3-31-2004

Energy-aware Ad Hoc on-demand distance vector routing protocol and optimizing the blocking problem induced in wireless Ad Hoc networks

Abdallah El Moutia
Florida International University

DOI: 10.25148/etd.FI15101278

FLORIDA INTERNATIONAL UNIVERSITY

Miami, Florida

ENERGY-AWARE AD HOC ON- DEMAND DISTANCE VECTOR ROUTING PROTOCOL AND OPTIMIZING THE BLOCKING PROBLEM INDUCED IN WIRELESS AD HOC NETWORKS

A thesis submitted in partial fulfillment of the

requirements for the degree of

MASTER OF SCIENCE

in

TELECOMMUNICATIONS AND NETWORKING

by

Abdallah El Moutia

2004

To: Dean Vish Prasad
College of Engineering

This thesis, written by Abdallah El Moutia, and entitled Energy-Aware Ad Hoc On-Demand Distance Vector Routing Protocol and Optimizing the Blocking Problem Induced in Wireless Ad Hoc Networks, having been approved in respect to style and intellectual content, is referred to you for judgment.

We have read this thesis and recommend that it be approved.

Jean Andrian

Niki Pissinou

Kia Makki, Major Professor

Date of Defense: March 31, 2004

The thesis of Abdallah El Moutia is approved.

Dean Vish Prasad
College of Engineering

Dean Douglas Wartzok
University Graduate School

Florida International University, 2004

DEDICATION

To My Father
Belkacem El Moutia

My Mother
Tiazza Housni

&

All my family members
Thank you

ACKNOWLEDGMENTS

I am very grateful to Dr. Kia Makki for encouraging me to make this work, may master's thesis. This thesis would have not been possible without his inspiring guidance, continuing support and encouraging. His suggestions and criticisms on the technical aspects of the thesis have been invaluable. The ideas included in this thesis are the result of our intense discussions; it has always been a challenge to propose an idea and justify it to him. I also thank him for his immense patience in reading, correcting and constructively criticizing with his thoughtful insights all the drafts of my thesis.

I am also fortunate to have Dr Niki Pissinou and Dr Jean Andrian serve on my thesis committee. I would like to thank them for members of my committee, and helping me improve the quality of my thesis with their comments. My special thanks go to Dr Niki Pissinou for her valuable advice during the course of my thesis work. She was willing to be my committee member, despite her busy schedule and spend time to edit several drafts of this thesis. Her comments have been very helpful in making the technical presentation of this thesis clearer.

I would also to thank Dr. Jean Andrian for his valuable interaction and comments. Last but not least, I thank all the people in the lab, and all my friends who have been encouraging and supportive at all times.

ABSTRACT OF THE THESIS

ENERGY-AWARE AD HOC ON- DEMAND DISTANCE VECTOR ROUTING PROTOCOL AND OPTIMIZING THE BLOCKING PROBLEM INDUCED IN WIRELESS AD HOC NETWORKS

by

Abdallah El Moutia

Florida International University, 2004

Miami, Florida

Professor Kia Makki, Major Professor

The purpose of this thesis was to investigate some of the issues related to routing and medium access control protocol in ad hoc networks. In routing protocol, the goal is to tackle the power consumption problem and to present a case for using new cost energy-aware metric for Ad Hoc On-Demand Distance Vector (AODV). The idea of the new cost metric is to be able to avoid routes with a low energy capacity. By using this approach, high efficiency in energy consumption can be achieved in Ad-Hoc networks.

The second goal of this thesis was to investigate the blocking problem induced by Request-to-Send/Clear-to-Send (RTS/CTS) mechanism in detail and provide a solution to overcome that problem. To do so, a new parameter is proposed by which the Medium Access control (MAC) protocol will decide when to switch between RTS/CTS mechanism (the 4-way-handshaking) and the Basic Access method (the 2-way-handshaking) in order to reduce the effect of the blocking problem in Ad Hoc networks.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE PAGE

Chapter 1 Introduction

1.1 Mobile Wireless Ad-Hoc Networks

Computing and communication anytime, anywhere is a global trend in today's development. Ubiquitous computing has been made possible by the advance of wireless communication technology and availability of many lightweight, portable-computing devices. Among the various network architecture that exist today, the development and design of mobile Ad Hoc network has drawn a lot of attention recently. The concept of mobile wireless ad hoc networking has unique features in which neither base stations nor wired backbone networks are required to setup a network and mobile nodes can communicate with each other beyond their transmission range by multi-hopping fashion. While Ad hoc networks have found many applications and attracted attention from research community since the early nineties, it is the technology provided by the IEEE 802.11 that allows its implementation to be possible. Currently, there are two types of mobile wireless networks; with the first being known as the wireless LANs (Local Area Networks) or an infrastructure network and the second is known as the Ad Hoc networks or an infrastructure-less networks. In this section, we will provide an overview of a wireless LAN and an Ad Hoc network that should be sufficient to distinguish the main differences between these two types of a wireless network.

1.1.1 Wireless LANs.

Wireless LAN is an extension to the wired Ethernet, defined in the IEEE 802.11 standards [1]. It is becoming very popular in providing mobile Internet access in offices and campus buildings and the ability to access critical information in corporate networks from any remote location. Wireless LAN is centralized in nature, meaning that the

network has an access point (AP) that acts as the interface between wireless and wired networks. In centralized networks, the AP manages and administers all the communications that take place between mobile users and mobile users with a wired network. Therefore, the AP could be considered as a router or a hub connecting several mobile end hosts to the LAN system. The system architecture of a wireless LAN is shown in Figure 1.1.

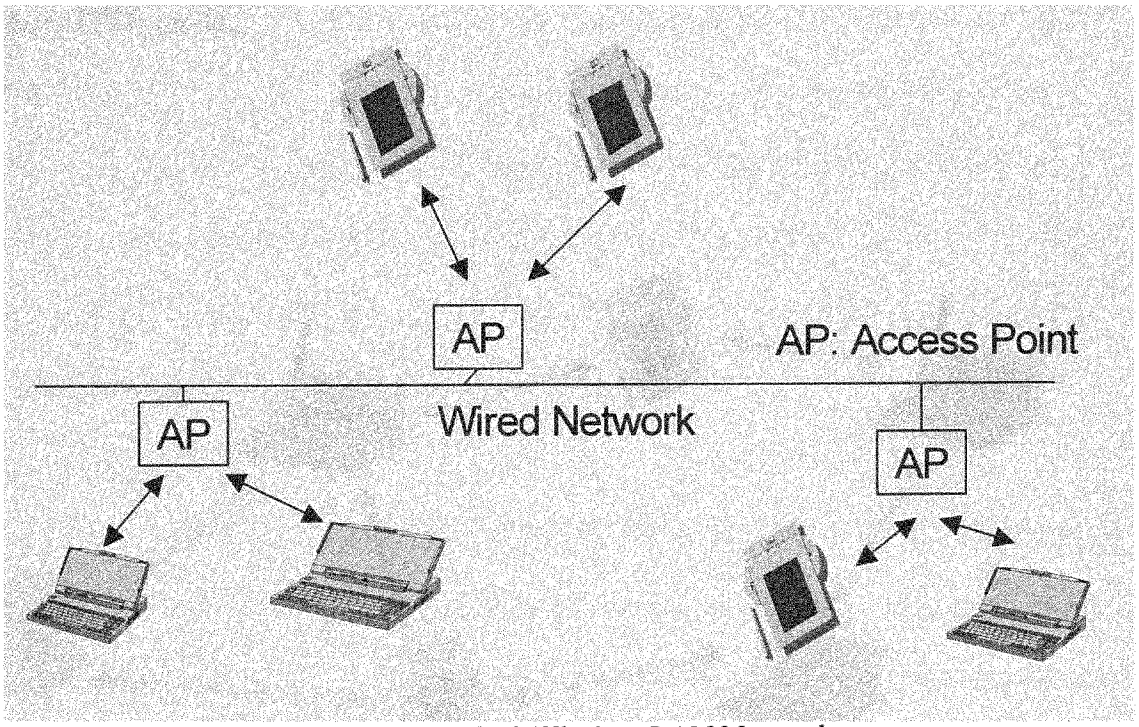


Figure 1.1 A typical Wireless LAN Network

1.1.2 Ad Hoc Networks.

Ad Hoc networks, also called distributed wireless networks, are sets of mobile wireless terminals communicating with one another with no pre-existing infrastructure in place; therefore, they are called infrastructure-less networks. A typical Ad Hoc network is illustrated in Figure 1.2. Ad Hoc networks are self-organizing and adaptive, meaning that

networks can be formed on the fly without the need of any system administration. Also, in Ad Hoc networks, nodes forward packets on behalf of each other and take their own decision in packet routing, accessing the medium, and managing the power consumption. All the data transmission and reception in Ad Hoc networks have to be in the same frequency band since there is no special node to translate the transmission from one frequency band to another. Therefore, all Ad Hoc networks operate in time division duplex (TDD) mode.

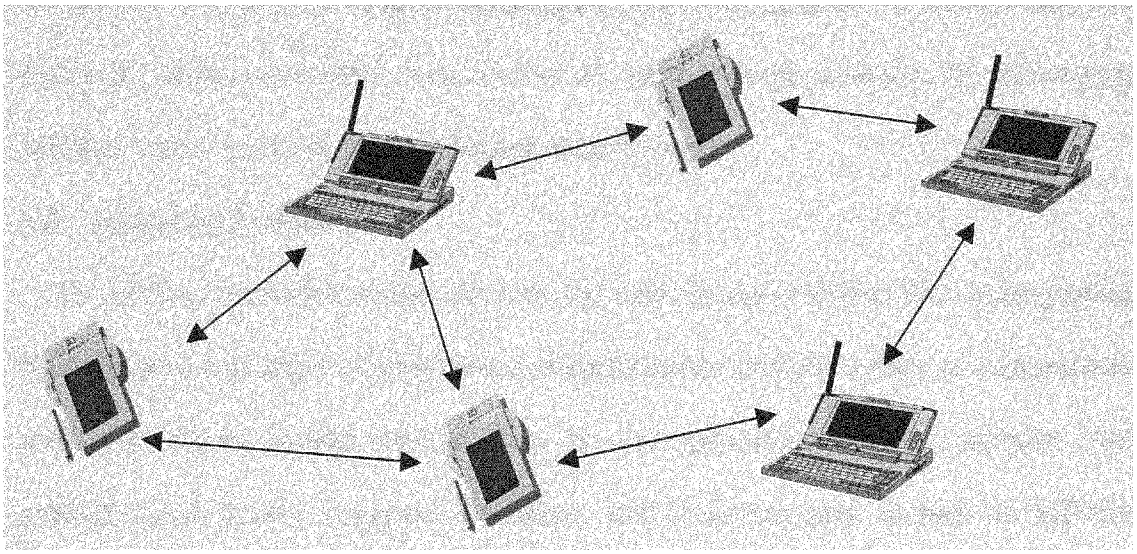


Figure 1.2 A typical Ad Hoc Network

1.2 Applications of Mobile Ad Hoc Networks.

In the coming years, Mobile Ad-hoc networks are expected to play an important role in commercial and military settings where mobile access to a wired network is either ineffective or impossible [2]. Potential applications for this class of network include instant network infrastructure to support collaborative computing in temporary or mobile environments, emergency rescue networks in disaster, remote control of electrical appliance, communication systems such as IVC (Inter-Vehicle Communications), and mobile access to the global Internet. Furthermore, ad-hoc networks have the potential to serve as a ubiquitous wireless infrastructure capable of interconnecting many thousands of devices with a wide range of capabilities and uses. In order to achieve this status, however, ad-hoc networks must evolve to support large numbers of heterogeneous systems with a wide range of application requirements.

1.3 Research Problem.

In Ad Hoc networks, mobile devices can exist in many forms. The heterogeneity of these devices can affect communuication performance and the design of communication protocols. These mobile devices vary in size, processing power, memory storage capacity, and battery capacity. Therefore, the challenge here is how to effectively establish a network beween these devices without violating the capability of each device. Due to the fact that these mobile nodes are battery-operated, research efforts for energy-aware design of network protocols for the ad hoc networking environmnet has been extensivelly explored in the recent years.

Moreover, since each mobile node host in ad hoc network performs routing functinality on behalf of other nodes, energy exhaustion of one or several devices might

cause a serious distruption of the entire network. Yhis can become more serious problem in long-lasting applications, such as distirbution of voice orvedio stream, which will be widely-deployed in the near future. In a conventional routing algorithm without consideration of energy consumtion at hosts, a connection-oriented traffic tends to use the shortest route path, which results in a quick exhaustion of energy of the nodes along the path in the presence of heavy load on that path. Thus, the problem of saving energy consumption of each host and maximizing the lifetime of the system can be an interesting problem.

In addition to the routing protocol, which in this context we see a clear need for improvement, mediun access control protocol or MAC needs some ehancement to better improve the overall performance of ad hoc networks. Also, due to the properties of Ad hoc networks, MAC protocol plays a crucial role in the efficient and fair sharing of the scarce wireless bandwidth and defines rules for accessing the medium. Therefore, the performance o f a w ireless A d-Hoc network critically depends upon the MAC protocol used. Request-to-send/Clear-to-send or RTS/CTS mechanism is often chosen because it solves the hidden and exposed node problems. Generally, the RTS/CTS mechanism works well in infrastructure-based networks, even though it may result to unfairness in some situations [3]. However, the RTS/CTS mechanism, in the setting of Ad-Hoc networks, results to situations where a large number of nodes are unable to transmit any packet and are blocked for long period of time.

In this thesis, we are primarily interseted in issues related to routing and medium access control protocol in ad hoc networks. In routing protocol, our goal is to tackle the power consumption problem. We are reaserching on using extensions to Ad-Hoc On-

Demand Distance Vector routing protocol (AODV) and providing our extenssion to AODV for achieving high efficiency in energy consumption in Ad-Hoc networks. The second goal of this thesis is to investigate the blocking problem induced by RTS/CTS mechanism in detail and provide a solution to overcome that problem. The following is an overview of this thesis:

Chapter 2 describes the related work in proposed area.

Chapter 3 describes the actual design of the thesis.

Chapter 4 provides an insight into the implementation of the thesis.

Chapter 5 explains the results obtained after implementation

Chapter 6 states the conclusion of the thesis and provides a few thoughts for future work.

Chapter 2 Related Work.

2.1 Introduction.

Designing energy-aware routing protocols for mobile Ad Hoc Networks is necessary for networks connecting Portable and self-networking devices. A number of research [4, 5, 6] on energy-aware routing protocols in Ad Hoc networks focus on minimizing the total energy consumption per packets. The problem of maximizing the span of the system is considered in [7, 8] and optimal and near optimal solutions are provided in case of single power level and multiple power levels respectively. However, Energy-Aware Ad Hoc on Demand Distance vector with the consideration of end-to-end delay has not been thoroughly explored.

2.2 Discussion of Power-Aware Routing.

In [9], they considered a routing algorithm for connection-oriented traffic in Ad Hoc networks. However, they focus on minimizing blocking probability with the minimal energy expenditures instead of maximizing lifetime of the system. They create several metrics and embed them in a distributed Bellman-Ford algorithm to obtain a good blocking probability along with less energy consumption.

[10] Contains simulation results about a performance comparison for well-known Ad Hoc routing protocols including AODV [11], DSR [12], DSDV [13] and TORA [14]. They obtain enough evidence to suggest that among all these, AODV and DSR are the best energy conservative algorithm. However, AODV consumes slightly less energy than DSR at comparable packet delivery ratio.

In [7], instead of minimizing the consumed energy, the problem of maximizing the lifetime of the system is proposed with the model of single destination. They identified

the maximum lifetime problem as a linear programming and provided an optimal solution. The work in [7] is extended in [8] to the multi-commodity case, where each commodity has its own set of destination. However, their approaches are suitable for static sensor networks rather than dynamic Ad Hoc Network environments since they are based on static topology of nodes and given traffic demands.

2.3 Energy cost metric used in current routing protocols.

The majority of the work reported in the literature focuses on the protocol design and performance evaluation in terms of traditional metrics such throughput, delay and routing overhead. In this section, we will discuss most of the energy cost metrics used in current routing protocols.

2.3.1 Minimize Energy Consumed per Packet.

This is one of the most obvious metrics that is used for conserving energy. Now assume that a packet (j) is traveling from N_1 (source) to N_k (Destination). Let E(S, D) denote the energy consumed in transmitting and receiving one packet from S to D. Then, the energy consumed for packet j is: $e_j = \sum E(N_i, N_{i+1})$, as $1 \leq i \leq k-1$. Thus, the goal of this metric is to minimize e_j. It is obvious to see that this metric will tend to minimize the average energy consumed per packet. Under a light load however, this will be identical to routes selected by the shortest-hop routing. This is an obvious observation because, if we assume that E(S, D) = E is a constant and E is the energy consumed per packet between two neighboring nodes, then the power consumed is:

$e_j = \sum E(N_i, N_{i+1})$, *as* $1 \leq i \leq k-1$.

$e_j = (N_2 - N_1) + (N_3 - N_2) + (N_4 - N_3) + \ldots\ldots + (N_k - N_{k-1})$.

After eliminating some of the terms, we obtain:

$e_j = E(-N_I + N_k) = E(N_k - N_I) = E(K-1)$.

$e_j = E(K-1)$.

To minimize this value, we need to minimize K, which is equivalent to finding the shortest path. One serious disadvantage of this metric is the early death for some nodes since nodes will tend to have different energy consumption due to their size and energy capacity. To better illustrate this, consider the network in Figure 2.1. Here, node G will be selected as the route for data packets going from [A-D], [B-E], and [C-F]. As result, node G will be used heavily and its battery resources will expend quickly than other nodes in the network and will die first. Therefore, this metric does not meet the goal of increasing node and network lifetime.

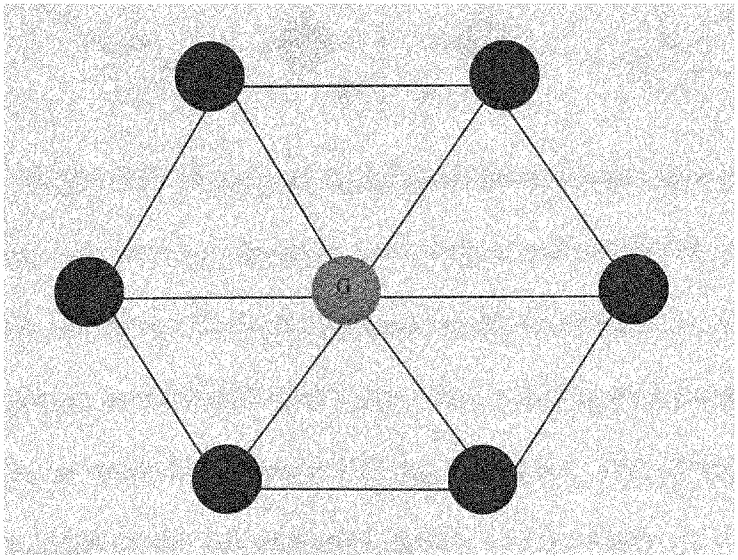


Figure 2.1 A network illustrating the problem with metric of the Min-Energy consumed Packet.

2.3.2 Maximize Time to Network Partition

The implementation of this metric is very important in critical application such as military, law enforcement, and rescue missions. However, to maintain low delay and high throughput, optimizing this metric is very difficult. Given the network topology, in Figure 2.2, and the use of the maximum-flow-min-cut theorem, we can find a minimal set of nodes or the cut-set the removal of which will lead to network partition. Therefore, the routes

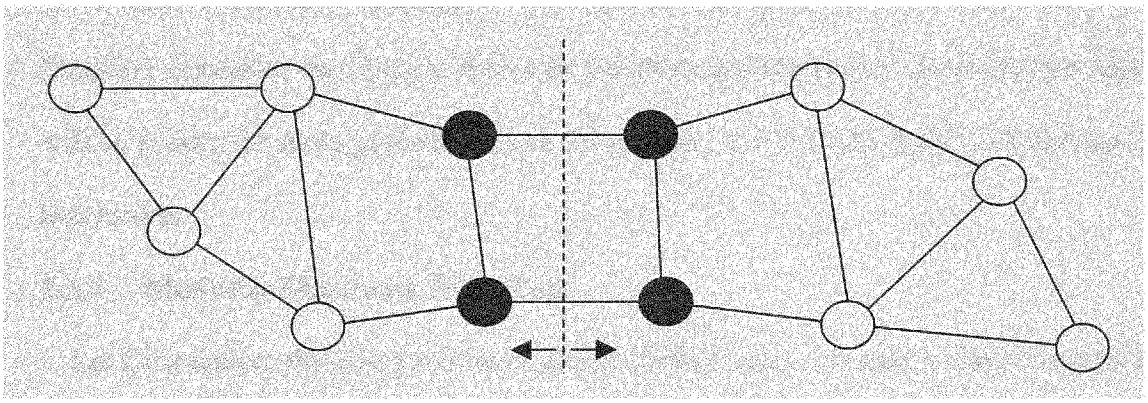


Figure 2.2 A network illustrating the maximum-flow-min-cut theorem

between these sets must go through one of these critical nodes. As a result, a routing protocol must divide the work among these nodes to maximize a network lifetime. This problem is similar to the load balancing problem where tasks need to be sent to one of the many servers in order to minimize the response time. This is known as NP-complete problem. Because of the heterogeneity nature mobile nodes in Ad Hoc networks, we cannot ensure that these nodes will drain their energy at equal rate. Therefore, we will see high delay and low throughput as soon as one of these nodes dies.

2.3.3 Minimize Cost per Packet

If the goal is to maximize the life of all nodes in the network, then metrics other than energy consumed per packet need to be used. The selected paths when using these metrics should be such that nodes with minimum energy reserves do not lie on many paths. Now, let $f_i(x_i)$ be a function that denotes the node cost i, xi represents the total energy expended by node i so far. Then, the total cost of sending a packet j along some path as the sum of the node cost of all nodes that lie along that path is denoted by:

$$c_j = \sum f_i(x_i) \text{ where } 1 \leq i \leq k\text{-}1 \text{ as packet } j \text{ travel from } n_1 \text{ to } n_{k-1}.$$

Thus, if f_i is a linearly increasing function, then node G in figure 3 will not overused therefore increasing its lifetime. However, the delay and the energy consumed per packet will be greater for some packets, such as those from [A-D], [B-E], and [C-F] that use 3-hop routes.

2.3.4 Minimize Maximum Node Cost

Let $Ci(t)$ denote the cost f routing a packet through node I at time t. then, the goal is to minimize the maximum node cost $C_i(t)$, where $t>0$ after routing N packets to their destinations or after T seconds. All of these variation ensure that node failure is delayed and the drawback is that the variance in power levels is also reduced, unfortunately, we see no way of implementing this metric directly in a routing protocol.; however, minimizing the cost per node does significantly reduce the maximum node cost and the time to first node failure.

2.4 Description of existing Ad Hoc Routing Protocols

Since the advent of Defense Advanced Research Projects Agency (DARPA) packet radio networks in the early 1970s [15], many routing protocols have been developed for ad hoc networks. As shown in Figure 2.3, these protocols may generally be categorized as:

- Table-Driven Routing Protocols.
- Source-Initiated On-demand Routing Protocols.

Even though, these routing protocols have been designed for the same type of network, the characteristics of each of these protocols are quite distinct. The following section describes some the protocols and categorized them according to their characteristics.

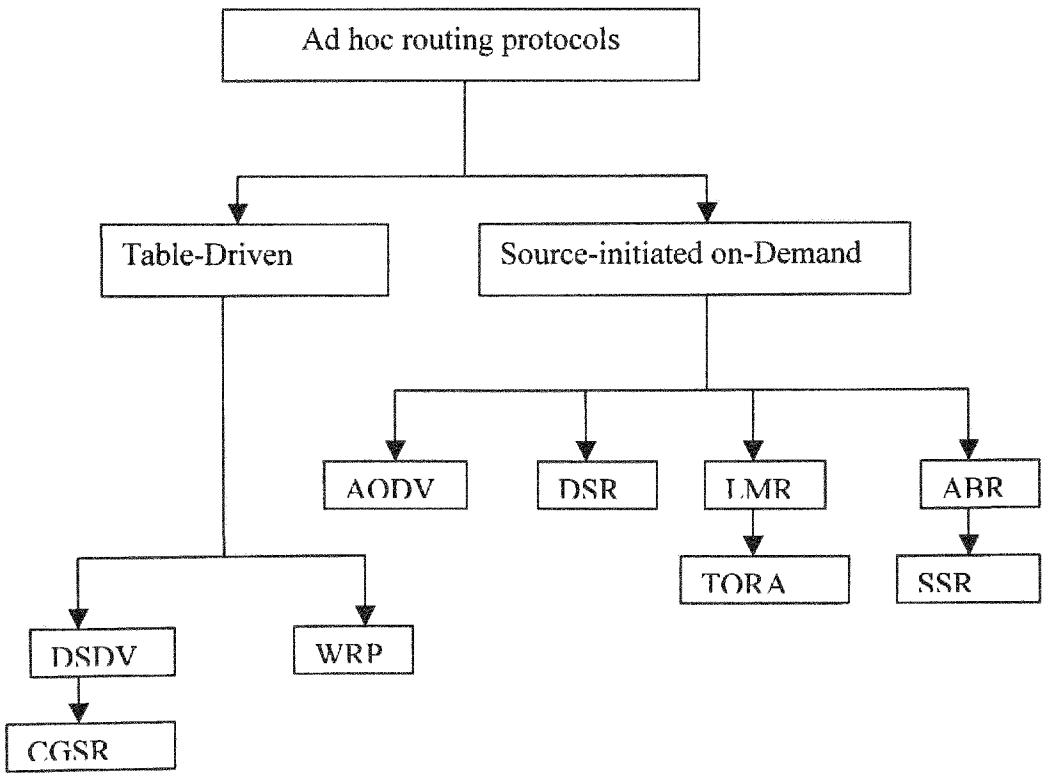


Figure 2.3 Categorization of ad hoc routing protocols.

2.4.1 Table-Driven Routing Protocols

All nodes, using table-driven routing protocols, attempt to have a complete knowledge of paths to all other nodes in a network. These protocols require each node to maintain more than one table to store routing information. They quickly respond to network topology changes by propagating updates throughout the network. They area in which they differ are the number of necessary routing-related tables and the methods by which changes in network structure are broadcast. Some of the existing table-driven ad hoc routing protocols are discussed in the following sections.

2.4.1.1 Dynamic Destination-Sequenced Distance-Vector Routing Protocol

The Destination-Sequenced Distance-Vector (DSDV) Routing Algorithm [16] is based on the idea of the classical Bellman-Ford Routing Algorithm with certain improvements. Every mobile station maintains a routing table that lists all available destinations, the number of hops to reach the destination and the sequence number assigned by the destination node. The sequence number is used to distinguish stale routes from new ones and thus avoid the formation of loops. The stations periodically transmit their routing tables to their immediate neighbors. A station also transmits its routing table if a significant change has occurred in its table from the last update sent. So, the update is both time-driven and event-driven. The routing table updates can be sent in two ways:- a "full dump" or an incremental update. A full dump sends the full routing table to the neighbors and could span many packets whereas in an incremental update only those entries from the routing table are sent that has a metric change since the last update and it must fit in a packet. If there is space in the incremental update packet then those entries may be included whose sequence number has changed. When the network is relatively

stable, incremental updates are sent to avoid extra traffic and full dump are relatively infrequent. In a fast-changing network, incremental packets can grow big so full dumps will be more frequent. Each route update packet, in addition to the routing table information, also contains a unique sequence number assigned by the transmitter. The route labeled with the highest (i.e. most recent) sequence number is used. If two routes have the same sequence number then the route with the best metric (i.e. shortest route) is used. Based on the past history, the stations estimate the settling time of routes. The stations delay the transmission of a routing update by settling time so as to eliminate those updates that would occur if a better route were found very soon.

2.4.1.2 The Wireless Routing Protocol (WRP)

The Wireless Routing Protocol (WRP) [17] is a table-based distance-vector routing protocol. Each node in the network maintains a Distance table, a Routing table, a Link-Cost table and a Message Retransmission list. The Distance table of a node x contains the distance of each destination node y via each neighbor z of x. It also contains the downstream neighbor of z through which this path is realized. The Routing table of node x contains the distance of each destination node y from node x, the predecessor and the successor of node x on this path. It also contains a tag to identify if the entry is a simple path, a loop or invalid. Storing predecessor and successor in the table is beneficial in detecting loops and avoiding counting-to-infinity problems. The Link-Cost table contains cost of link to each neighbor of the node and the number of timeouts since an error-free message was received from that neighbor. The Message Retransmission list (MRL) contains information to let a node know which of its neighbor has not acknowledged its update message and to retransmit update message to that neighbor. Nodes exchange

routing tables with their neighbors using update messages periodically as well as on link changes. The nodes present on the response list of update message (formed using MRL) are required to acknowledge the receipt of update message. If there is no change in routing table since last update, the node is required to send an idle Hello message to ensure connectivity. On receiving an update message, the node modifies its distance table and looks for better paths using new information. Any new path so found is relayed back to the original nodes so that they can update their tables. The node also updates its routing table if the new path is better than the existing path. On receiving an ACK, the mode updates its MRL. A unique feature of this algorithm is that it checks the consistency of all its neighbors every time it detects a change in link of any of its neighbors. Consistency check in this manner helps eliminate looping situations in a better way and also has fast convergence.

2.4.1.3 Cluster head Gateway Switch Routing Protocol

Cluster head Gateway Switch Routing (CGSR) [18] uses as basis the DSDV Routing algorithm described in the previous section. The mobile nodes are aggregated into clusters and a cluster-head is elected. All nodes that are in the communication range of the cluster-head belong to its cluster. A gateway node is a node that is in the communication range of two or more cluster-heads. In a dynamic network cluster head scheme can cause performance degradation due to frequent cluster-head elections, so CGSR uses a Least Cluster Change (LCC) algorithm. In LCC, cluster-head change occurs only if a change in network causes two cluster-heads to come into one cluster or one of the nodes moves out of the range of all the cluster-heads. The general algorithm works in the following manner. The source of the packet transmits the packet to its cluster-head.

From this cluster-head, the packet is sent to the gateway node that connects this cluster-head and the next cluster-head along the route to the destination. The gateway sends it to that cluster-head and so on till the destination cluster-head is reached in this way. The destination cluster-head then transmits the packet to the destination.

2.4.2 Source-Initiated On-Demand Routing Protocols

Paths are discovered when they are required. Generally, a source initiates a route discovery when it desires to send packets. Once the route has been established, it is maintained until either the destination becomes inaccessible or until the route is no longer desired. The followings are some of On-Demand or Source-Initiated routing protocols.

2.4.2.1 Cluster based Routing Protocols

In Cluster Based Routing protocol (CBRP) [19], the nodes are divided into clusters. To form the cluster the following algorithm is used. When a node comes up, it enters the "undecided" state, starts a timer and broadcasts a Hello message. When a cluster-head gets this hello message it responds with a triggered hello message immediately. When the undecided node gets this message it sets its state to "member". If the undecided node times out, then it makes itself the cluster-head if it has bi-directional link to some neighbor otherwise it remains in undecided state and repeats the procedure again. Cluster heads are changed as infrequently as possible. Each node maintains a neighbor table. For each neighbor, the neighbor table of a node contains the status of the link (uni- or bi-directional) and the state of the neighbor (cluster-head or member). A cluster-head keeps information about the members of its cluster and also maintains a cluster adjacency table that contains information about the neighboring clusters. For each neighbor cluster, the table has entry that contains the gateway through which the cluster can be reached and

the cluster-head of the cluster. When a source has to send data to destination, it floods route request packets (but only to the neighboring cluster-heads). On receiving the request a cluster-head checks to see if the destination is in its cluster. If yes, then it sends the request directly to the destination else it sends it to all its adjacent cluster-heads. The cluster-heads address is recorded in the packet so a cluster-head discards a request packet that it has already seen. When the destination receives the request packet, it replies back with the route that had been recorded in the request packet. If the source does not receive a reply within a time period, it backs off exponentially before trying to send route request again. In CBRP, routing is done using source routing. It also uses route shortening that is on receiving a source route packet, the node tries to find the farthest node in the route that is its neighbor (this could have happened due to a topology change) and sends the packet to that node thus reducing the route. While forwarding the packet if a node detects a broken link it sends back an error message to the source and then uses local repair mechanism. In local repair mechanism, when a node finds the next hop is unreachable, it checks to see if the next hop can be reached through any of its neighbor or if hop after next hop can be reached through any other neighbor. If any of the two works, the packet can be sent out over the repaired path.

2.4.2.2 Ad hoc On-demand Distance Vector Routing

Ad hoc On-demand Distance Vector Routing (AODV) [20] is an improvement on the DSDV algorithm. AODV minimizes the number of broadcasts by creating routes on-demand as opposed to DSDV that maintains the list of all the routes. To find a path to the destination, the source broadcasts a route request packet. The neighbors in turn broadcast the packet to their neighbors till it reaches an intermediate node that has recent route

information about the destination or till it reaches the destination. A node discards a route request packet that it has already seen. The route request packet uses sequence numbers to ensure that the routes are loop free and to make sure that if the intermediate nodes reply to route requests, they reply with the latest information only. When a node forwards a route request packet to its neighbors, it also records in its tables the node from which the first copy of the request came. This information is used to construct the reverse path for the route reply packet. AODV uses only symmetric links because the route reply packet follows the reverse path of route request packet. As the route reply packet traverses back to the source, the nodes along the path enter the forward route into their tables. If the source moves then it can reinitiate route discovery to the destination. If one of the intermediate nodes move then the moved nodes neighbor realizes the link failure and sends a link failure notification to its upstream neighbors and so on till it reaches the source upon which the source can reinitiate route discovery if needed.

2.4.2.3 Dynamic Source Routing Protocol

The Dynamic Source Routing Protocol [21] is a source-routed on-demand routing protocol. A node maintains route caches containing the source routes that it is aware of. The node updates entries in the route cache as and when it learns about new routes. The two major phases of the protocol are: route discovery and route maintenance. When the source node wants to send a packet to a destination, it looks up its route cache to determine if it already contains a route to the destination. If it finds that an unexpired route to the destination exists, then it uses this route to send the packet. But if the node does not have such a route, then it initiates the route discovery process by broadcasting a route request packet. The route request packet contains the address of the source and the

destination, and a unique identification number. Each intermediate node checks whether it knows of a route to the destination. If it does not, it appends its address to the route record of the packet and forwards the packet to its neighbors. To limit the number of route requests propagated, a node processes the route request packet only if it has not already seen the packet and its address is not present in the route record of the packet. A route reply is generated when either the destination or an intermediate node with current information about the destination receives the route request packet [22]. A route request packet reaching such a node already contains, in its route record, the sequence of hops taken from the source to this node. As the route request packet propagates through the network, the route record is formed. If the route reply is generated by the destination then it places the route record from route request packet into the route reply packet.

On the other hand, if the node generating the route reply is an intermediate node then it appends its cached route to destination to the route record of route request packet and puts that into the route reply packet. To send the route reply packet, the responding node must have a route to the source. If it has a route to the source in its route cache, it can use that route. The reverse of route record can be used if symmetric links are supported. In case symmetric links are not supported, the node can initiate route discovery to source and piggyback the route reply on this new route request. DSRP uses two types of packets for route maintenance: - Route Error packet and Acknowledgements. When a node encounters a fatal transmission problem at its data link layer, it generates a Route Error packet. When a node receives a route error packet, it removes the hop in error from its route cache. All routes that contain the hop in error are truncated at that point. Acknowledgment packets are used to verify the correct operation

of the route links. This also includes passive acknowledgments in which a node hears the next hop forwarding the packet along the route.

Chapter 3 Design of Energy-Aware AODV & Optimizing the Blocking problem Induced by RTS/CTS mechanism in Ad Hoc Networks

3.1 Introduction

In this chapter, we describe the design of Energy-Aware AODV (EA-AODV) and the optimizing of the blocking problem induced by RTS/CTS mechanism in Ad Hoc networks. To do so, we first explain in detail how AODV works with an example. Then, we present our modified version EA-AODV. After that, we introduce our approach in optimizing the blocking problem induced by RTS/CTS mechanism. Within this context, we first provide a brief description of the 802.11 MAC protocols. Then, we discuss 802.11 MAC Issues such as a hidden terminal problem, exposed terminal problem, and blocking problem. Finally, we present our proposed solution to solve the blocking problem induced in ad Hoc networks.

3.2 Detail description of AODV

3.2.1 Route Discovery

As briefly mentioned in section 2.4.2, AODV is source initiated and reactive protocol. It discovers and maintains routes only if and when necessary. To describe the protocol in detail, consider the Ad Hoc network of Figure 3.1, in which a process at node (A), a source, wants to send a packet to node (I), a destination. Let's suppose that node (A) does not have an entry for node (I) in its table. Now, it has to discover a route to (I) and route discovery process works as follow. Node (A) constructs a special ROUTE REQUEST packet (RREQ) and broadcasts it. As shown in Figure 3.1, the packet reaches nodes (B) and (C). The format of the RREQ packet is shown in Figure 3.6 and it contains the following fields:

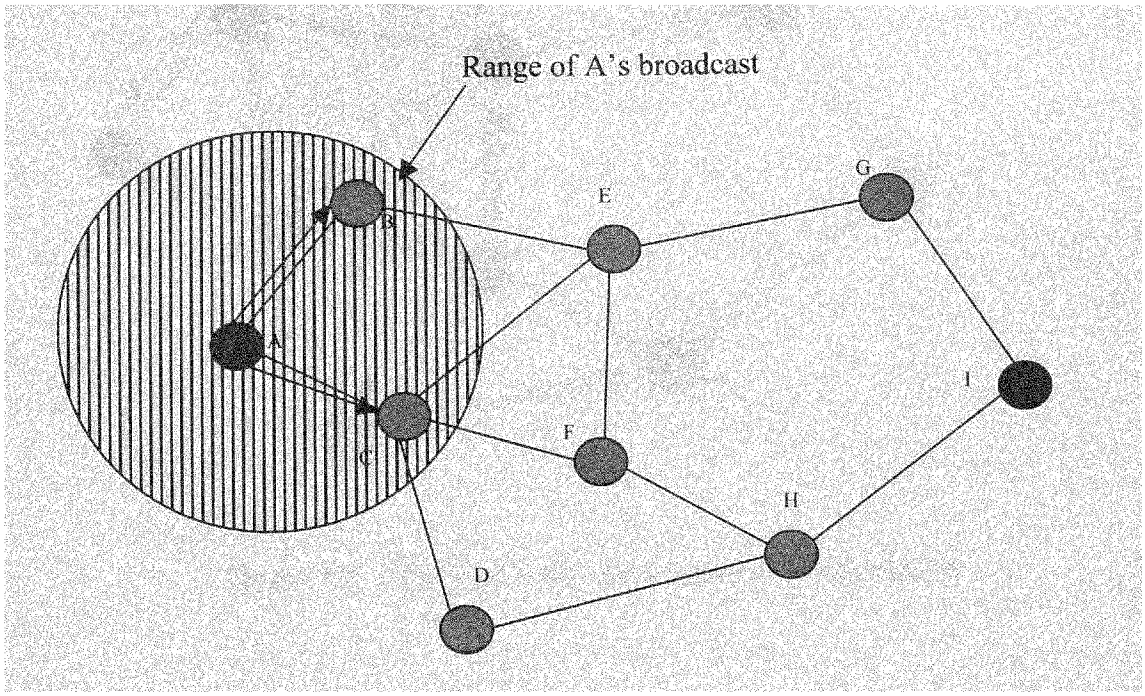


Figure 3.1 Range of A's broadcast

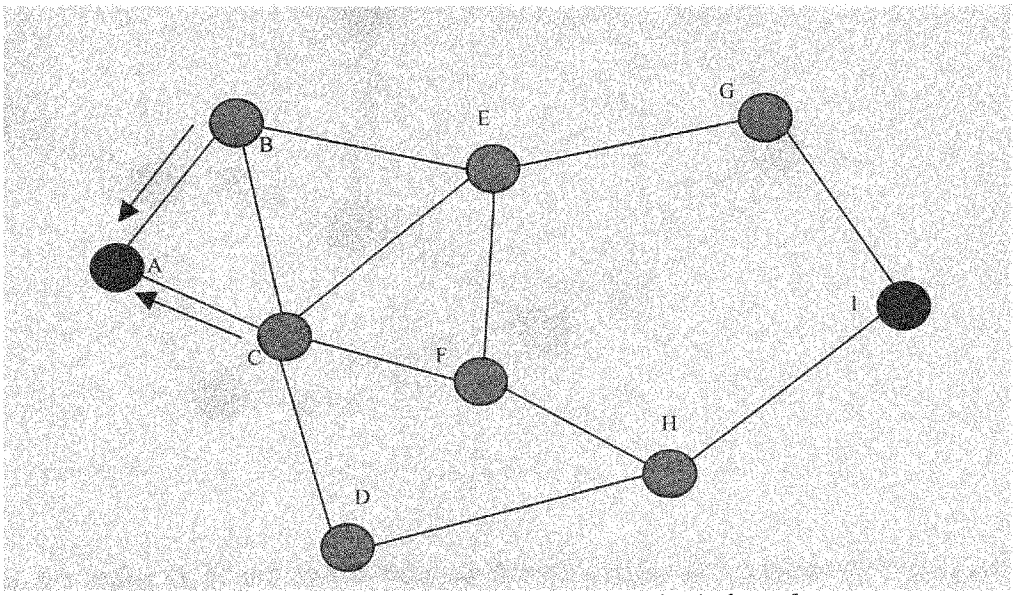


Figure 3.2 After B and C have received A's broadcast

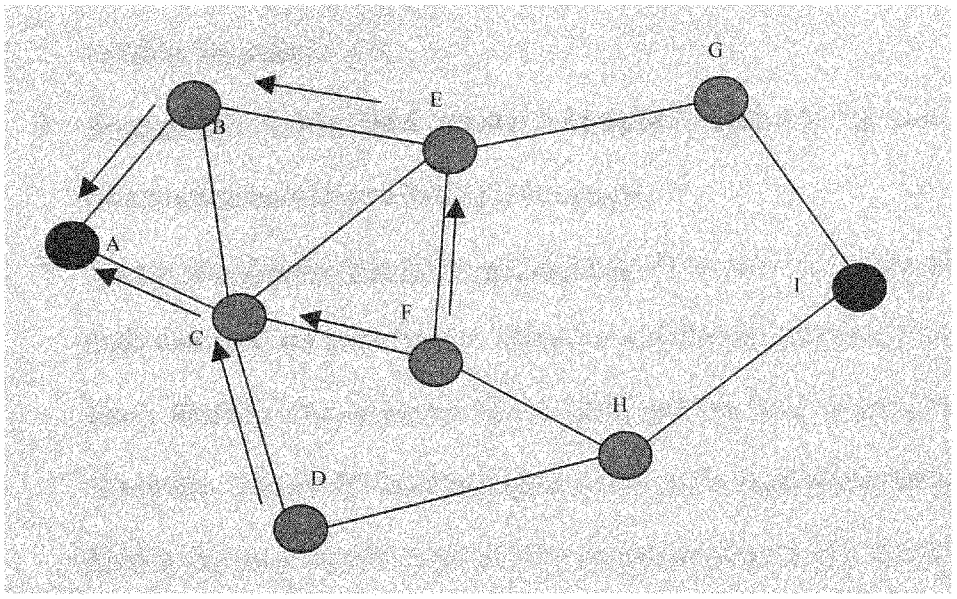


Figure 3.4 After D, E and F have received A's broadcast.

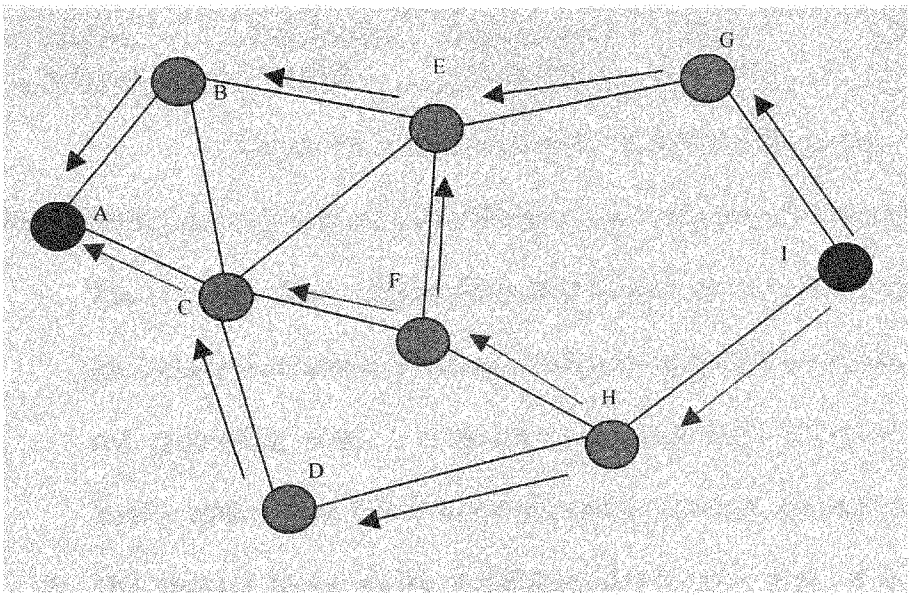


Figure 3.5 After G, H and I have received A's broadcast. The arrows show the possible reverse routes

- *The Source and Destination address*: The IP addresses that identify who is looking for whom.
- *Request ID*: It is a local counter maintained separately by each node and is incremented each time a RREQ is broadcast.
- *Source and Destination Sequence number*: They are maintained locally by each node and they are incremented whenever a RREQ is sent or (a reply to someone else's RREQ). They function like a clock and are used to tell new routes from old routes. The fourth field of Figure 3.6 is (A)'s sequence counter and the fifth field is the most recent value of (I)'s sequence number that the source has seen (0 if it has never seen).
- *The hop count*: It keeps track of how many hops the packet has made. It is initialized to 0.

Source Address	Destination Address	Destination Sequence #	Hop Count	Lifetime

Figure 3.6 Format of a ROUTE REQUEST packet.

When RREQ packet arrives at nodes (B) & (C), it is processed in the following steps:

1. The source address, request ID pair is looked up in a local history table to see if this request has already been seen and processed. If it is a duplicate, it is discarded and processing stops. If it is not a duplicate, then the pair is entered into the history table so that future duplicates can be rejected, and processing continues.
2. The receiver looks up the destination in its route table. If a fresh route to the destination is known, a RREP packet is sent back to the source telling it how to get to the destination. Fresh means that the *Destination Sequence Number* stored

in the routing table is greater than or equal to the *Destination Sequence Number* in the RREQ packet. If it is less, the stored route is older than the previous route the source had for the destination, so step is executed.

3. Since the receiver does not know a fresh route to the destination, it increments the Hop count field and rebroadcasts the RREQ packet. It also extracts the data from the packet and stores it as a new entry in it reverse route table. This information will be used to construct the reverse route so that the reply can get back to the source later. A timer is also started for the newly made reverse route entry. If it expires, the entry is deleted.

Since neither node (B) nor (C) knows where node (I) is, both nodes creates a reverse route back to A and broadcasts the packet with *Hop count* set to 1. The broadcast from B will reach (C) and (E). (E) makes an entry for it in its reverse route table and rebroadcasts it. On the other hand, node (C) rejects the packet as a duplicate. Similarly, (B) rejects (C)'s broadcast. After F, G, H, and I receive the broadcast, the RREQ finally reaches a destination that knows where node I is. In response to the incoming request, node I builds a RREP packet as shown in Figure 3.7. The Hop count field is set to 0. The lifetime field controls how long the route is valid. This packet is unicasted to the node that the RREQ packet came from, in this case, node H. Then, it follows the reverse path to A through H, F, and C as shown in Figure 3.4. Once the source node receives the RREP, it may begin to forward data packets to the destination. If the source receives a RREP with a greater sequence number or contains the same sequence number with a smaller *hop-count*, it updates its routing information for that destination and begins using a better route.

Source Address	Destination Address	Destination Sequence #	Hop Count	Lifetime

Figure 3.7 Format of a ROUTE REPLY packet.

3.2.2 AODV Route Maintenance

The mobility and the dynamic nature of the Ad Hoc Networks make routing very challenging. Going back to Figure 3.1, if node F moved away, node A will not know that the route {ACFHI}, it was using is no longer valid. Therefore, the protocol needs to be able to maintain routes as fast as the change of topology in order to check the validation of any route. Periodically, each node broadcast a *Hello* message and each of its neighbors is expected to reply to it. If there is no reply, the broadcaster knows that the neighbor has moved out of range and is no longer connected to it. As an example of route maintenance, when C discovers that F is gone, it looks at its routing table and sees that F was used on routes to E, F, H and I. The union of the active neighbors for these destinations is the set {A, B}. In other words, A and B depend on F for some of their routes, so they have to be informed that these routes no longer work. C tells them by sending them packets that cause them to update their own routing tables accordingly. C also purges the entries for E, F, H, and I from its routing table.

3.3 Energy-Aware AODV

Our approach in conserving power in Ad Hoc Networks is based on two techniques. In the first, we modify the AODV routing protocol to route around nodes with lower energy capacity. In other words, the protocol chooses routes according to their energy cost. In

the second, we strategically allow nodes to make some local decision based on their energy resources available whether nor not to process the RREQ packet.

3.3.1 Energy Cost Metric

In AODV, activity begins with the source flooding the network with the RREQ packet when it has data to send. An intermediate node will process and broadcast the RREQ packet unless it gets a path to the destination from its cache or it has already processed and broadcast the same packet. The destination node will reply only to the first arrived PREQ packet since that packet tends to take the shortest path (low delay). The potential problem in this current protocol is that it finds the shortest path and uses that path for every communication. However, that is not the best thing to do for network lifetime. Using the shortest path more frequently leads to energy depletion of nodes along that path and may cause network partition.

In EA-AODV, when the source has data to send, it broadcast the RREQ packet. Since many portable devices today display a battery discharge curve, we can use this curve to allow intermediate nodes to make local decision whether or not to process the RREQ packet. When an intermediate node receives the RREQ packet, it first checks its energy resource available during communication time. If the consumed battery energy is less or equal α then process the packet, else send an error message. All the nodes except the destination calculate their link cost ($Lc = Pt*(F/R(t))$ and add it to the total cost ($C = \sum Ci$) in the header and broadcast the RREQ. Once the destination receives the first RREQ, it starts a timer (Tr). During that time, the destination examines the total cost of every arrived packet. If the total cost of every link is less or equal than certain a threshold

value (β: 10% of the total initial energy), then the destination node will use the shortest route to unicast

the RREP packet ; otherwise, it will choose the route with the minimum cost. Taking this approach will compensate the tradeoff between latency and prolonging the network lifetime. In other words, when the network is new, the shortest path approach is applied; but, when the network is being utilized for certain period of time, the proposed cost function is used. Table 3.1 describes the basic energy-aware AODV algorithm.

Table 3.1 The basic Energy-Aware AODV Algorithm

Node	Steps
Source Node	*Broadcast the Route_Request packet;* *Wait for the Route_Reply packet;*
Intermediate Node	*If the consumed battery energy ≤ α then process the packet;* *Else discard the packet & send an error message;* *If the consumed battery energy ≤ α then do:* *Ri(t) = Initial_Energy – Energy(tx/rx).* *Ci(t) = Pi*[F/Ri(t)].* *Ej(t) = ∑Ci(t).*
Destination Node	*Receive the first Route_Reply;* *Start a timer Tr;* *During Tr, Examine the total cost of every arrived packet;* *If Ej(t) ≤ β (of the total initial energy) then choose the shortest path;* *Else choose the route with the Min-Cost Ej(t);*

3.4 Optimization of the Blocking problem Induced by RTS/CTS Mechanism .

In ad hoc wireless networks, the medium access control (MAC) protocols dictate the ability of multiple devices to share the limited communication bandwidth of wireless channel. The RTS (Request-To-Send)/CTS (Clear-To-Send) mechanism is used to mitigate the number of dropped data packets via data packet collisions to eliminate the hidden terminal problem and exposed terminal problem; thus, achieving high throughput. In ad hoc networks, however, the exchange of the RTS and CTS messages while clearing the channel for communication between two devices will restrict other devices from transmitting any packet for long periods of time. This resultant effect can be seen in the increase of the end-to-end delay and the blocking problem.

3.4.1 IEEE 802.11 MAC protocols

The IEEE 802.11 specification [23] allows three kinds of physical layer: direct sequence spread spectrum (DSSS), frequency hopping spread spectrum (FHSS) and infrared (IR). In particular, the DSSS design supports data rates of 1 and 2 Mbps. Subsequently, while maintaining backward compatibility to the DSSS 802.11 specification, the 802.11b was adopted to support data rates of 5.5 and 11 Mbps, operating in the 2.4 GHz band (the ISM band). As a result, the 802.11b network can support 1, 2, 5.5 and 11 Mbps, depending on radio conditions. Another extension is 802.11a, which uses an entirely different physical layer known as orthogonal frequency division multiplexing (OFDM). 802.11a can support data rates ranging from 6 to 54 Mbps, operating in the 5.5 GHz band (the U-NII band). Moreover, the MAC protocol supports the independent basic service set (BSS), which has no connection to wired

networks (i.e., an ad-hoc wireless network), as well as an infrastructure BSS, which includes an access point (AP) connecting to a wired network.

We provide a brief description of the 802.11 MAC protocol here [23, 24]. The 802.11 specification defines five timing intervals for the MAC protocol. Two of them are considered to be basic ones that are determined by the physical layer: the short inter-frame space (SIPS) and the slot time. The other three intervals are defined based on the two basic intervals: the priority inter-frame space (PIFS) and the distributed inter-frame space (DIFS), and the extended inter-frame space (EIFS). The SIFS is the shortest interval, followed by the slot time. The latter can be viewed as a time unit for the MAC protocol operations, although the 802.11 channel as a whole does not operate on a slotted-time basis. For 802.11b networks (i.e., with a DSSS physical layer), the SIFS and slot time are 10 and 20us, respectively. The slot time of 20us is chosen to account for the signal propagation and processing delays. The PIFS is equal to SIFS plus one slot time, while the DIFS is the SIFS plus two slot times. The EIFS is much longer than the other four intervals, and is used if a data frame is received in error.

The 802.11 MAC supports two modes of operation: the Point Coordination Function (PCF) and the Distributed Coordination Function (DCF). The PCF provides contention free access; while the DCF uses the carrier sense multiple access with collision avoidance (CSMA/CA) mechanism for contention based access. Since Ad Hoc networks operate in distributed topology, we will focus on the DCF.

In the Distributed Coordination Function (DCF) mode [25], there are two access methods: the CSMA/CA, which is also referred to as basic access mechanism or 2-way handshaking, and RTS/CTS mechanism or 4-way handshaking. In the basic access

method as shown in Figure 3.8, a node transmits a Data packet if it senses the channel to be idle for DIFS period. The receiver returns an Ack if it receivers an error-free packet. If the transmitting node does not get an Ack back during SIFS, it enters into back off and retransmits after the back off time is expired.

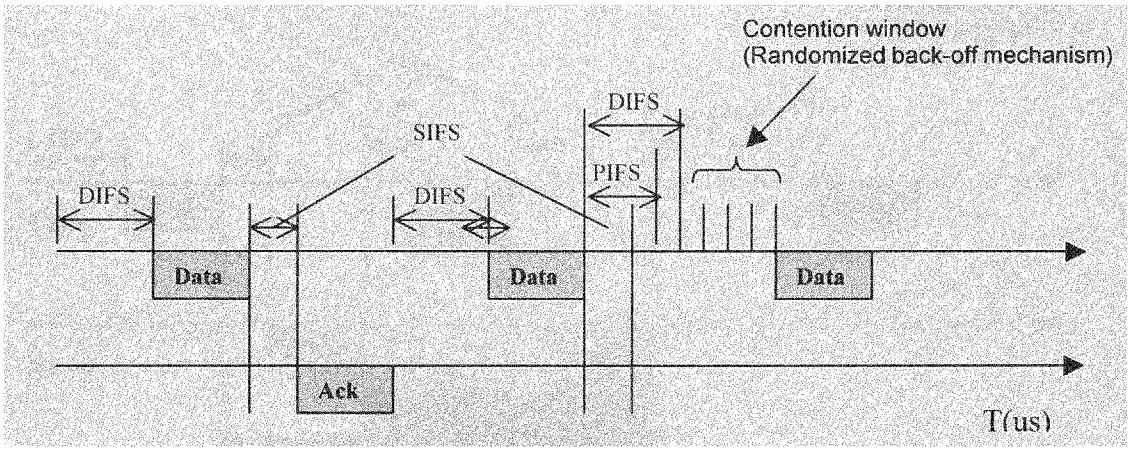


Figure 3.8 Basic Access mechanism

In the RTS/CTS mechanism as shown in Figure 3.9, when a node A (sender) has data to send to a node B (destination), it initially sends a small packet called Request-to-Send (RTS) after waiting for DIFS. The receiving node responds with a small packet called Clear-to-Send (CTS) indicating that it is ready to receive the data. The sender then completes the packet transmission. If the packet is received without error, the destination node responds with an ACK packet. If an ACK is not received after SIFS time interval, the packet is assumed to be lost and will be retransmitted. If the RTS fails, the node attempts to resolve the collision by doubling the wait period. This contention resolution method is called binary exponential Back off (BEB). In addition to the physical channel sensing, virtual carrier sensing is achieved by using time fields in the packets, which indicate to other nodes the duration of the current transmission. This time field is called Network Allocation Vector (NAV) field. All nodes that hear the RTS or CTS message

back off NAV amount of time before sensing the channel again. A detail description of the protocol can be found in [26, 27]

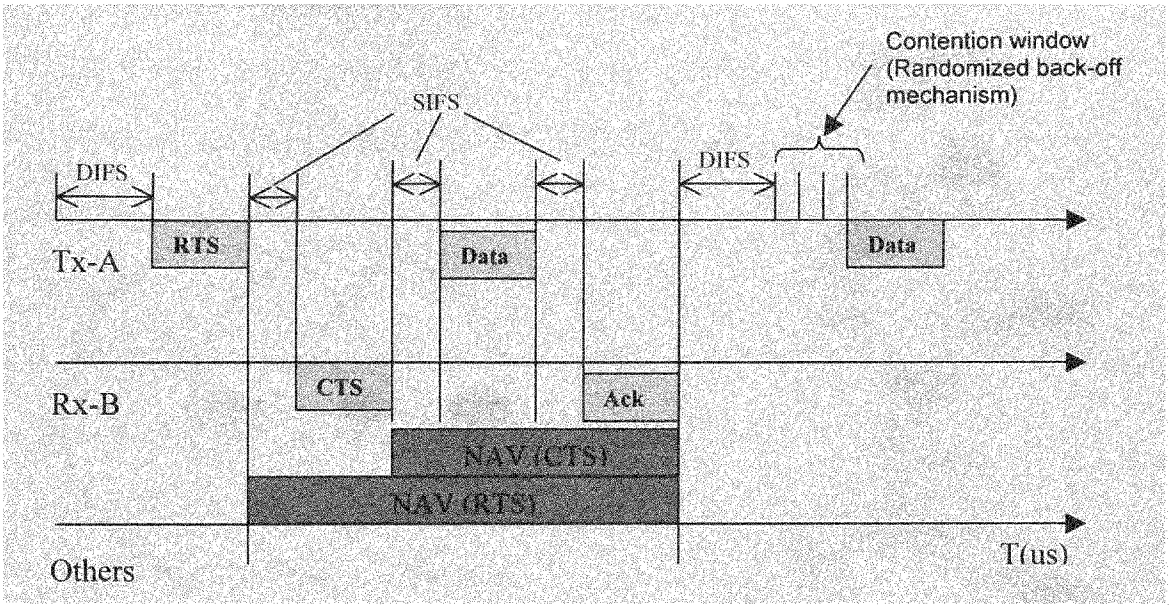


Figure 3.9 RTS/CTS mechanism

3.4.2 802.11 MAC Issues

The unique characteristics of the wireless Ad Hoc Networks make the design of the MAC protocols more challenging and different than the wired networks. The following are some of the main problems:

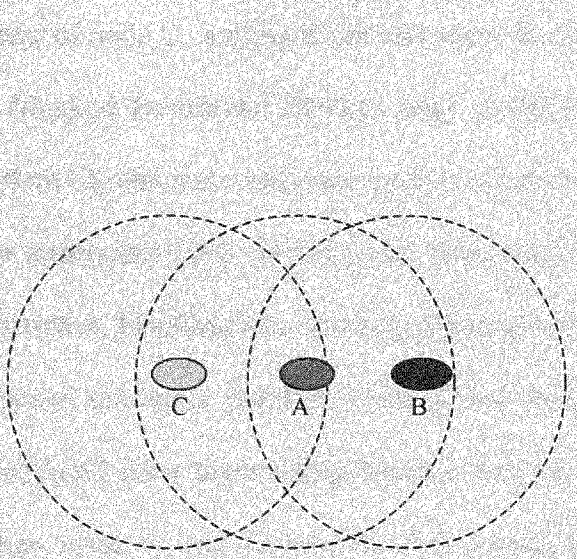


Figure 3.10a

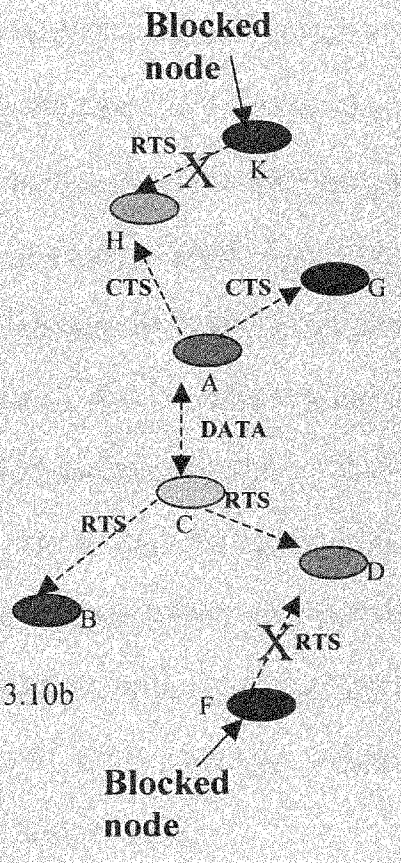


Figure 3.10b

Hidden terminals: A hidden node is one that is within the range of the intended destination but out of range of the sender [28]. From Figure 3.10a, we see that when node C is transmitting to node A, node B cannot hear the transmission of node C because it is out of range of node C. during this transmission, node B senses free medium. If node C starts transmitting to node A, a collision will occur at node A. In this case, node B is hidden from node C.

Exposed terminals: Exposed node is one that is within the range of the sender but out of range of the destination [28]. In figure 3a, consider the case that node A has data to send to node C, and node B wants also to send data to another node (not A or B). Now, node B has to wait because it senses a busy medium. However, since node B is out of range of node C, waiting is not necessary. In this case, node B is exposed to node A

Blocked terminals: Blocked node is one that is prohibited from transmit at a given instant of time since only one node is allowed to transmit at any time within the range of the transmitter [29]. If this is the case, many nodes in wireless system may be blocked. Therefore, blocking may severely affect network performance. In Figure 3.10b describes blocked terminals problem. Node C has data to send to node A. Nodes B and D are prohibited from transmitting because they receive RTS packet form node C. Similarly, when node H and G receive CTS packet form node A, they also prohibited from accessing the medium for a period of time. While the communication is taking between nodes A and C place, nodes F and K send an RTS packet to nodes D and H respectively. Since nodes D and H are blocked, they cannot respond with CTS packet. Thus, since nodes F and K do not get any response, they enter into back off mode.

3.4.3 Proposed Solution

Our solution to the blocking problem is based on defining a certain packet threshold size (δ) to be transmitted in which the 802.11 MAC protocol can switch between the RTS/CTS mechanism and the Basic Access method (the 2-way-handshaking). As we will see, this technique will dramatically reduce the effect of the blocking problem on Ad Hoc networks. Figure 3.11 describes the proposed algorithm that the 802.11 MAC protocol executes to switch between the two access mechanisms.

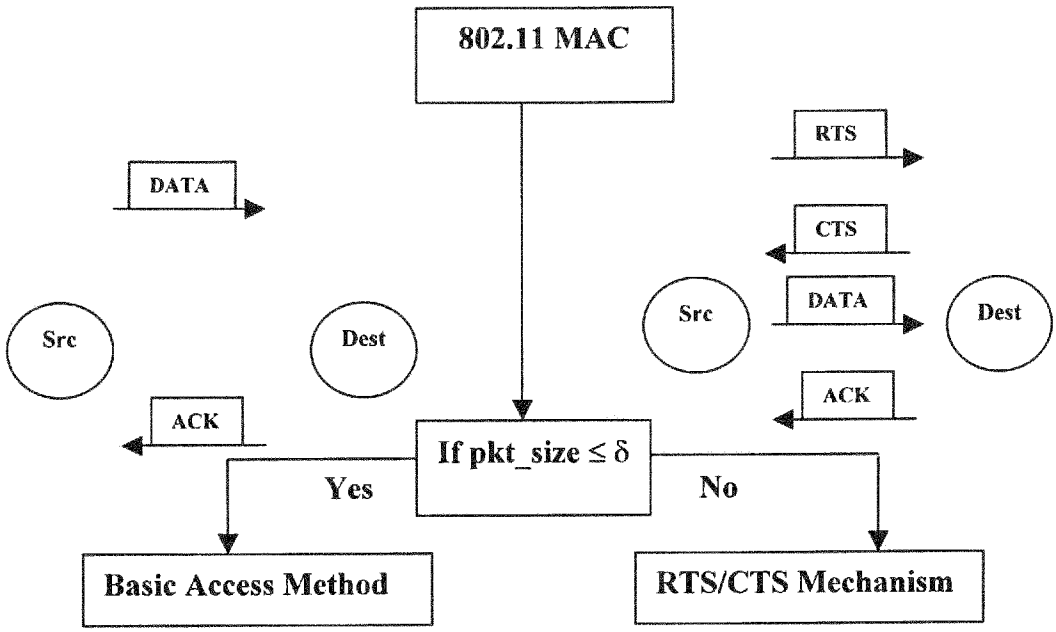


Figure 3.11 The proposed Algorithm for MAC protocol to switch between the two access mechanisms.

Chapter 4 Simulation Environment

We simulated our energy-aware routing techniques as an extension to AODV for a mobile ad hoc network. The simulation results were obtained using Global Mobile Information System Simulator (GloMoSim) [30].

4.1 Simulator

4.1.1 GloMoSim

GloMoSim is a scalable simulator environment for large wireless and wireline communication networks and uses a parallel discrete-event simulation capability provided by Parsec. GloMoSim simulates networks with up to thousand nodes linked by a heterogeneous communications capability that includes multicast, asymmetric communications using direct satellite broadcasts, multi-hop wireless communications using ad hoc networking, and traditional Internet protocols. The following table lists the GloMoSim models currently available at each of the major layers:

Table 4.1 The list of GloMoSim modes

Layer	Models
Physical (Radio propagation)	Free space, Two-Ray
Data Link (MAC)	CSMA, MACA, TSMA, 802.11
Network (Routing)	Bellman-Ford, FSR, OSPF, DSR, WRP, LAR, AODV
Transport	TCP, UDP
Application	Telnet, FTP

The node aggregation technique is introduced into GloMoSim to give signification benefits to the simulation performance. Initializing each node as a separate entity inherently limits the scalability because the memory requirements increase dramatically for a model with large number of nodes. With node aggregation, a single entity can

simulate several network nodes in the system. Node aggregation technique implies that the number of nodes in the system can be increased while maintaining the same number of entities in the simulation. In GloMoSim, each entity represents a geographical area of the simulation. Hence, the network nodes, which a particular entity represents, are determined by the physical position of the nodes.

4.1.2 Parsec

PARSEC (PARallel Simulation Environment for Complex Systems)[31] is a parallel simulation environment for complex systems which was developed at UCLA to provide researchers with more efficient and convenient ways to simulate various testbeds that cannot be easily structured in real environment such as huge wired and/or wireless networks with a number of nodes. PARSEC is a high performance version of Maise, which is based on the C language. It provides a C style interface for programming with extensions to develop parallel simulations on multiple machines. Also, PARSEC provides powerful message receiving constructs that result in shorter and more natural simulation programs.

4.2 Simulation Testbed: GlomoSim

The GloMoSim environment was used for implementing our energy-aware AODV. The operating system used was Red Hat Linux 8.2, and the platform used was an Intel Pentium 4 PC compatible, running at 1.4GHz over a local area network. The simulation environment provides variable stack size assignments for each entity, as the network grows larger in size. A common simulation clock is also provided for synchronizing operations. The two fundamental data structures provided by PARSEC are outlined below:

1. *Entities*: the notion of an entity is the same as that of a class in any object oriented language. Methods in PARSEC correspond to statement blocks that are nested send and receive operations. These statements are executed in a C style switch structure (or case structure) in that, each block executes itself when the appropriate receive operation occurs.

2. *Messages*: the data is transferred between entities via buffered messages for actual communication. Messages are time-stamped with the current simulation time to provide accurate simulations. Also, Messages are relayed through asynchronous send and receive operations that respectively deposit and remove messages from an entity's message buffer.

PARSEC provides a timeout feature that allows actions to be taken upon non-receipt of messages (leading to default action). It supports a single timeout counter, which can be easily be customized to support multiple counters. Finally, the driver entity in PARSEC starts the entire sequence of simulation. This process is asynchronous, and the scheduling of messages is done on a timestamp basis. Messages that have the same time-stamp are arbitrarily ordered. A typical PARSEC program is illustrated in appendix A. the next section describes the internal process of the GloMoSim architecture.

4.2.1 Messaging Architecture of GloMoSim

Messages are defined and employed to make it possible to exchange information between intra-layers. The messaging architecture can be thought of as being made up of two parts. The first part is the skeleton of the messaging architecture, and the other part is an example of its working. Figure 4.1 shows the block diagram of the messaging architecture. From GloMoSim's point of view, here are two message architectures. One is

the intrinsic architecture owned by GloMoSim, which can be called a generic architecture, and the other one is an optional architecture, which can be defined by users according to their specific protocols.

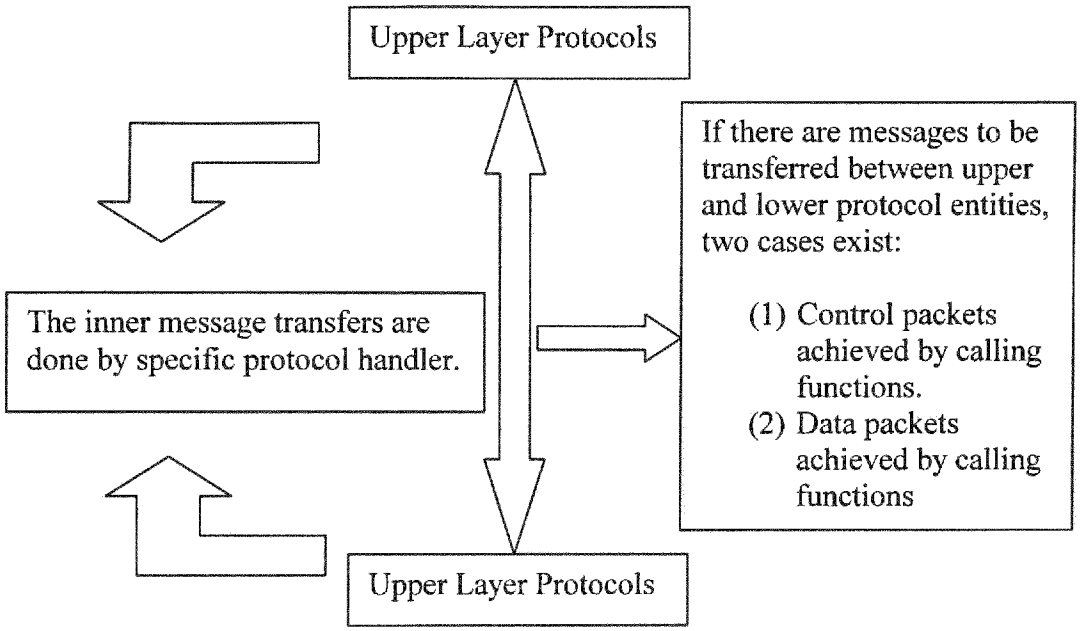


Figure 4.1 Messaging Architecture of GloMoSim

4.3 Methodology

The overall goal of this thesis in one hand was to measure and compare the energy consumption behavior of our energy-aware AODV and the classical AODV routing protocols. On the hand, the goal was also to optimize the blocking problem induced by RTS/CTS mechanism. Our basic methodology consisted of first selecting the most representative parameters for ad hoc networks. Then, based on those parameters, we simulate and evaluate all the protocols of interest.

4.3.1 Transmission Range

Because of way radio transmissions are affected by the environment in such a complex way, it is quite difficult to predict the transmission range of a node. The radio range is the average maximum distance in usual operating conditions between two nodes. There is no standard and common operating procedure to measure a range (except in free space, which is useless), so we cannot really compare different products from the ranges as indicated in mobile devices data-sheets. If we want to compare mobile nodes in term of range performance, we must look closely at the *transmitted power* and *sensitivity values*

1. *Transmitted power*: is the strength of the emissions measured in Watts (or milliWatts). Government regulations limit this power, but also having a high transmit power will also be likely to drain the batteries faster. Nevertheless, having power will help to emit signals stronger than the interferers in the band.

2. *Sensitivity*: is the measure of the weakest signal that may be reliably heard on the channel by receiver. In other words, it is able to read the bits from the antenna with a low error probability). This indicates the performance of the receiver, and the lower the value the better the hardware. Usual values are around - 80dBm. A possible methodology to determine the transmission radio range in GloMoSim would be the following:

- Set *the propagation path-loss model* (PROPAGATION-PATHLOSS parameter).
- Fix the received power of the destination antenna (RADIO-RX-THRESHOLD parameter).
- Fix the distance and calculate the transmitted power according to the selected propagation path-loss.
- Set this value to the RADIO-TX-POWER parameter

4.3.2 Mobility

The only available mobility model in GloMoSim is the Random Waypoint Mobility Model (RWPM) [7]. In this model a node randomly selects a destination from the physical terrain, and moves in the direction of that destination in a speed uniformly chosen between MOBILITY-WP-MIN-SPEED and MOBILITY-WP-MAX-SPEED parameters (defined in meter/sec). After it reaches its destination, the node stays there for a MOBILITY-WP-PAUSE time period.

In our case, we specified the parameter MOBILITY TRACE in order to indicate GloMoSim that individual movements for nodes will be taken from file specified by MOBILITY-TRACE-FILE. The MOBILITY-INTERVAL parameter is used to indicate nodes to update their position every MOBILITY-INTERVAL time period, while MOBILITY-D-UPDATE is used when a node updates its position based on the distance (in meters).

4.3.3 Energy Consumption Model

According to the specification of the NIC modeled, the energy consumption varies from 230mA in receiving mode to 330mA in transmitting mode, using a 3.3V or 5.0V energy supply. In this work we are assuming an energy supply of 5V and the energy consumption of 250mA is the same for both transmitting and receiving. These values correspond to a 2.4GHz Wave-LAN implementation of IEEE 802.11.

When a node sends or receives a packet, the network interface of the node decrements the available energy according to the following parameters:

- The specific NIC characteristics.
- The size of the packets.
- The used bandwidth.

The following equation represents the energy used (in Joules) when a packet is transmitted or received and the packet size is represented in bits:

- $Energy_{tx/rx}$ = Number of packet $_{tx/rx}$*(250*5*Packet-Size)/2*10^6

In our model, we assume the listen operation is energy free although actual equipments consume energy not only when sending and receiving but also while listening

Chapter 5 Results and Discussion

The objective of the simulation is first to check the performance of our Energy-Aware AODV with respect to the classical AODV and then, to determine the packet size threshold value in which the MAC layer will switch between the RTS/CTS mechanism and the basic access technique in order to optimize the blocking problem induced in Ad hoc networks.

5.1 EA-AODV Performance

5.1.1 Simulation Model

Using GloMoSim simulator[9], we simulated a two dimensional network of 20 nodes move around in a square area of 2000m X 2000m according to mobility model defined by GloMoSim. In our simulations, the nodes move at an average speed of 30m/sec. Each node uses IEEE 802.11 standard [4] MAC layer. The radio model is very similar to the first generation WaveLAN radios with nominal radio range of 250m. The nominal bit rate is 2 Mbps in the radio frequency of 2.4 GHz. The traffic sources start at random times towards the beginning of the simulation and stay active throughout. The sources are CBR (constant bit rate) and generate TCP packets at 10packet/sec, each being 512 bytes. Each node can transmit up to 10000 packets throughout the simulation time and simulation is run for 1000 seconds. The following performance metrics are evaluated. These metrics are typical metrics usually evaluated for analyzing performance of routing protocols and the efficiency of Energy-Aware AODV.

- *Average energy remaining*: measured as the average of remaining battery energy (in Joules) at the end of the experiment.

- Average End-to-End Delay: measured as the average end-to-end latency of data packets

5.1.2 Results and Discussion

In figure 5.1, we observe that from the beginning of the simulation up to 200sec, the remaining energy is decreasing at the same rate for both energy-aware AODV and the classical AODV. At this point the average remaining energy of the network is still above 90% of the total initial energy in which the energy-aware AODV functions exactly like the classical AODV by choosing the shortest path, the path with least *hop count.* However, as the total initial energy drops by 10%, we see a dramatic decrease of the total energy in the classical AODV comparing to the energy-aware AODV, which also decreases but at slower rate and that is because the energy-aware AODV chooses the path with a high remaining energy instead with least hop count. Also, we observe that the remaining energy at 800 sec of the simulation time is much higher for energy-aware AODV than the classical AODV. As a result, the energy-aware AODV prolong the network lifetime by 30%.

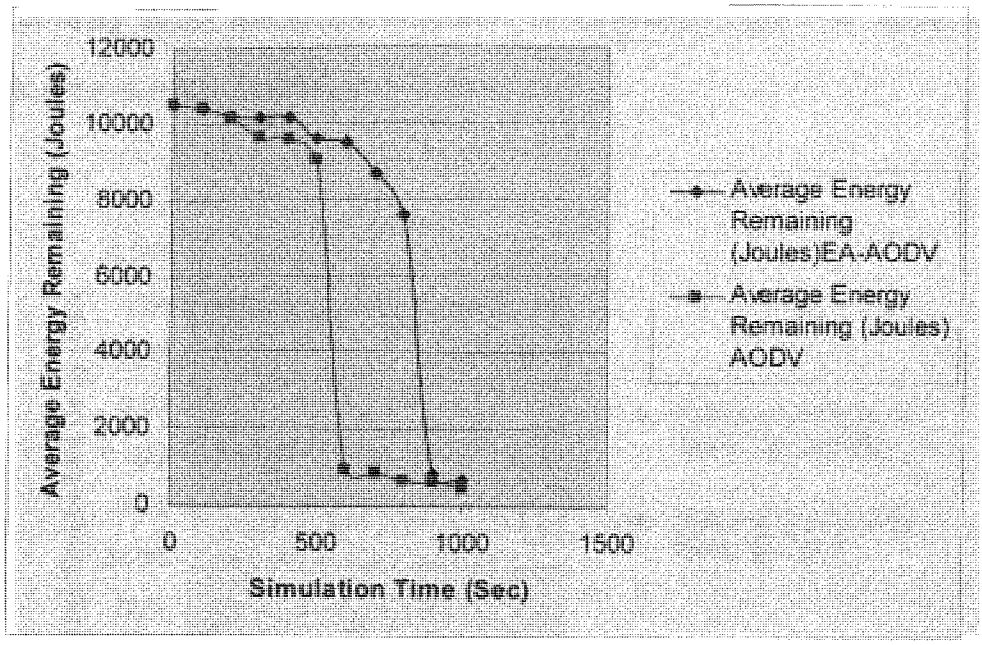


Figure 5.1 The Average Energy Remaining Vs Simulation Time for EA-AODV and AODV

Similar argument applies in the analysis of the average end-to-end delay plot in figure 5.2. The average end-to-end delay of both the energy-aware AODV and the classical AODV is increasing at the same rate up to 200 sec of the simulation time. At this point as was mentioned earlier, the average remaining energy of the network is still above 90% of the total initial energy in which the energy-aware AODV functions exactly like the classical AODV by choosing the shortest path, the path with least *hop count*. Also, from figure we observe that as the simulation time progresses, the average end-to-end delay for energy aware AODV is increasing at slightly higher rate than the classical AODV. This is because the energy-aware AODV chooses paths with higher average remaining energy, which are not necessary, the shortest ones. Moreover, we see from figure that average

end-to-end delay at the end of the simulation is (15%-20%) higher for energy-aware AODV than the classical AODV.

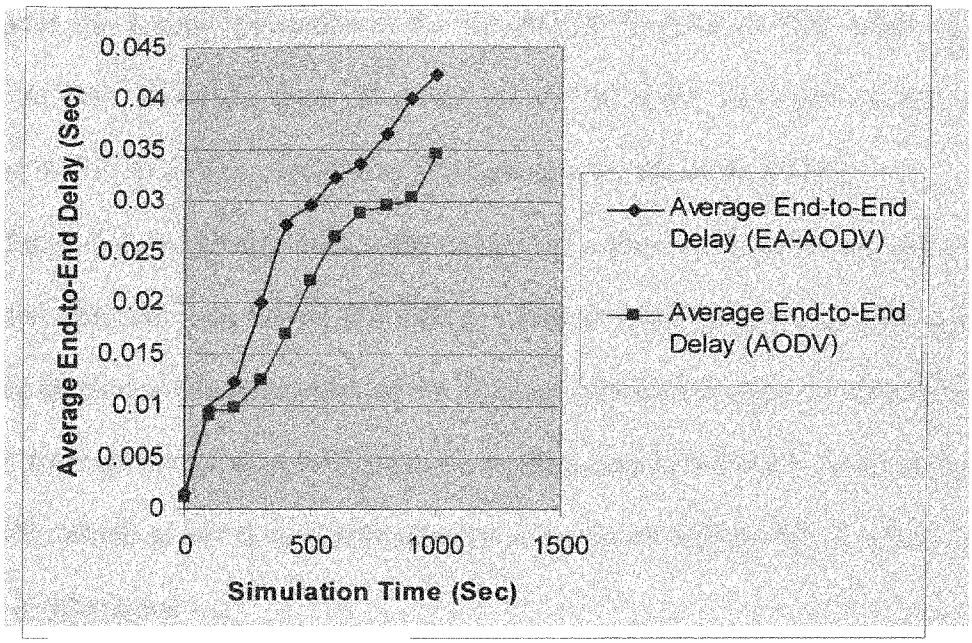


Figure 5.2 The Average End-to-End Delay Vs Simulation Time for EA-AODV and AODV

Figure-5.3 shows the total average energy consumed of all nodes with respect to the number of nodes in the network. We see that as the number of nodes increases, the average energy consumed for both the classical AODV and EA-AODV decreases; because, it should be noted here that the number of nodes are being increased while the total area of the network remains the same. This means that as the number of nodes increases, more nodes are participating in routing packets and nodes uses less power to transmit packet due to the short distances between each other. Also, as the number of nodes increase, a destination node will have more routes from which to choose the optimal route, the route with minimum cost, to unicast the RREP.

Also, we observe that AODV flatters out at around 70 nodes, with the average energy consumed being almost the same On the other hand, as the number of nodes increases, the total average energy consumed decreases for EA-AODV. Additionally, we see that up to 40 nodes the energy consumption for EA-AODV is 10% to 15% lower than the classical AODV. However, from 50 nodes up to 100 nodes the total average energy consumed by EA-AODV is 20% to 40% less than the classical AODV.

From figure-5.4, we observe that as the number of nodes increases we see that average end –to-end delay for both EA-AODV and the classical AODV increases at the same rate and up to 60 nodes the difference of the average end-to-end delay is not significant. However, from 60 nodes up to 100 nodes the average end-to-end delay is 15% higher for EA-AODV, which is not that significant comparing to the energy the classical AODV consume at 100 nodes.

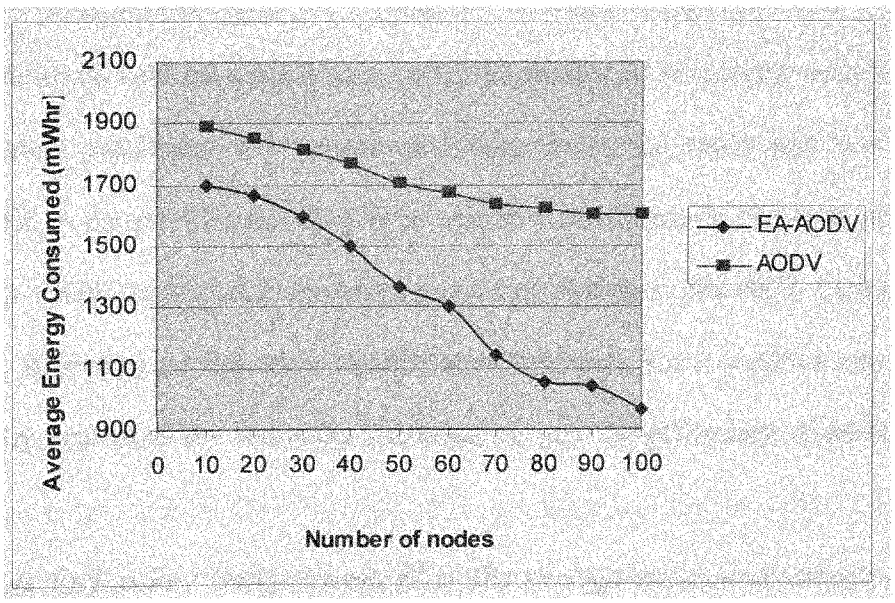


Figure 5.3 The Average Energy Consumed Vs the number of nodes in the network for EA-AODV and AODV.

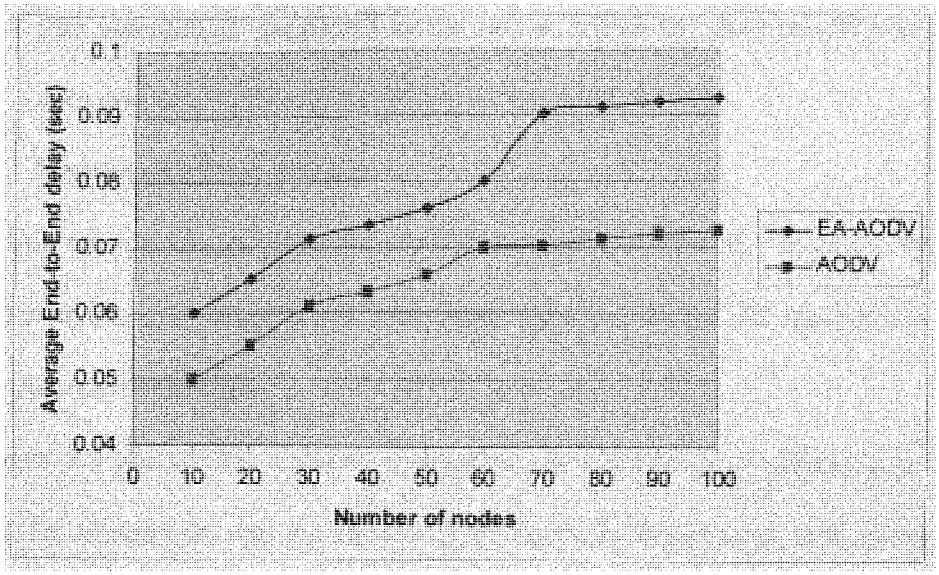


Figure 5.4 The Average End-to-End Delay Vs the number of nodes in the network for EA-AODV and AODV.

5.2 The blocking problem optimization

5.2.1 Simulation Model

Using GloMoSim simulator [9], we simulate a two dimensional network of 20 nodes. Every node transmits with the same power, using omni-directional antenna with the same gain and receiver sensitivity. All transmission experiences the same path loss versus distance profile. A constant bit rate (CBR) is used to simulate traffic and each node can transmit up to 10000 packets at a transmission rate of 2Mbps in the radio frequency of 2.4GHz. The following performance metrics are evaluated. These metrics are typical metrics usually evaluated for analyzing performance and the efficiency of MAC layer protocols.

- *Average End-to-End Delay*: measured as the average end-to-end latency of data packets

- *The throughput*: measured as the rate at which the data can be sent through the network

5.2.2 Results and Discussion

In this section we compare the performance of the network based on the throughput and the end-to-end delay when each node uses the RTS/CTS mechanism against the Basic Access mechanism.

Figure 5.3 shows the network throughput as a function of packet size when using RTS/CTS mechanism and the Basic Access method. We observe that as the packet size increases the network throughput of both access methods is almost the same. However, at the packet size of 1500 bits, we see that the throughput increases rapidly when using RTS/CTS mechanism versus the Basic Access mechanism in which the network throughput increases but at very small rate. Another important aspect of network performance is the end-to-end delay. Figure 5.4 shows the average end-to-end delay as the packet size increases. We observe that the difference in the two access methods is not significant up to a load of about 1500 bits. After that, however, the average end-to-end delay increases at a faster rate if RTS/CTS mechanism is used and that is due to the transmission RTS and CTS packets. Therefore, we conclude from our results that a packet size of 1500 bits is an optimal threshold packet size in which a MAC protocol can switch from the Basic Access method to the RTS/CTS mechanism when the packet size is 1500 bits or more. Taking this approach, the network performance will improve by solving hidden and exposed terminals using RTS/CTS mechanism and blocking problem using the Basic Access mechanism.

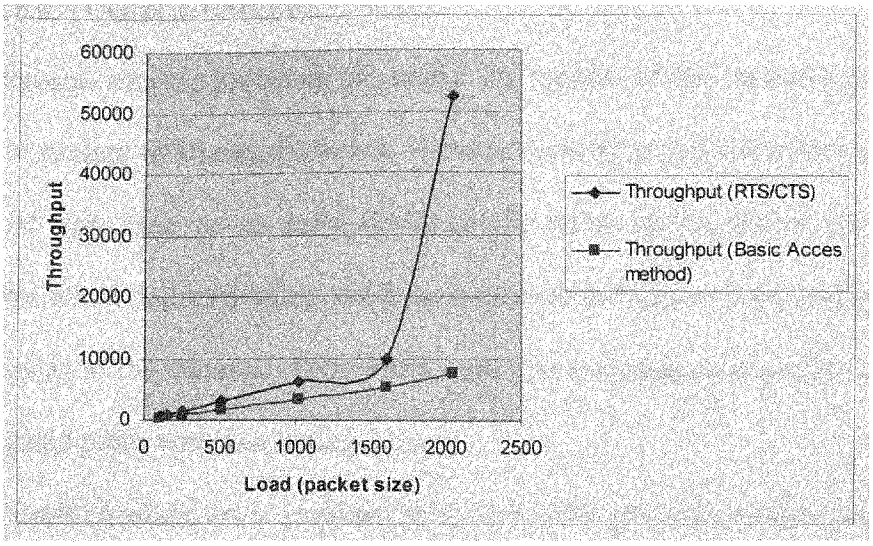


Figure 5.5 Throughput Comparison between the RTS/CTS mechanism and the Basic Access method

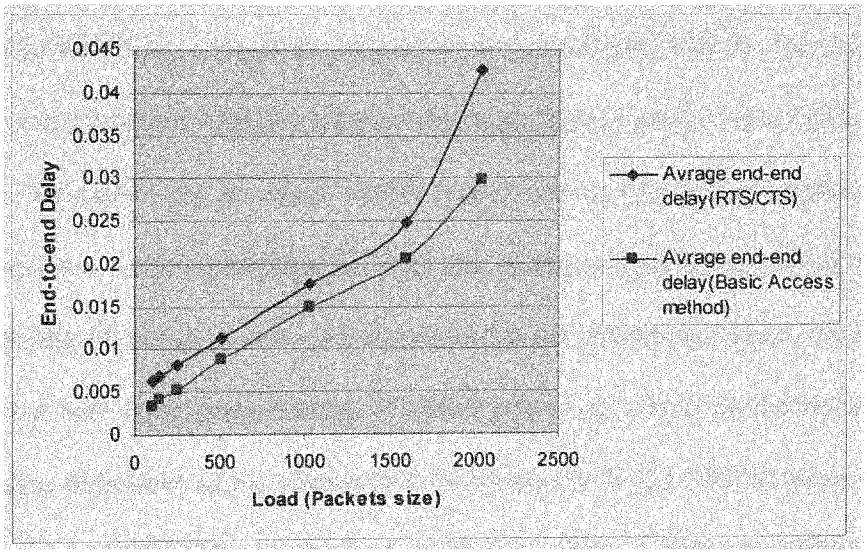


Figure 5.6 Average End-to-End Delay: Comparison between the RTS/CTS mechanism and the Basic Access method.

Chapter 6 CONCLUSION

On-demand routing protocols are useful for mobile ad hoc network environment for their low routing overhead. However, if battery energy is not taken into account in their design, it may lead to an early depletion for some nodes, which may also lead to premature network partitioning. We have proposed an energy-aware routing technique as an extension to the classical AODV that uses a new routing cost metric to ovoid the use of nodes and paths with low battery power.

The result obtained from implementing our technique is favorable and encouraging. Performance evaluation using a GloMoSim simulator shows that the longevity of the network is extended by 30%. There is a slight detrimental effect on the average end-to-end delay, which is (15%-20%) higher for energy-aware AODV. Overall, we conclude that the energy-aware AODV demonstrates significant benefits in increasing the network lifetime. We expect this protocol will be used in ad hoc networking applications.

Moreover, we have demonstrated that the implementation of RTS/CTS mechanism results to the blocking problem in which nodes are prohibited from transmitting even if no near by node is not transmitting. We have proposed a simple solution based on certain packet size threshold to be transmitted in which the 802.11 MAC protocol can switch between the RTS/CTS mechanism and the Basic Access method (the 2-way-handshaking) to improve network performance and reduce the effect of the blocking problem. The simulation results have showed that a packet size of 1500 bit is an optimal threshold value.

6.1 Future Work

So far our cost metric for energy-aware AODV is only implemented in simulations. It would be more convincing results if the protocol is tested in a prototype system. Also, it would be more interesting if our cost metric were implemented in other On-demand routing protocols such DSR and CBRP to see which protocol performs well with out cost metric.

6.1.2 Security

Due to the nature of ad hoc networks, which use free air as a medium for communication, security is a major concern. Information sent in ad hoc routes can be protected in some way but since multiple nodes are involved, the relaying of packets has to be authenticated by recognizing the originator of the packet and flow ID.

6.1.3 Service Location, Provision, and Access

While protocols are important for the proper operation of an ad hoc wireless network, service location, provision, and access are equally important. Should we continue to assume that the traditional client/server RPC (remote procedure call) paradigm is appropriate for ad hoc networks? Ad hoc networks comprise heterogeneous devices and machines and not every one is capable of being a server. The concept of a client initiating task requests to a server for execution and awaiting results to be returned may not be attractive limitations in bandwidth and power. Also, how can mobile device access a remote service in ad hoc network? How can a device that is well equipped advertise its desire to provide services to the rest of the members in the network? All these issues demand research.

6.1.4 Media Access

Unlike cellular networks, there is a lack of centralized control and global synchronization in ad hoc wireless networks. Hence, TDMA and FDMA schemes are not suitable. In ad hoc wireless networks, since multiple mobile ad hoc nodes share the same media, access to the common channel must b made in a distributed fashion, through the presence of a MAC protocol. Given the fact that there is no static node, nodes cannot rely on a centralized coordinate. The MAC protocol must contend for access to the channel while at the same time avoiding possible collisions with neighboring nodes. The presence of mobility, hidden terminals, exposed nodes problems, and blocking problem must be accounted for when it comes to designing MAC protocols for ad hoc wireless networks.

6.1.5 Spectrum Allocation

Regulations regarding the use of radio spectrum are currently under the control of the FCC. Most experimental ad hoc networks are based on the ISM band. To prevent interference, ad hoc networks must operate over some form of allowed or specified spectrum range. Most microwave ovens operate in the 2.4GHz band, which can therefore interfere with wireless LAN systems. Frequency spectrum is not only tightly controlled and allocated, but it also needs to be purchased. With ad hoc networks capable of forming ad deforming on the fly, it is not clear who should pay for this spectrum.

REFERENCES

[1]Toh C-K, Ad Hoc Mobile Wireless Networks: Protocols and system, Prentice Hall PTR, Upper Saddle River, N.J., 2002.

[2]Z. J. Hass, M. R. Pearlman, "The performance of Query Control Schemes for the Zone outing Protocol," ACM SigComm, 1998.

[3]" ANSI/IEEE Std 802.11-1999 Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, " 1999.

[4]Dennis J. Baker and Anthony Ephremides, " The architectural organization of a mobile radio network via a distributed algorithm," *in IEEE Transactions on Communications*, 1981.

[5] Anthony Ephremides, Jeffery E. Wieselthier, and Dennis J. Baker, "A design concept for reliable mobile radio networks with frequency hopping signaling", *in Proceedings of the IEEE,* 1987.

[6] Volkan Rodoplu and Teresa H. Meng, "Minimum energy mobile wireless networks," *in proceedings of International Conference on communications*, 1998

[7] J. H. Chang and L. Tassiulas, "Routing for maximum system lifetime in wireless ad-hoc networks," *in Proceedings of 37th Annual Allerton Conference on Communication, Control and computing,* 1999.

[8] J. H. Chang and L. Tassiulas, "Energy conserving routing in wireless ad-hoc networks," *in IEEE Infocom* 2000.

[9] A. Michail and A. Ephremides, " Energy efficient routing for connection-oriented traffic in ad-hoc wireless networks," *in Personal Indoor and Mobile Radio Communications* (PIMR) 2000.

[10] Juan-Carlos Cano and Pietro Manzoni, "A performance comparison of energy consumption for mobile ad hoc network routing protocols," *in Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, 2000.

[11] C.E. Perkins and E. Royer, "Ad hoc on-demand distance vector (AODV) routing," *in proceedings of he 2nd IEEE workshop on Mobile computing systems and Applications*, 1999

[12] D. B. Johnson D. A. Maltz, and J. Broch, "The dynamic source routing protocol for mobile ad hoc network," *in Internet Draft, MANET Working group, draft-ietf-manet-dsr-03.txt*, 1999

[13] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," *in Proceedings of Sigcomm*, 1994.

[14] V. Park and M. Corson, "A highly adaptive distributed routing algorithm for mobile wireless networks," in *proceedings of IEEE INFOCOM* 1997.

[15] J. Jubin and J. Tornow, "The DARPA Packet Radio Network Protocols," *Proc. IEEE*, vol. 75, no. 1, 1987, pp. 21–32.

[16] C.E. Perkins and P. Bhagwat, "Highly Dynamic Destination-SequencedDistance-Vector Routing (DSDV) for Mobile Computers", Comp. Comm. Rev., Oct. 1994, pp.234-244. http://www.svrloc.org/~charliep/txt/sigcomm94/paper.ps *Discusses Destination-Sequenced Distance-Vector Routing Algorithm.*

[17] S. Murthy and J.J. Garcia-Luna-Aceves, "An Efficient Routing Protocol for Wireless Networks", ACM Mobile Networks and App. J., Special Issue on Routing in Mobile Communication Networks, Oct. 1996, pp. 183-97. http://www.ics.uci.edu/~atm/adhoc/paper-collection/aceves-routing-winet.pdf *Discusses Wireless Routing Protocol.*

[18] C.-C. Chiang, "Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel" Proc. IEEE SICON'97, Apr.1997, pp.197-211. http://www.ics.uci.edu/~atm/adhoc/paper-collection/gerla-routing-clustered-sicon97.pdf *Discusses Clustered Gateway Switch Routing Algorithm.*

[19] Mingliang Jiang, Jinyang Li, Y.C. Tay, "Cluster Based Routing Protocol" August 1999 IETF Draft, 27 pages. http://www.ietf.org/internet-drafts/draft-ietf-manet-cbrp-spec-01.txt *Discusses Cluster Based Routing Protocol.*

[20] Charles E. Perkins, Elizabeth M. Royer, Samir R. Das, "Ad Hoc On-demand Distance Vector Routing", October 99 IETF Draft, 33 pages. http://www.ietf.org/internet-drafts/draft-ietf-manet-aodv-04.txt

[21] David B. Johnson, Davis A. Maltz, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks" October 1999 IETF Draft, 49 pages. http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-03.txt

[22] David B. Johnson, Davis A. Maltz, "Dynamic Source Routing in Ad Hoc Networks", Mobile Computing, T. Imielinski and H. Korth, Eds., Kulwer, 1996, pp. 152-81.

[23] IEEE 802.11, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification,"1997.

[24] B. O'Hara and A. Petrick, *IEEE 802.11 Handbook,* IEEE Press, New York, 1999.

[25] " ANSI/IEEE Std 802.11-1999 Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, " 1999.

[26] B. Crow et al., " Performance of IEEE 802.11 Wireless Local Area Networks," *SPIE, vol.* 2917. pp. 480-91.

[27] B. P. Crow et al., IEEE 802.11: Wireless Local Area Networks, *IEEE commn. Mag.*, vol. 35, no. 9, Sept. 1997, *http://www.comsoc.org/ci/,* pp. 116-26.

[28] V. Bharghavan, " A New Protocol for Medium Access in Wireless PacketNetworks," publication of the Timely group, 1997.

[29] S. Ray, B. Jeffery, Carruthers, and D. Starobinski, " RTS/CTS-Induced Congestion in Ad Hoc Wireless LANs." *http:// www.ipsit.bu.edu.*

[30] GloMoSim: *Global Mobile Information Systems Simulation Library.http://pcl.cs.ucla.edu/projects/glomosim*

[31] R. A. Meyer. "PARSEC User Manual Release 1.1", Parallel Computing Laboratory, University of California, Los Angeles, 1998.

Appendix A

Appendix A contains a sample PARSEC program

Sample PARSEC program

A typical parsec program contains messages and entities. The messages are declared with their appropriate data structures, and are used for information transfer only. Hence, They do not contain methods for execution. This followed by the entity declaration. Entities are analogous to classes that hold and process information, and send and receive messages in a continuous loop until the simulation clock expires. Timeouts can be specified within this loop for periodic events. He clock is set to the maximum simulation duration in he main driver function, which is the initial point of control.

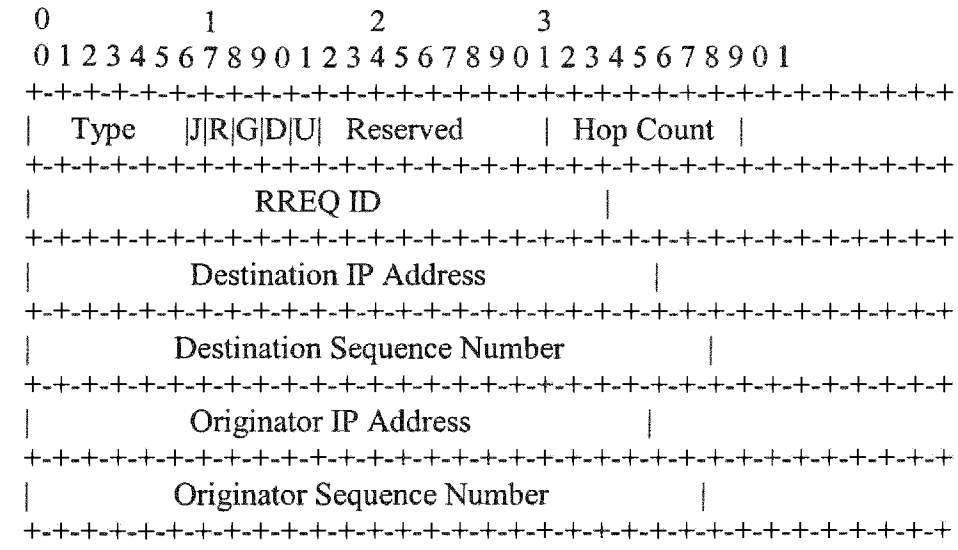
```
/* entity definition*/
entity Manager(int maxResiurces) stacksize (2000)
{
int unitsAvailable = maxResources;
int totalRequest – 0;
....
Finalize
{
printf("Manager got %d total request.\n", totalRequest);
}
}
/*entity creation*/
ename s1,s2;  /*entity identifier*/

/*instantiation*/

s1 = newManager (5);
s2 = new Manager (10);
/*Messages*/
message Data
{
int value,ename sender;
};
message ack {};
```

```
entity node (int node_no)
{
int num_pkts;
mssage Data data; /*declaration*/
num_pkts = data.value; /*referencing data*/
}
/* initiate simulations*/
entity driver
{
/*parameters*/
ename node;
setmaxclock(200);
send data to node;
....
}
```

Appendix B

Appendix B contains information of the fields in the route request, route reply and route error packets.

Route Request (RREQ) Message Format



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

Type 1

J Join flag; reserved for multicast.

R Repair flag; reserved for multicast.

G Gratuitous RREP flag; indicates whether a gratuitous RREP should be unicast to the node specified in the Destination IP Address field (see sections 6.3, 6.6.3).

D Destination only flag; indicates only the destination may respond to this RREQ (see section 6.5).

U Unknown sequence number; indicates the destination sequence number is unknown (see section 6.3).

Reserved Sent as 0; ignored on reception.

Hop Count The number of hops from the Originator IP Address to the node handling the request.

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

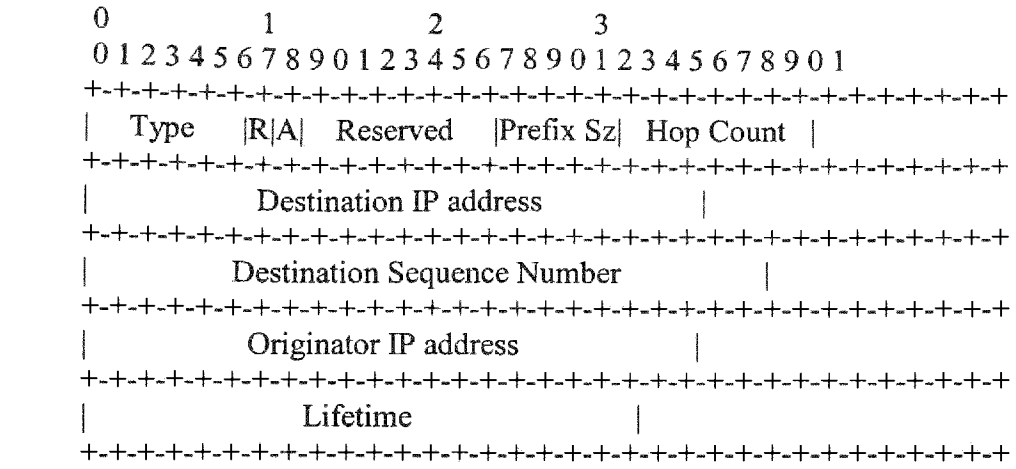
Destination IP Address
The IP address of the destination for which a route is desired.

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

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

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

Route Reply (RREP) Message Format



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

Type 2

R Repair flag; used for multicast.

A Acknowledgment required; see sections 5.4 and 6.7.

Reserved Sent as 0; ignored on reception.

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

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

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

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

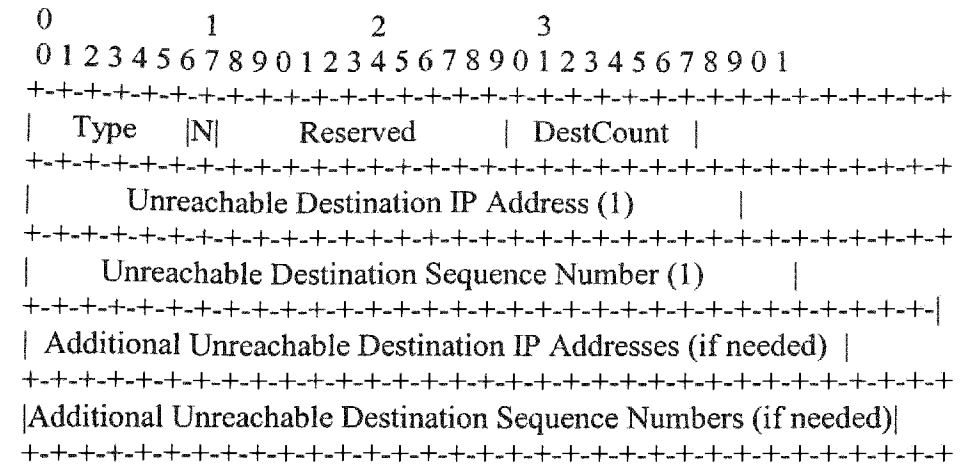
Originator IP Address
The IP address of the node which originated the RREQ for which the route is supplied.

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

Note that the Prefix Size allows a subnet router to supply a route for every host in the subnet defined by the routing prefix, which is determined by the IP address of the subnet router and the Prefix Size. In order to make use of this feature, the subnet router has to guarantee reachability to all the hosts sharing the indicated subnet prefix. See section 7 for details. When the prefix size is nonzero, any routing information (and precursor data) MUST be kept with respect to the subnet route, not the individual destination IP address on that subnet.

The 'A' bit is used when the link over which the RREP message is sent may be unreliable or unidirectional. When the RREP message contains the 'A' bit set, the receiver of the RREP is expected to return a RREP-ACK message. See section 6.8.

Route Error (RERR) Message Format



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

Type 3

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

Reserved Sent as 0; ignored on reception.

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

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

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

The RERR message is sent whenever a link break causes one or more

destinations to become unreachable from some of the node's neighbors.