figure 4.17 the scenario of simulation is based on a number of fixed node (50) that moves at the speed of 5 m/s with different traffic classes.

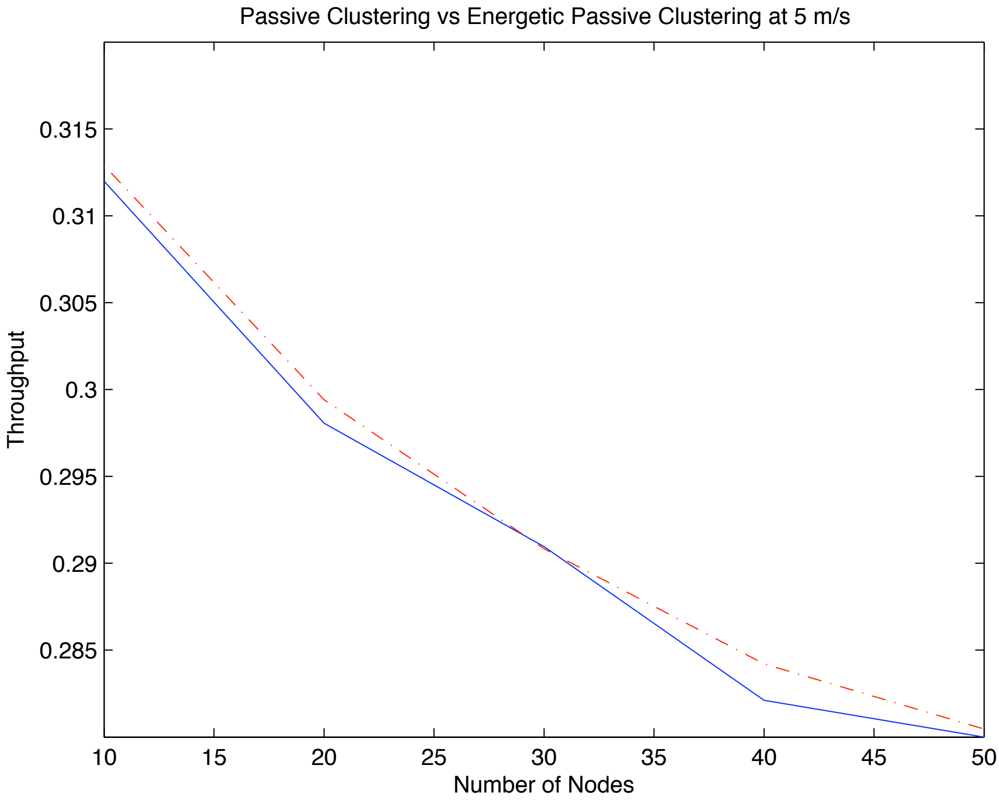

Figure 4.14. Throughput PC (blue line) vs PC 2.0 (red dotted line)

4.3 Discussion of results

Several simulations have dimostrated the validity of Passive Clustering solution utilized with a reactive routing protocol like Aodv.

PC is a self sufficient clustering scheme that according to its passive nature reduce the flooding without extra overhead, and creates a dynamic structure in presence of mobility. It's really interesting the 30% reduction of control packets shown in figure 4.10. This means that the throughput is increased, and it can reach a 5% of gain as seen in figure 4.14.

Also the estimation of energy was considered. In fact with Pure Aodv there is a

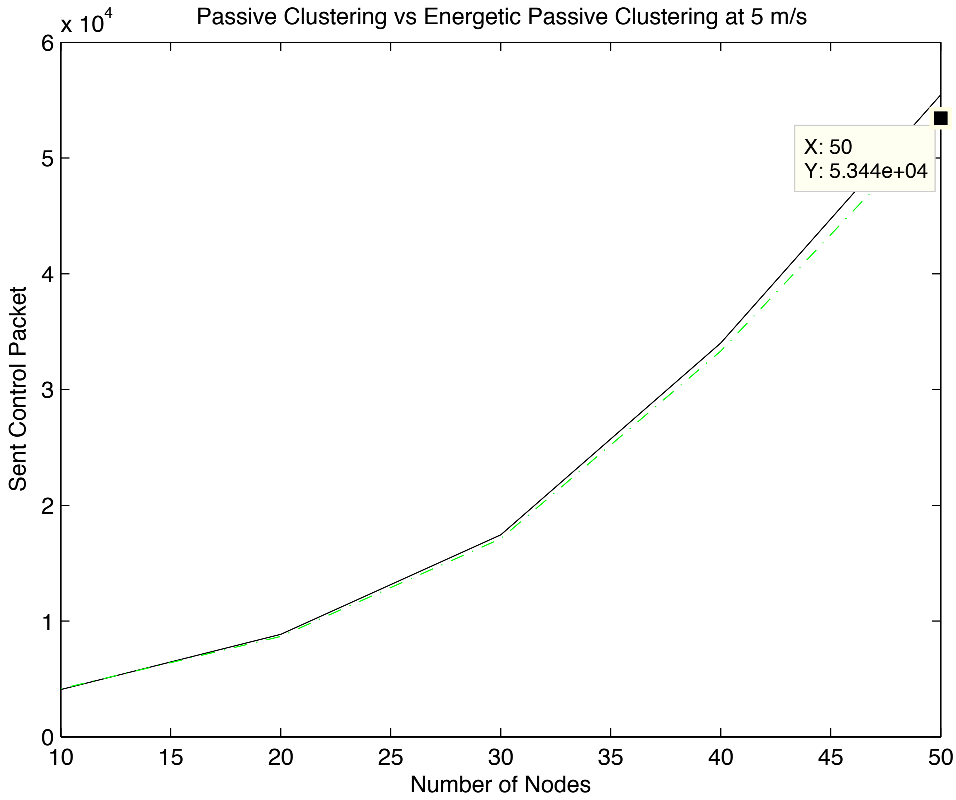

Figure 4.15. Sent Control Packets in PC (black line) and PC 2.0 (green dotted line)

greater consumption of energy per packet (figure 4.13) and a minor utilization of energy per node (figure 4.12). With the introduction of the new energetic rule that decreases the probability to elect as a clusterhead or a gateway a node with low energy, the standard Passive Clustering is improved. The energetic implementation gives better performance in terms of:

1. throughput
2. sent control packets
3. energy

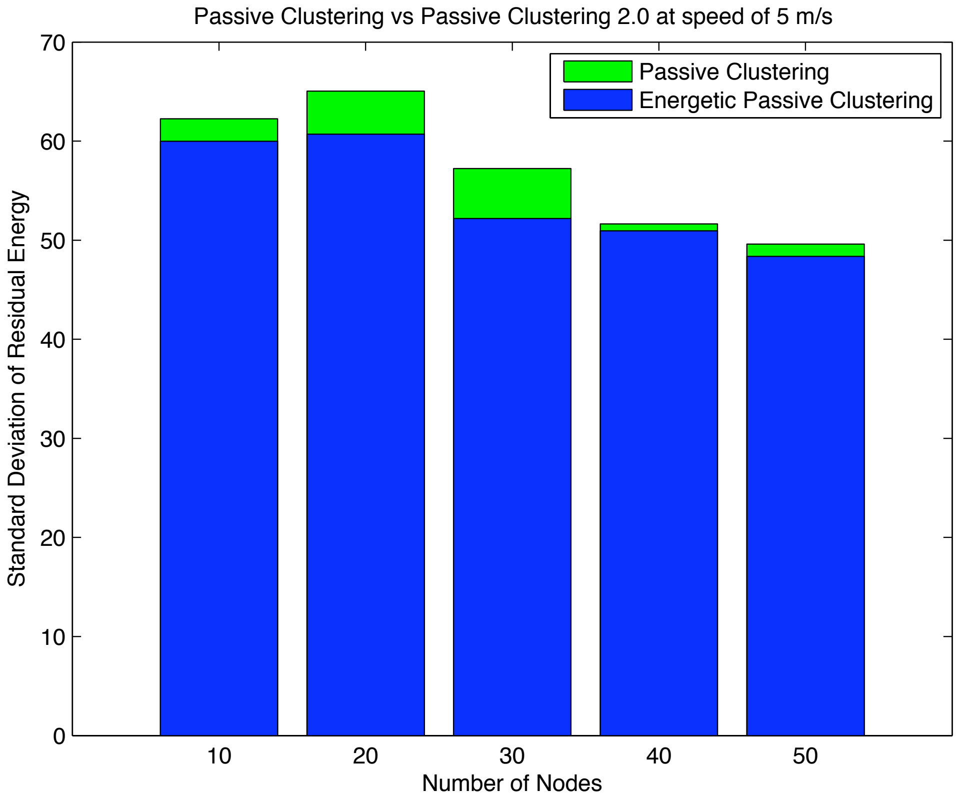

Figure 4.16. Standard deviation with node increasing PC and PC 2.0

As shown in figure 4.14, there is little but not negligible improvement of throghput in comparison with the standard passive clustering. This is reflected also in figure 4.15 that shows a reduction of control packets versus the increasing number of hosts. Finally, the energetic evaluation based on standard deviation demonstrates that the Pc 2.0 has a reduced energy dispersion for different network topology and traffic classes.

Infact after extensive simulation it has seen that the energy dispersion of Pc 2.0 is always lower than the classic Pc.

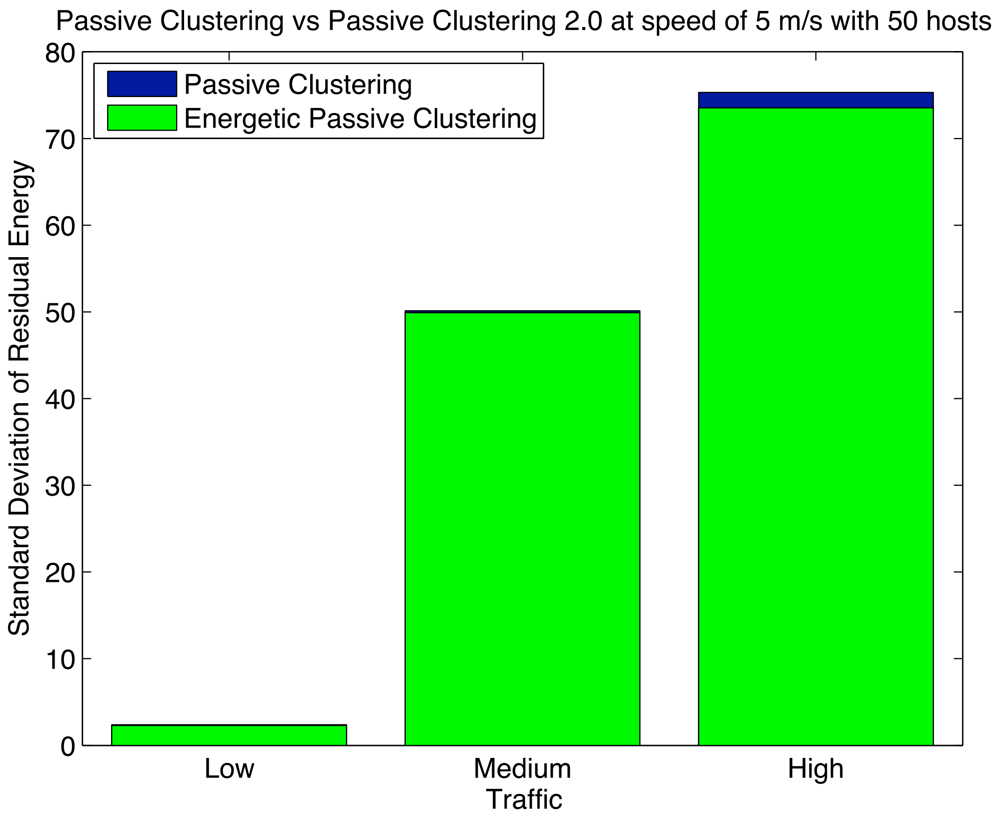

Figure 4.17. Standard deviation with different traffic classes for PC and PC 2.0

Chapter 5

Conclusions and future works

In this work Passive Clustering technique has been implemented, evaluated and improved. Results show that this technique can reduce overhead caused by indiscriminated flooding on the network. Passive Clustering provides a clustering structure without additional overhead (only 2 bit of extra information). An additional performance increase was obtained by introducing energy awareness in clusterhead's and gateway's selection. This result could be achieved by determining the CH and GW selection timers as a function of node energy. Interestingly, the performance gains were provided by the energetic strategy, obtained completely for free, as no additional information was introduced.
The work carried out in this thesis forms the basis for several future research lines. They are briefly suggested in the following:

1. Tuning of timers and time out

2. Introduction of tables of connectivity in clusterhead and gateway evaluation, in order to determine a Dominant Set of CHs and GWs

3. New selection strategies for α and β

As a first follow up of this thesis careful investigation of tuning of timers for the elections and lifetime of clusterhead and gateways could lead to larger performance gains.
It should be noted that the introduction of new features in Passive Clustering can require additional overhead. For example the introduction of information connectivity may require additional traffic for exchanging connectivity tables between nodes. This overhead must be taken into account in performance evaluation because could lead to an overall reduction in throughput and energy. The impact of reducing the set of CHs and GWs to a dominant set should also be investigated, as it is not guaranteed that this would lead to improved performance.
Finally, parameters like α and β, that were kept constant in this thesis, could be modelled in future works by using different variables like energy, inteference, or connectivity, keeping in mind the above considerations on overhead. Those two parameters could be seen in a cognitive scenario, for example a dense network, where some nodes set a value for α and β, and other group of nodes set another pair of values α' and β' as shown in fig 5.1.

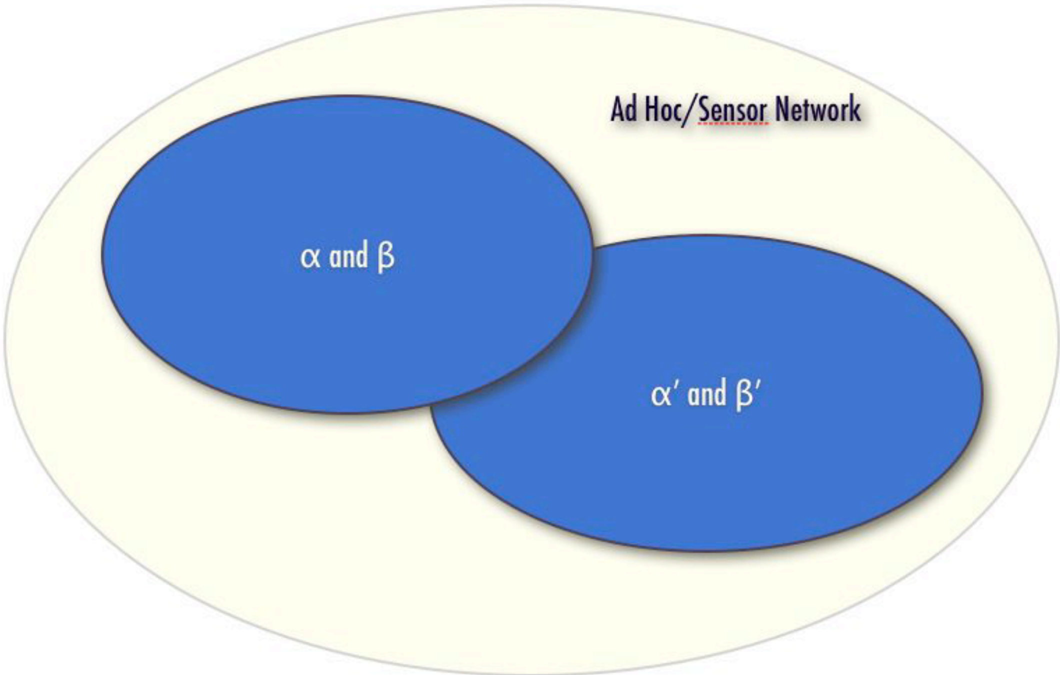

Figure 5.1. Cognitive Scenario

Appendix A

Mobility Framework

This framework [33] is intended to support wireless and mobile simulations within OMNeT++. The core framework implements the support for node mobility, dynamic connection management and a wireless channel model. Additionally the core framework provides basic modules that can be derived in order to implement own modules. With this concept a programmer can easily develop own protocol implementations for the Mobility Framework (MF) without having to deal with the necessary interface and interoperability stuff.

The two core components of the Mobility Framework (MF) are an architecture for mobility support and dynamic connection management and a model of a mobile host in OMNeT++. The figure below shows a network setup with 10 nodes. The ChannelControl module controls and maintains all potential connections between the hosts. An OMNeT++ connection link in the MF does not automatically indicate that the corresponding hosts are able to exchange data and communicate with each other. The ChannelControl module only connects all hosts that possibly interfere with each other. A communication link is probably easiest defined by its complement: All hosts that are not connected definitely do not interfere with each other. Following this concept a host will receive every data packet that its

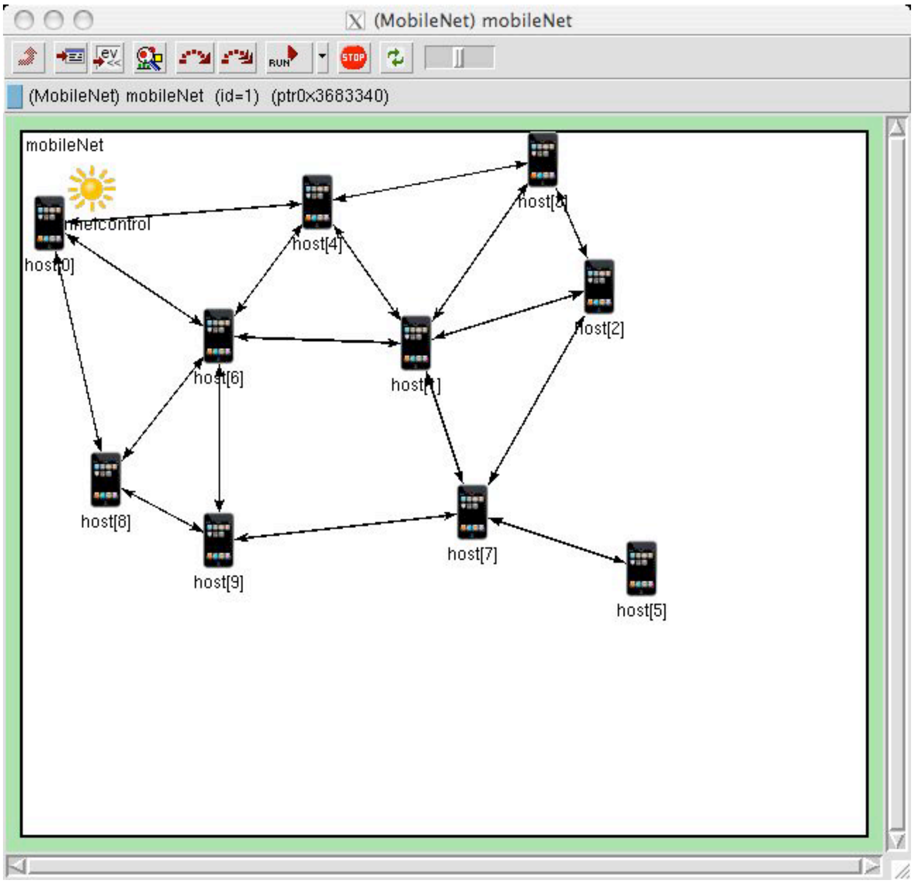

Figure A.1. MF Network Initialization

transceiver is potentially able to sense. The physical layer then has to decide dependent on the received signal strength whether the data packet will be processed or whether it will be treated as noise. Apart from the standard ISO/OSI layers there is also a Mobility module and a module called Blackboard. The Mobility module provides a geographical position of the host and handles its movement.

The Blackboard module is used for cross layer communication. It provides information relevant to more than one layer like the actual energy status of the host, the display appearance or the status of the radio. All other modules implement the corresponding ISO/OSI under layer functionality.

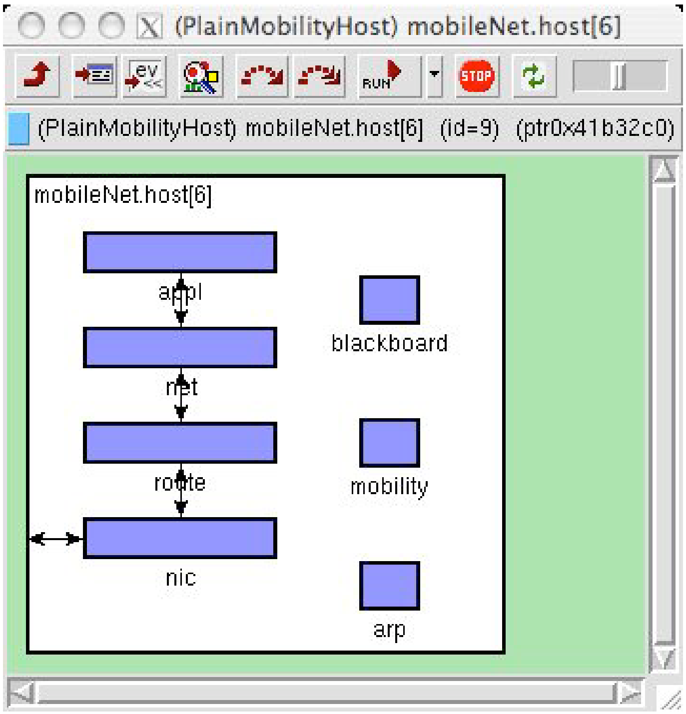

Figure A.2. Layers ISO/OSI

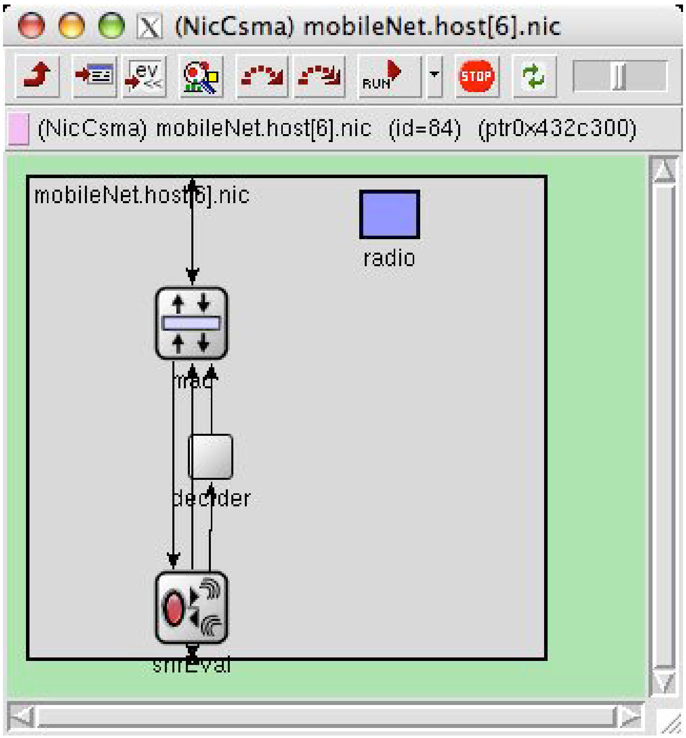

Figure A.3. Nic Layers

Bibliography

[1] [Online]. Available: http://en.wikipedia.org/wiki/Wireless

[2] [Online]. Available: http://en.wikipedia.org/wiki/Dataclustering

[3] T. J. Kwon, M. Gerla, V. K. Varma, M. Bourne, and T. R. Hsing, "Efficient flooding with passive clustering- an overhead free selective forward mechanism for ad hoc sensor networks," *IEEE*, 2003.

[4] J. Jubin and J. Tornow, "The darpa packet radio network protocols," *Proceedings of the IEEE*, vol. 75, no. Issue 1, pp. 21–32, Jan 1987.

[5] S. Papavassiliou, S. Xu, P. Orlik, M. Snyder, and P. Sass, *Scalability in Global Mobile Information Systems (GloMo): Issues, Evaluation Methodology and Experiences*, vol. 8, no. 6, pp. 637–648, November 2002.

[6] R. Ruppe, S. Griswald, P. Walsh, and R. Martin, "Near term digital radio (ntdr) system," *MILCOM 97 Proceedings*, vol. 3, pp. 1282–1287, November 1997.

[7] "IEEE 802.15.4 MAC standard," Available at http://www.ieee.org/.

[8] P. Kinney. [Online]. Available: http://www.zigbee.org/imwp/idms/popups/popdownload.asp?contentID=812

[9] "IEEE 802.15.TG4a official web page," http://www.ieee802.org/15/pub/TG4a.html.

[10] C. E. Perkins and T. Watson, "Highly dynamic destination sequenced distance vector routing (dsdv) for mobile computers," in *ACM SIGCOMM94 Conference on Communications Architectures*, London, 1994.

[11] P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot,

"Optimized link state routing protocol for ad hoc networks," *IEEE INMIC*, 2001.

[12] D. Johnson, D. Maltz, and J. Jetcheva, "The dynamic source routing protocol for mobile ad hoc networks, internet," *Draft, draft-ietf-manet-dsr-07.txt, work in progress*, 2002.

[13] M. Corson and A. Ephremides, "A distribuited routing algorithm fot mobile wireless networks," *ACM/Baltzer Wireless Networks*, vol. 1, pp. 61–81, 1995.

[14] V. Park and M. Corson, "A highly adaptive distributed routing algorithm for mobile wireless networks, in: Pro- ceedings of infocom, april 1997. v.d. park, m.s. corson, a highly adaptive distributed routing algorithm for mobile wireless networks," *Proceedings of INFOCOM*, April 1997.

[15] C. Perkins, "Ad hoc on demand distance vector (aodv) routing," 1997. [Online]. Available: citeseer.ist.psu.edu/article/perkins00ad.html

[16] Z. Hass and R. Pearlman, "Zone routing protocol for ad-hoc networks," *Internet Draft draft-ietf-manet-zrp-02.txt work in progress*, 1999.

[17] I. W. D. Algorithms, M. for Mobile Computer, and Communications, Eds., *On Calculating Connected Dominant Set for Efficient Routing in Ad Hoc Wireless Networks*, no. 3rd, 1999.

[18] Z. El-Bazzal, M. Kadoch, B. L.Agba, F. Gagnon, and M. Bennani, "An efficient management algorithm for clustering in mobile ad hoc network," *ACM SIGCOMM Computer Communication Review*, 2006.

[19] M. Gerla and J. T. Tsai, "Multiuser, mobile, multimedia radio network," *Wireless Networks*, vol. 1, pp. 255–265, October 1995.

[20] C. R. Lin and M. Gerla, "Adaptive clustering for mobile wireless networks."

[21] P. Basu, N. Khan, and T. Little, "A mobility based metric for clustering in mobile ad hoc networks," *IEEE ICDCS Workshop on Wireless Networks and Mobile Computing*, April 2001.

[22] A. B. McDonald and T. F. Znati, "A mobility-based framework for adaptive clustering in wireless ad hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, August 1999.

[23] A. K. Somani, S. Kher, P. Speck, and J. Chen. Distributed dynamic clustering algorithm in uneven distributed wireless sensor network.

[24] R. Zheng and R. Kravets, "On demand power management for ad-hoc networks," *IEEE Proceedings INFOCOM*, vol. 1, pp. 481–491, March/April 2003.

[25] J. Mingliang, "Cbrp: A cluster based routing protocol for mobile ad hoc networks."

[26] A. D. Amis and R. Prakash, "Load balancing clusters in wireless ad hoc networks."

[27] M. Chatterjee, S. K. Das, and D. Turgut, "An on-demand weighted clustering algorithm (wca) for ad hoc networks," *IEEE*, 2000.

[28] H.-C. Lin and Y.-H. Chu, "A clustering technique for large multihop mobile wireless networks," *IEEE Wireless Communications*, 2000.

[29] S. Basagni, "Distribuited and mobility-adaptive clustering for multimedia support in multi-hop wireless network," *IEEE Wireless Communications*, 1999.

[30] G. Venkataraman, S. Emmanuel, and S. Thambipillai, "A novel distributed cluster cluster maintenance technique for high mobility ad-hoc networks."

[31] R. Min and A. Chandrakasan, "A framework for energy-scalable communication in high-density wireless networks," in *International Symposium on Low Power Electronics and Design (ISLPED)*, 2002, pp. 36–41.

[32] N. Concer, "AdHocSim project," 2005. [Online]. Available: http://www.cs.unibo.it/ concer/adHocSim.html

[33] [Online]. Available: http://www.omnetpp.org/doc/manual/usman.html