AN ALTERNATIVE ROUTING MECHANISMS FOR MOBILE AD-HOC NETWORKS

AWOS KHAZAL ALI

A project report submitted in partial fulfillment of the
requirements for the award of the degree of
Master of Science (Computer Science)

Faculty of Computer Science & Information Systems
Universiti Teknologi Malaysia

OCTOBER 2008

My Lord, Allah, thanks for giving me a working mind with eyes and hands,

That is more than I deserve

My parents, I cannot express how much I do LOVE YOU

My family members, all of you, thanks for the great support

My friends, I thank Allah for gifting me you

ACKNOWLEDGEMENT

"In the name of Allah, the most Beneficent, the most Merciful"

In preparing this project report, I've been in touch with many people who have given me a great support and beneficial ideas that lead to the success of it. At the very beginning, I'd thank my parents for their unlimited support to me and my research, I thank them very much, and I ask Allah the almighty to grant them the Paradise. I'd like to thank my sisters and my brother for their great support.

I'd sincerely thank my supervisor Dr. Shukor abd. Razak for his time and facilities he has offered me, and the great advice he used and still giving me.

Not to forget my fellow postgraduate students who have spared some of their time praying for my success, I thank you sincerely and wish you more and brighter success in your lives.

ABSTRACT

Routing is the essential task of transferring packets form source node to destination, or can be described as the process of path finding. A lot of routing mechanisms have been proposed for wired and wireless networks and some of them have been widely used. In Mobile Ad Hoc Network (MANET), every node in the network acts as a router, forwarding packet from one node to another. Since Ad Hoc networks are wireless and the nodes often battery driven, it is very important that the routing protocol in use can handle a large degree of node mobility and at the same time be very energy efficient. The goal of this study is to propose a hybrid routing protocol for MANET based upon existing protocols. NS2 was used to perform the simulation and evaluate the performance of proposed protocol and compare it with existing protocols.

ABSTRAK

Routing adalah proses asas dalam penghantaran paket-paket dari nod asal ke destinasi, atau boleh digambarkan sebagai suatu proses pencarian jalan atau *route*. Terdapat pelbagai mekanisma *routing* yang telah dicadangkan bagi rangkaian berwayar dan tanpa wayar dan kebanyakkan daripadanya telah pun digunakan secara meluas. Di dalam rangkaian *Mobile Ad Hoc Network* (*MANET*), setiap nod di dalam rangkaian bertindak sebagai penentu arah laluan (*router*), menghantar paket dari satu nod ke nod yang lain. Disebabkan rangkaian-rangkaian *ad hoc* adalah tanpa wayar dan nod-nod kebanyakkannya digerakkan menggunakan bateri, adalah penting protokol *routing* yang digunakan dapat menangani kadar pergerakan yang tinggi dan pada masa yang sama mempunyai penggunaan tenaga yang cekap. Matlamat bagi kajian ini ialah untuk mencadangkan mekanisma penukaran (*switching)* bagi protokol *routing* secara *hybrid* bagi rangkaian-rangkaian *MANET* berdasarkan protokol-protokol yang sedia ada. *NS2* telah digunakan untuk menjalankan simulasi dan mengukur tahap perlaksanaan protokol yang telah dicadangkan dan membandingkannya dengan protokol-protokol yang sedia ada.

TABLE OF CONTENTS

CHAPTER	TITLE	PAGE
	DECLARATION	II
	DEDICATION	III
	ACKNOWLEDGEMENT	IV
	ABSTRACT	V
	ABSTRAK	VI
	TABLE OF CONTENTS	VII
	LIST OF TABLES	XI
	LIST OF FIGURES	XII
	LIST OF APPENDICES	XIII
1	**PROJECT OVERVIEW**	
	1.1. Introduction	1
	1.2. Problem Background	4
	1.3. Problem Statement	5
	1.4. Project Objectives	5
	1.5. Project scopes	6
	1.6. Project Organization	7
2	**LITERATURE REVIEW**	
	2.1. Introduction	8

CHAPTER	TITLE	PAGE

2.2. Routing 9

2.3. Classification of routing Protocols in MANET 10

2.3.1 Table Driven Routing Protocols (Proactive) 12

2.3.1.1. Destination - Sequence Distance - Vector Routing (DSDV) 13

2.3.1.2. Cluster head Gateway Switch Routing (CGSR) 14

2.3.1.3. The Wireless Routing Protocol (WRP) 17

2.3.1.4. Fisheye State Routing (FSR) 19

2.3.1.5. Optimized Link State Routing (OLSR) 20

2.3.1.6. Summary of Table-Driven Ad-Hoc Routing Protocols (Proactive) 21

2.3.2 Source Initiated On Demand Routing Protocols (Reactive) 22

2.3.2.1. Ad hoc On Demand Distance Vector Routing (AODV) 23

2.3.2.2. Dynamic Source Routing (DSR) 25

2.3.2.3. Temporally Ordered Routing Algorithm (TORA) 26

2.3.2.3. Associativity Based Routing (ABR) 28

2.3.2.4. Signal Stability Routing (SSR) 30

2.3.2.5. Summary of On-Demand Ad-Hoc Routing Protocols (Reactive) 31

2.3.2 Hybrid routing protocols 32

2.3.3.1.Zone Routing Protocol (ZRP) 33

2.4. Summary 34

CHAPTER	TITLE	PAGE

3 RESEARCH METHODOLOGY

3.1. Introduction 37
3.2. Project Structure 38
3.3. Operational Framework 39
3.3.1. Initial Study 40
3.3.2. Reviewed Related Literature 40
3.3.3. Propose New Routing Protocol 40
3.3.4. Simulation 40
3.3.5. Comparison 41
3.4. Software Requirements 43
3.5. Summary 43

4 RESULTS AND DISCUSSION

4.1. Introduction 44
4.2. Simulation setup 44
4.2.1. Network Simulator (NS-2) 45
4.2.2. Protocols Implementation 47
4.2.2.1. AODV Simulation Scenario 47
4.2.2.2. DSDV Simulation Scenario 48
4.2.2.3. Hybrid Simulation Scenario 48
4.2.3. Metrics 49
4.3. Results 49
4.3.1. AODV 50
4.3.2. DSDV 52
4.3.3. Hybrid 55
4.3.4. End-to-End Delay 57
4.4. Discussion 59
4.4.1. AODV vs. DSDV 60

CHAPTER	TITLE	PAGE

	4.4.2. Hybrid vs. AODV	61
	4.4.3. Hybrid vs. DSDV	62
	4.4.4. Hybrid Protocol Performance	62
	4.5. Summary	68
5	**CONCLUSION AND FUTURE WORK**	
	5.1. Introduction	69
	5.2. Achievement	69
	5.3. Challenging	70
	5.4. Future Work	71
REFERENCES		72
APPENDIXES A – B		75-80

LIST OF TABLES

TABLE NO.	TITLE	PAGE
2.1	Differences of mobile ad-hoc routing protocols (Proactive)	22
2.2	Differences of mobile ad-hoc routing protocols (On-Demand)	32
2.3	The main differences in mobile ad-hoc routing protocols	35
4.1	Simulation setup	45
4.2	Results	59

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE
1.1	Example of an infrastructure wireless network	2
1.2	Example of a mobile ad hoc network with three participating nodes	3
2.1	Classification of ad hoc routing protocols	11
2.2	CGSR: routing from node 1 to node 9	16
2.3	Scopes of Fisheye State Routing (FSR)	20
3.1	Project Structure Steps	38
3.2	Project Operational Framework	39
3.3	Research Methodology Steps	42
4.1	Relationship between C++ and OTcl script	46
4.2	packets Dropped Ratio in AODV	50
4.3	Packets generation ratio in AODV	51
4.4	Packets loss ratio in AODV	52
4.5	Packets drop ratio in DSDV	53
4.6	Packets generation ratio in DSDV	54
4.7	Packets loss ratio in DSDV	55
4.8	Packets loss ratio in Hybrid approach	56
4.9	Packets generation ratio in Hybrid approach	56
4.10	Packets loss ratio in Hybrid approach	57
4.11	End-to-End delay	58
4.12	Switching mechanism code from DSDV to AODV in C++	65
4.13	Switching mechanism code from AODV to DSDV in C++	67

LIST OF APPENDIXES

APPENDIX NO.	TITLE	PAGE
A	OTcl Simulation Code	75
B	C++ Hybrid Code	79

CHAPTER 1

Project Overview

1.1. Introduction:

In Latin, ad hoc literally means “for this”, and is often applied in terms of network where devices may be added and integrated using wireless technology [2]. Wireless networks have become popular in the past few years, when they are being adapted to enable mobility and wireless devices became popular such as Mobile hand phone and laptop computers

There are two kinds of mobile wireless networks. The first is known as infrastructure network, with mobile nodes and base station. The base station is responsible for routing, security and all issues in this kind of network. Typical applications for this type of wireless network include wireless local area networks (WLANs). Figure 1.1shows an example of infrastructure wireless network. The second type of mobile wireless network is the infrastructure-less mobile network, commonly known as the ad hoc network [3].

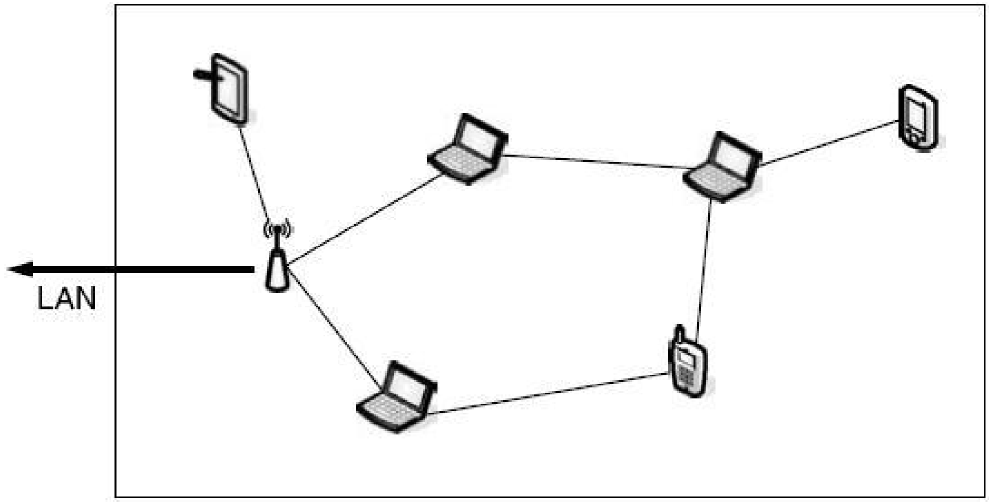

Figure 1.1: Example of an infrastructure wireless network.

Mobile ad hoc networks (MANETs) are collections of mobile nodes, dynamically forming a temporary network without pre-existing network infrastructure or centralized base station. These nodes can be arbitrarily located and are free to move randomly at any given time, thus allowing network topology and interconnections between nodes to change rapidly. Node mobility can vary between network to other depending on the particular network's structure and purpose.

As a general rule, high mobility usually results in low link capacity, whereas low mobility leads to high capacity links. An ad hoc network is usually a self-organizing and self-configuring "multi-hop" network which does not require any fixed infrastructure such as transceiver base stations or even cables. Mobile devices (e.g. notebook computers, PDAs, and cell phones) with wireless radio equipment are supposed to communicate with each other, without the help of any fixed devices [1]. In an ad hoc network, all nodes are dynamically and arbitrarily located, and are required to relay packets for other nodes in order to deliver data across the network. There is no router like in wired network, thus the node it self operate the router job.

Ad hoc networks are suited for use in situations where infrastructure is either not available, not trusted, or should not be relied on in times of emergency. A few examples include: military solders and equipments in the battlefield ("open environment"), sensor networks for various research purposes, emergency rescue after an earthquake or flood, and temporary offices, conference or meeting in such company ("close or limited environment"). Figure 1.2 illustrates an example of ad-hoc network with three nodes.

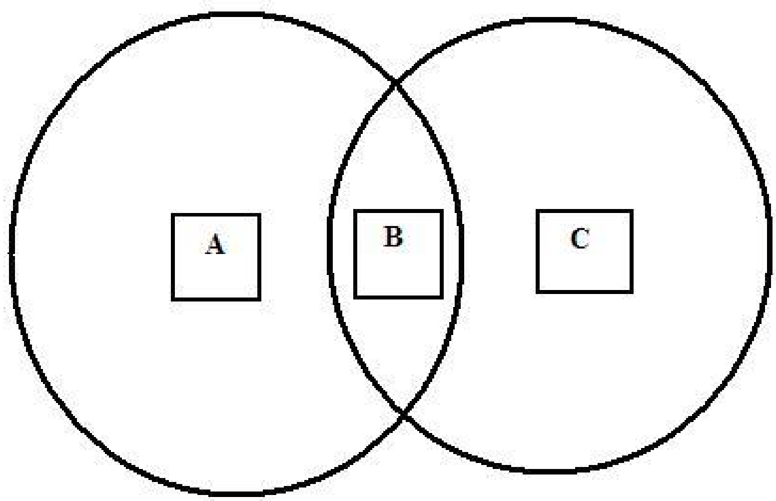

Figure 1.2: Example of a mobile ad hoc network with three participating nodes.

1.2. Problem background

In any network communication there is a need for a suitable routing mechanism to deliver packets from source to destination nodes. The concept of routing can be described as the process of path finding. In case of mobile ad hoc networks (MANETs) the main problem in routing mechanism, is how to send a packet from one node to another when there is no direct link exist between source and destination nodes.

Routing in ad hoc networks has always been a challenging task. The major reason for this is because of the characteristics of MANET itself that consists of high degree of node mobility, limited bandwidth, variable capacity links, energy-constrained operation, limited physical security, and no stationary infrastructure or base station for communication. Such characteristics make routing mechanism a challenging issue in MANET environment.

There are several techniques that have been proposed so far and all of them have their own advantages and limitations [9][17][19]. Each technique tries to determine the optimal path for packets delivery from source to destination nodes. The proposed routing mechanisms also should enable nodes to send and receive packets in reliable and fast manner. Some of the proposed routing mechanisms are suitable to be used in various MANET environments (i.e. open, localized, and organized environments), while some others are only suitable to be deployed in a specific environment.

1.3. Problem Statement:

Despite of long history of mobile ad-hoc networking (MANET's), there are still quite a number of problems that are open for discussion. Routing is one of the most important open issues in MANET research. The problem of routing can be divided into two areas: route discovery and route maintenance. When nodes want to communicate with each other, they must initially discover a suitable route to be used for the communication.

The high mobility, low bandwidth, and limited computing capability characteristics of mobile hosts make the design of routing protocols very challenging. The protocols must be able to keep up with unpredictable changes in network topology with as minimal message exchanges as possible and in the most efficient way.

This project proposes an alternative solution for ad-hoc network routing mechanism which aims to improve the reliability of the path chosen between source and destination nodes.

1.4. Project Objectives

This project aims to propose an alternative routing mechanism that could improve the reliability of packets sent between source and destination nodes in MANET environment. In order to achieve the aforementioned aim, the following objectives need to be accomplished.

1. To investigate existing routing mechanisms in MANET environment.
2. To propose an alternative routing mechanism based upon existing schemes
3. To analyze the performance of the proposed mechanism compared to the existing schemes.

1.5. Project scopes

The scope of this project is as follow.

1. The investigation of existing MANET's routing mechanisms is focusing upon open environment.
2. The proposed routing mechanism will focus upon the route discovery performance not on the security issues.
3. The analyses of the proposed routing mechanism will be conducted in simulation experiments.

1.6. Project organization

The organization of this report is as follow

Chapter 1 gives an introduction to the mobile ad-hoc wireless networks, the problem background, problem statement, project objectives, and project scopes.

In Chapter 2, sufficient reviews of previous works that have been done by researches in this area are presented, which covers the classification of mobile ad-hoc networks (MANET) routing protocols, their advantages and also the limitations.

Chapter 3 will be about the methodology of this study. Chapter 4, present the finding which be done in simulation experiments, Chapter 5, will be the summary of this project.

CHAPTER 2

LITERATURE REVIEW

2.1. Introduction

This chapter discusses some of the literature reviews that related to MANET routing mechanism based on previous work. The discussion starts with an introduction to routing mechanism, the categorization of existing ad-hoc routing protocols, and some examples of protocols that belong to each routing category.

The mechanism for each protocol, were discussed a bit detail, and how the protocol choose a route from source node to destination, the performance, advantages, and limitations for each protocol, which were tested and applied by the researchers.

Then, conclusion of what have been reviewed were mentioned, and the needs for a new protocol to reduce all the limitations exist in the existing ad-hoc routing protocols.

Finally, a summary of the chapter was written, with the main points highlighted.

2.2. Routing

Routing is the essential task of transferring packets form source node to destination, or can be described as the process of path finding [2]. A lot of routing algorithms have been proposed for wired networks and some of them have been widely used [1].

The routing concept has two activities, firstly, determining optimal routing paths and secondly, transferring the information groups (called packets) through the network [4].

Routing is classified into static routing and dynamic routing. Static routing refers to the routing strategy being stated manually or statically, in the router. Static routing maintains a routing table usually written by a networks administrator. The routing table does not depend on the state of the network status, i.e., whether the destination is active or not [3].

Dynamic routing refers to the routing strategy that is being learnt by routing protocol. This routing depends on the state of the network i.e., the routing table is affected by the activeness of the destination [4].

The large variety of routing protocols reflects the fact that these protocols do implement strategies very differently. The following section focuses on some types of ad hoc routing protocols, and their classifications.

2.3. Classification of routing Protocols in MANET

Routing mechanism in mobile ad-hoc network (MANET) a challenging task. Existing protocols try to overcome upon this challenging task, but most of the existing studies of these protocols are simulation based, with few real implementations. In early of 1970's many protocols have been developed for mobile ad-hoc network. Some of these protocols are suitable for limited network's nodes, and some of them deal with low bandwidth or high power consumption or high error rate. The mobile ad-hoc routing protocols can be categorized as (a) Table-driven , (b) on-demand driven and (c) Hybrid protocols [5] [6]. Figure 2.1. illustrates the classification of ad-hoc routing protocols.

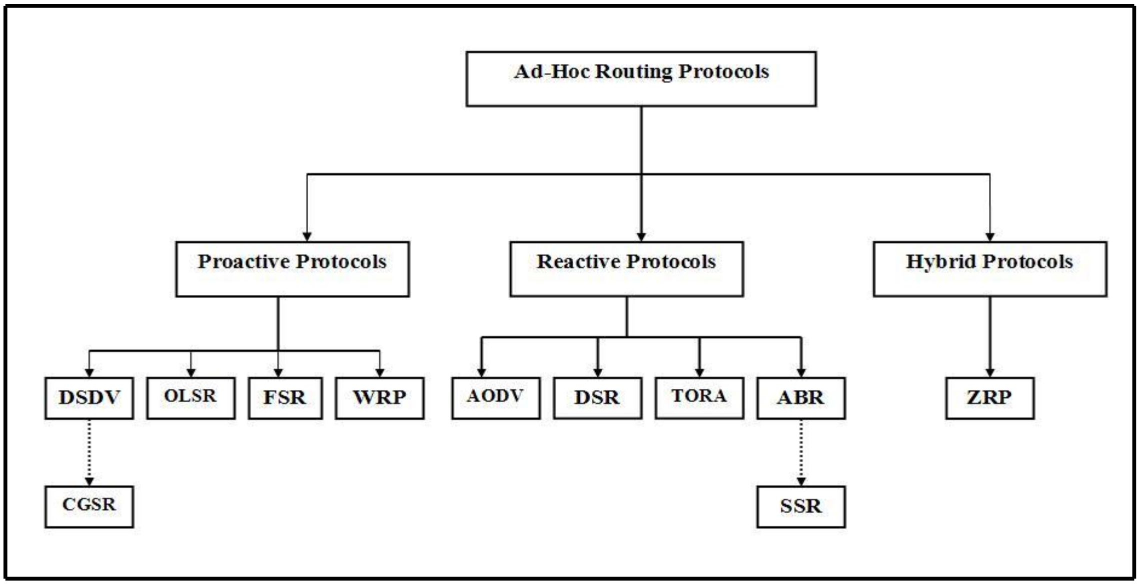

Figure 2.1: Classification of ad hoc routing protocols.

Solid lines in this figure represent direct descendants, while dotted lines depict logical descendants. Despite being designed for the same type of underlying network, the characteristics of each of these protocols are quite different.

Each approach has many protocols and each protocol has different mechanism to deliver the packet to destination node, and has it own advantages and limitations. Different protocols are specialized in different aspects of the routing. Some of the aspects other than finding a short path are low overhead communication and balancing [7].

The following sections described the protocols and categorize them according to their characteristics.

2.3.1. Table Driven Routing Protocols (Proactive)

In proactive routing protocols, each node maintains routing information to every other node (or nodes located in a specific part) in the network [9]. The routing information is usually kept in a number of different tables. These tables are periodically updated if the network topology changes, each node update its own routing tables by exchanging the recent update information from the other nodes in the same network. The exchanging information is happened by broadcasting the messages between nodes [5]. The differences between these existing routing protocols are the way of updating routing information, and how the information is kept at the routing tables [9].

These protocols are also called as proactive protocols since they maintain the routing information even before it is needed [8].

The proactive protocols are not suitable for larger networks, because this causes more overhead in the routing table leading to consumption of more bandwidth and memory [4].

The following section will describe the table-driven (proactive) protocols, the advantages and limitations for each protocol, and will discuss the mechanism for the each proactive protocol.

2.3.1.1. Destination-Sequence Distance-Vector Routing (DSDV)

The destination sequenced distance vector routing protocol (DSDV) is a table-driven routing protocol, based on the classical Bellman-Ford routing mechanism [10]. In the Bellman-Ford algorithm, every node sends its table to its neighbors at time intervals. When a node receives a table from a neighboring node it updates its own table [7].

The routing table contains all the information about routes to other nodes in same network and number of hop to these nodes. So each entry to this routing table labeled with sequence number by destination node. The mobile node can recognize the old route from new one through this sequence number. This property can help to avoiding the routing loop problem. The routing table is periodically updated through the neighbors nodes. The packet used by nodes to update the routing table can be classified into two possible types of packets, the first kind called a "full dump", this type of packets contain all available routing information about others nodes, the second type is smaller than first one, is called "incremental". This kind of packets carried only the information about topology changes since last full dump, and this packet decreasing the traffic on network [11] [9].

The information a bout new route could be broadcasted or multicasted [12]. The packet of this information contain the address of the destination, the number of hops to reach the destination, the sequence number of the information received regarding the destination, as well as a new sequence number unique to the broadcast [10].

The DSDV algorithm has many advantages such as:

- Guarantees loop free paths [13].
- Count to infinity problem is reduced [13].

- One can avoid extra traffic with incremental updates instead of full dump updates.
- Path Selection: DSDV maintains only the best path instead of maintaining multiple paths to every destination. With this, the amount of space in routing table is reduced.
- It provides a single path to a destination that could lead to overhead in the network [7].

However, the DSDV algorithm also has some limitation as follow:

- Wastage of bandwidth due to unnecessary advertising of routing information even if there is no change in the network topology [15].
- DSDV does not support multi path routing.
- It is difficult to determine a time delay for the advertisement of routes [16].
- It is difficult to maintain the routing table's advertisement for larger network. Each and every host in the network should maintain a routing table for advertising. However, for a larger network this would lead to overhead, which consumes more bandwidth and memory.

2.3.1.2. Cluster head Gateway Switch Routing (CGSR)

Cluster head Gateway Switch Routing protocol (CGSR) is different from DSDV in type of network organization. CGSR is a clustered mobile multi hop network, that is mean the network topology is divided to groups, each group of nodes have a cluster head node and getaway node to communicate with another groups (hierarchical). The cluster head control of it is own group of ad-hoc nodes. The election process of cluster head is done by using distributed algorithm with nodes (cluster) [17] [5].

A Least Cluster Change (LCC) algorithm was used to solve cluster-head problems. This algorithm used when two cluster head become in the same group or when node move out of the network topology, (any cluster head can not control this node) [7] [1].

Every node has two routing tables. The first table contains the information of the nodes in the network belongs to cluster head (member table). The second table is a distance-vector-routing table with the information of the next hop node for all cluster heads [5].

CGSR is improvement of DSDV using cluster head to getaway routing approach to route data from source to destination [5].

Getaway node is a node that is communicates two cluster head or more**.** The mechanism of routing is that a packet sent by the node first to cluster head and then the packet is routed again from cluster head to getaway node and to another cluster head, until the packet reached to the cluster head of destination node, then the packet transmit to destination. Figure2.2 illustrates an example of this routing scheme [17].

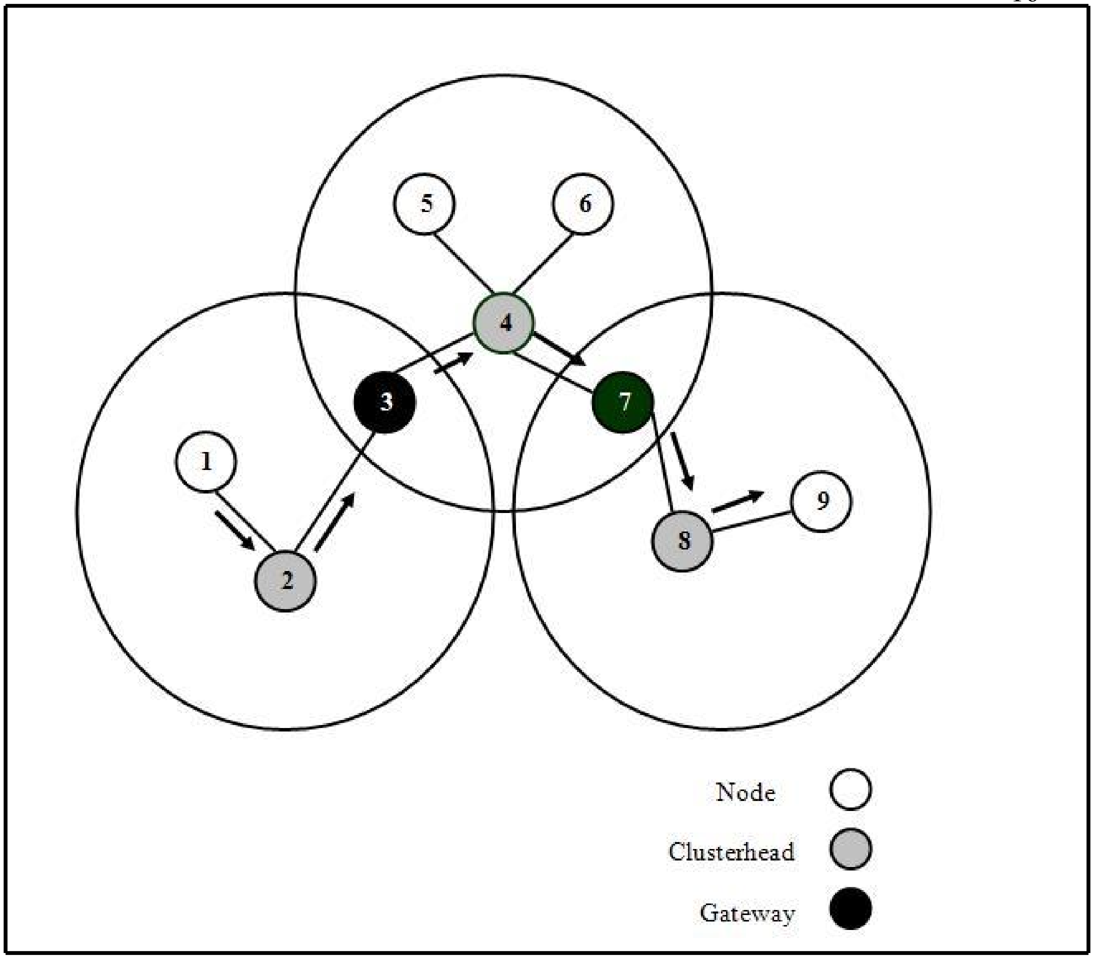

Figure 2.2: CGSR: routing from node 1 to node 9.

When a message is to be delivered from node 1 to node 9 in the example above, routing is done by the following steps. First the cluster head of node 9 is looked up in the cluster member table to find node 9. Second the next hop node for cluster head-node 9 is looked up in the distance-vector table to find node 2. The message is then sent to node 2 which passes it on to its gateway-node leading to cluster head- node 7. It is them passed on between gateways and cluster heads until it reaches node 8. Node 8 then passes is on to the destination node 9 [7].

The advantages of CGSR is

- Can use several methods to improve the protocol performance.

- Reduce the size of distance vector table because the routing is performed only over cluster head

The limitation of CGSR is

- The cluster member table required periodic update.
- Selecting cluster head take time and cause delay in routing mechanism .
- During link failure the time complexity is higher than DSDV.

2.3.1.3. The Wireless Routing Protocol (WRP)

The Wireless Routing Protocol (WRP) is a proactive routing protocol; it is broadcast routing information to all nodes in the network. WRP was proposing in 1996 [20]. It is related to Bellman-Ford routing algorithm [21].

Each node in the network has four tables, distance table, routing table, link-cost table, and message retransmission list (MRL) table [19].

Each entry of the MRL contains the sequence number of the update message, The MRL records which updates in an update message need to be retransmitted and which neighbors should acknowledge the retransmission. The benefit of update message is to inform the nodes in the same network about the changing in network topology. the update message is sent only between the neighbors and it contain a list of updates (The destination , the distance to destination , and the specific hops to destination).each node broadcast its current best route to specified destinations , if some information is missing the broadcast will be repeated with(MRL) [18] .

In case of link failure or disconnect the mobile node send a update message to their neighbors , the neighbors update their distance table entries and check the new possible route through another node to destination . if node don't send a message it must send a "hello message" within specified time period (such as 1 second) to say to other nodes it is still in connection with them [12] .otherwise that mentioned to link failure and cause false alarm. When other nodes receive a "hello message" form new node that new node added mobile's routing table and the node send back to new node a copy of its routing table information.

WRP belong to path finding algorithms [18], it avoid "count to infinity" and free from loops, by keep all the nodes to check the old information that broadcasting by all neighbors and make comparison with new information ,this process eliminate looping situations and provide faster link reconstruction when link failure .

WRP has advantages and limitations such as other protocols. Can be summarizing the advantages as follow:

WRP avoid creating temporary routing loop problem, during link failure the time complexity is lower than DSDV, because informs another nodes only about link status changing.

Limitations in WRP are:

WRP Using four routing tables hence its need memory requirement when number of nodes is large, more than other protocols. WRP use "Hello" message to keep route active and that consume bandwidth. Disallow to enter the node to sleep mode, and WRP has a limited scalability and is not suitable for large mobile ad hoc networks [19].

2.3.1.4. Fisheye State Routing (FSR)

The Fisheye State Routing (FSR) is a table-driven (Proactive) routing protocol based on Link State routing algorithm with effectively reduced overhead to maintain network topology information [19].

The concept of (FSR) ad-hoc routing protocol is utilizes a function similar to a fish eye. The eyes of fishes catch the pixels near the focal with high detail, and the detail decreases as the distance from the focal point increases [22].

FSR scheme is built on top of another routing scheme called “global state routing” (GSR) [24].

FSR introduce scopes. The scopes are the number of hops that a packet has reached from its source. Nodes within the smallest scope are considered most often in update packets; the nodes, are far away from focal, are considered less frequent. This means, can be reduced the message size, as information for most nodes can be discard. Although routes may become inaccurate for distant destinations under increased mobility of destination node, packets will find more and more accurate routes while getting closer to the target, thus they don't suffer much from the inaccuracy [23][12].

Figure 2.3 illustrates the function of fisheye in a mobile ad-hoc wireless network. The circles with different shades of grey define the fisheye scopes with consider the center node. The scope is defined as the set of nodes that can be reached within a given number of hops. In this case, three scopes are shown for one, two, and three hops, respectively. Nodes are color coded as black, grey and white define the scopes [23].

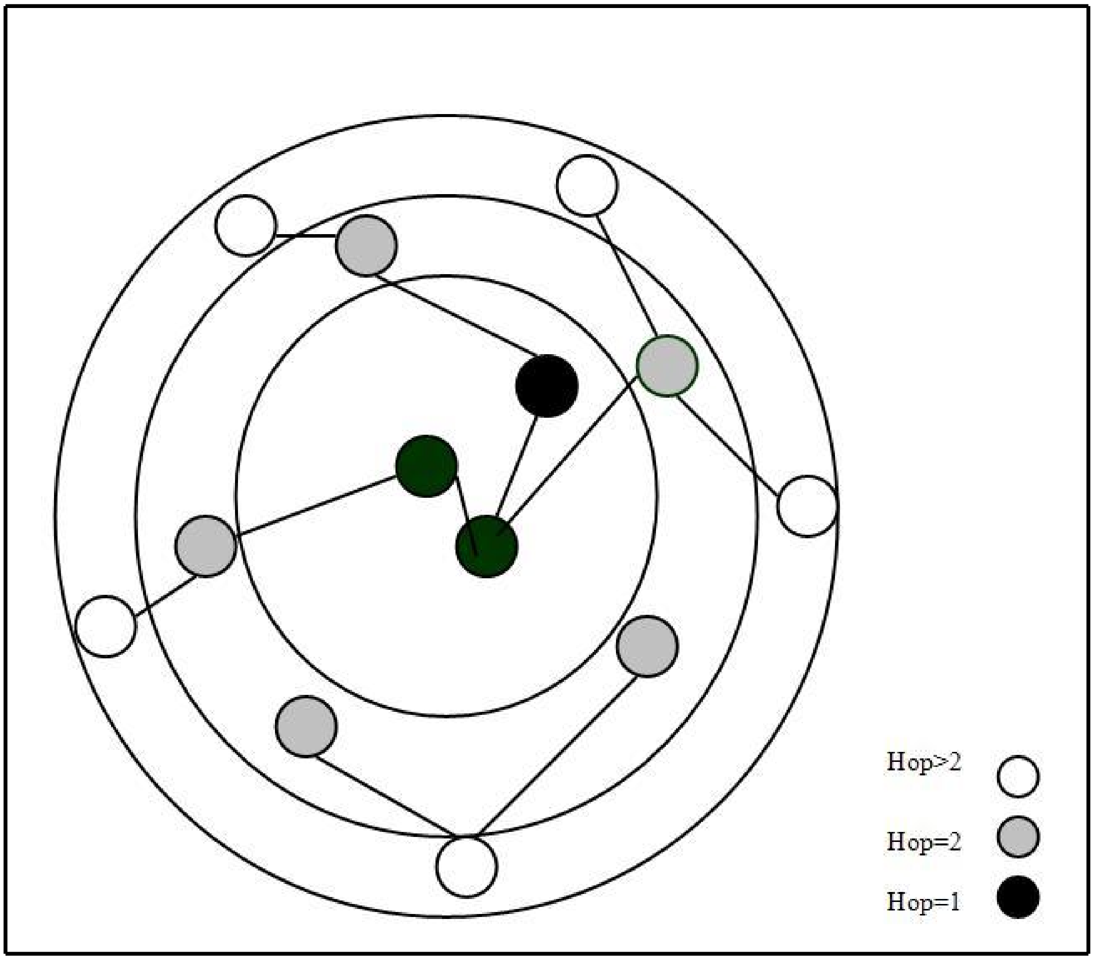

Figure 2.3: Scopes of Fisheye State Routing (FSR)

In summary, FSR support a good scalability to big size network compared to other link state protocols, because it does keeping all nodes in the network on the same knowledge level about link states [25]. FSR reduces the traffic overhead caused by exchanging link state information because this information is exchanged more frequently with node nearby than with nodes far away from the center [19].

2.3.1.5. Optimized Link State Routing (OLSR)

Optimized link state routing (OLSR) is a proactive routing protocol and it is link-state protocol [26]. OLSR protocol is an optimization for MANET of link-state protocols. OLSR is using two different message types, the Hello message, and the

Topology Control-message. The concept used in this protocol is that of Multipoint Relays (MPRs) [2].

The concept of multipoint relays (MPRs) is to minimize the overhead of flooding messages in the network by reducing duplicate retransmissions in the same region, where every node retransmits each message when it receives the first copy of the packet [27]. Multipoint relays (MPRs) are selected nodes which forward broadcast messages during the flooding process. A node's MPR set is a subset of its neighbors whose combined radio range covers all nodes two hops away. Each node obtains the two-hop topology through its neighbors' periodic broadcasting of HELLO packets containing the neighbors' lists of neighbors. Each node has to select MPRs among its Neighbors. The selection of MPRs has two rules:

_ Any neighbor two hops away has to be covered by a MPR [29].
_ the number of MPRs should be minimized [29].

In link-state protocol, a node's link information updating periodical throughout the network. However, in OLSR, when a node forwards a link updating packet, only those neighbors in the node's MPR forwarding the packet. Furthermore, a node only start link updates concerning those links between itself and the nodes in its MPR set. Therefore, routes are computed using a node's view of the network topology [28].

2.3.1.6. Summary of Table-Driven Ad-Hoc Routing Protocols (Proactive)

Table-driven routing protocols try to keep an up-to-date routing table in memory all the time. The updating of routing information should be accruing even the nodes dose not need a route to other nodes. .

In table 2.1 shows the main differences in mobile ad-hoc routing protocols (table-driven) are listed. Each table driven-routing protocol has different properties from others. Some of protocols have same properties.

Table 2.1: Differences of mobile ad-hoc routing protocols (Proactive)

PROTOCOLS	NUMBER OF TABLES	CRITICAL NODES	ROUTE SELECTION	ROUTES	CHARACTERISTIC FEATURE	BROADCAST	UPDATE DESTINATION
DSDV	**2**	**NO**	**Link State**	**Single**	**Loop free**	**Full**	**Neighbors**
CGSR	**2**	**Yes, Clusterhead**	**Shortest Path**	**Single or Multiple**	**Clusterheads exchange routing information**	**Full**	**Neighbors and cluster head**
WRP	**4**	**NO**	**Shortest Path**	**Single**	**Loop freedom using predecessor info**	**Local**	**Neighbors**
FSR	**3 and a list**	**NO**	**Shortest Path**	**Single or Multiple**	**Controlled frequency of updates**	**Limited**	**Neighbors**
OLSR	**3(Routing, neighbour And topology table)**	**NO**	**Shortest Path**	**Single or Multiple**	**Reduces Control overhead using MPR**	**Local to MPR nodes**	**MPRs nodes**

2.3.2. Source Initiated On Demand Routing Protocols (Reactive)

Table-driven (Proactive) different approach from routing is source-initiated on-demand routing (Reactive). This type of routing creates routs only when needed by source node to send a packet of information to destination node. When a node requires a route to destination, it is initiates a route discovery process within the network, when a route found or many possible routes discovered then the node choose a specific route with smaller metric (shortest path).

In this approach of routing (Reactive) try to eliminate the routing tables and reduce the need of updating these tables in contrast with proactive (table driven) routing protocols which maintain all tables up-to-date at every node .in reactive routing protocols (on demand) routes is always available with reduce of network traffic and power consumption but on demand routing suffer longer delay while route discovery.

In the following sections described some of reactive routing protocols such as AODV, DSR, TORA, ABR, and SSR.

2.3.2.1. Ad hoc On Demand Distance Vector Routing (AODV)

Ad hoc On Demand Distance Vector Routing (AODV) routing protocol build on DSDV described previously (see 2.3.1.1), AODV is a improvement of DSDV because it is minimize the number of required broadcast by creating routes on demand , in contrast with DSDV algorithm, its creates list of routes [17].

AODV classified as a pure on demand routing protocol system, when a node want to send a message to another destination node and dose not have a valid route to that destination its initiates a path discovery process to find the destination node. The source node broadcast a route request (RREQ) packet to its neighbors, and these neighbors forward the request to their neighbors , and so on until reach to destination node or reach intermediate node have a information a bout the route to destination node [28] [30].

Each node recode its own sequence number known as a broadcast ID, The broadcast ID is incremented for every RREQ the node initiates, and record also the nodes IP address , a RREQ with its own sequence number and the broadcast ID, the source node includes in the RREQ the most recent sequence number it has for the destination, Intermediate nodes can reply to the RREQ only if they have a route to the destination if the destination sequence number is greater than or equal to that contained in the RREQ [30].

During forwarding process the intermediate node records the address of the neighbors, from which node first copy of RREQ is broadcasted is received, if additional copy is received from same RREQ, these packets are discarded to avoid looping problem. When the RREQ reached the destination node or intermediate node with recent route to destination, the destination or intermediate node responds by

unicasting a route replay (RREP) packet back to the neighbors that received the first RREQ [7] .

When a intermediate node discover a broken link or failure in active route, it broadcast a route error (RERR) packet to inform its neighbors it discover a route failure, and then these neighbors forward this RERR packet to all nodes that use this broken route, then the source node can re-initiate route discovery process if the route is still needed [7][30].

The advantage of AODV is avoiding creating temporary routing loop problem. During link failure the time complexity is lower than DSDV, because informs another nodes only about link status changing. Routes are established on demand and destination sequence numbers are used to find the latest route to destination. The connection setup delay is less.

And the limitations of AODV need big memory capacity. Route Request packet can lead to heavy control overhead.

2.3.2.2. Dynamic Source Routing (DSR)

The Dynamic Source Routing (DSR) protocol is an on-demand (Reactive) routing protocol that is based on the concept of source routing [18]. DSR consist of two major phases, route discovers and route maintain.

In DSR each node uses caching technology to keep route information that it has obtained it.

When a node want to send a packet of information to other node, the first thing is check route cash to determine if has a route to destination or not, if has unexpired route to destination (valid route) the source node will use this route otherwise, the source node will initiate a discovery process by broadcast a route request packet. This packet contains the address of destination and address of source node with unique identification number [31].

Each node receive the route request packet, a node check its route cash. If the node doesn’t have routing information for the requested destination, the node adds its own address to route record of the packet and forward the packet to its neighbors and so on until the packet reach the destination or a node know the route to destination node[32] [19][18].

When a packet reached the destination or a intermediate node know the route to destination, the intermediate node or the destination generate the route reply, it places the route record contained in the route request into the route reply, If the responding node is an intermediate node, it will append its own address to the route record and then generate the route reply. To return the route reply, the responding node must have a route to initiate node (source node), If it has a route to the source node in its route cache, it will use that route, otherwise, the node will reverse the route in the route record [18].

Route maintenance is done through the use of route error packet, when a node discover a route transmission problem, it send a route error packet to the source node, which may reinitiate route discovery if alternate route not available [32] [31].

DSR has many of advantages such as, DSR does not use periodic routing advertisement hence saving in bandwidth and reduce power consumption. DSR allows nodes keep multiple routes to destination in their cash. Intermediate nodes use the route cache information to reduce the control overhead. Route recovery is faster than in many of the other on demand protocols (Reactive).

2.3.2.3. Temporally Ordered Routing Algorithm (TORA)

Temporally Ordered Routing Algorithm (TORA) is a distributed routing algorithm based on the concept of link reversal, and TORA is highly adaptive and loop free protocol.

TORA is suitable for high dynamic mobile networking environment, the mechanism depend on source initiate and provide multiple route between source and destination node, The concept of TORA is localization of control message to a very small group of nodes, this protocol perform three basic function, route creation, route maintaining, and route deletion [7].

In TORA, the network topology is considered as a directed graph. During the route creation and maintenance phases, nodes use a height metric to create a Directed Acyclic Graph (DAG) rooted at the destination, after that links are assigned a direction upstream or downstream, depend on the relative height metric of neighboring nodes, This process of establishing a DAG is similar to the query reply process [9]. Times of node mobility the DAG route is broken and route maintenance is necessary to re-establish a DAG rooted at the same destination [18].

Node generates a new reference level which results in the propagation of that reference level by elective neighboring nodes coordinating a structured reaction to the failure Links are reversed to the change in adapting to the new reference level; this has the same effect as reversing the direction of one or more links when a node has no downstream links.

When multiple nodes wants erasing routes and building new routes based on each other, its instability problem is similar to the count to infinity problem in distance vector routing protocols and routes cross will occur [9].

The advantages of TORA is, good to use with large network and high dynamic topology. Support multiple routes. The route reconstruction in TORA is not necessary until make sure that the route between source and destination is invalid, hence use low bandwidth. Support multicast [7]. TORA is that it has reduced the control messages to a set of neighboring nodes, where the topology change has occurred [9]. TORA has limitation also such as, route re-building is not quickly process as another protocols.

2.3.2.4 Associativity Based Routing (ABR)

Associativity Based Routing (ABR) **is** totally different form previous protocols, in this protocol define a new kind of metric in mobile ad hoc networks. This metric is known as a degree of association stability [33].

In ABR the route chosen based on degree of association stability of mobile node, each node creates a beacon, the periodic beacons are exchanged between neighboring nodes. Every node keeps an associativity table; the node records the connection stability between itself and its neighbors over time and space. When receiving a beacon, nodes increase the associativity tick with respect to the sender. Therefore, the link with higher associativity tick is more stable than the one with lower associativity tick. When a neighboring node moves out of network, the associativity tick for it will be reset [19].

There are three phases in ABR, and these phases are, route discovery, route reconstruction (RRC) and route deletion.

When some node want a routing path to another node, the source initiate the route discovery operation by Broadcast Query (BQ).When intermediate node receiving a BQ, the node check if the same message has been receive before. If yes, it discards the BQ. Otherwise, it appends its address and its association ticks inside the BQ packet and broadcasts the BQ again [34]. When a subsequent node receives the BQ from its upstream node, it erases the associativity tick entries of its upstream node, so, when a BQ arrives at the destination, it contains the addresses of the intermediate nodes and the associativity ticks about all the links along the route. The destination selects the best route according to the old one of all possible routing paths.

The destination node will choose the shortest hop number (shortest path) if there are multiple paths with the same degree of association stability. Then, the destination sends a reply packet that contains routing information of the selected path back to the source.

When an intermediate node forwards the reply packet, it marks its route as valid, therefore, no duplicated data packet will be sent to the destination [33][19].

If the movement of the source causes a RRC (route reconstruction), the source node will initiated a new route discovery operation and the source sends a route notification (RN) message to erase the route entries a bout the out-of-date route [19].

When the destination moves, its immediate upstream node erases the routing entry associating with the destination. Then, a localized query (LQ) is sent by the immediate upstream node to test if the destination is still reachable. The hop number from the node to the destination is included in the LQ. If the destination receives the LQ, it selects the best route and replies. Otherwise, the next upstream node will repeat the same operation, and an RN message is sent to the next upstream node to erase the invalid route and a LQ is sent to the destination. If the operation has backtracked more than halfway to the source and the destination still cannot be reachable, the source initiates a new BQ process instead of the localized query [34][19].

In ABR, a source node initiates the Route Delete (RD) operation by broadcasting a RD message when a route is not needed. After receiving the RD message, all nodes along the route delete the route entries [19].

The advantages of ABR is, the route be a longer-lived, higher throughput, because route process required fewer route reconstruction, Free from packet-duplicate, free from loop and deadlock [18].

The limitation of ABR is, ABR use beaconing, hence need more power consumption. ABR does not use route cash. The choosing path is not depended on shortest path but depend on stable one and longer-live. Longer-live route reconstruction cause delay because failure link can not solve locally but should intervention the source node [18][33][34].

2.3.2.5. Signal Stability Routing (SSR):

Signal Stability protocol is unlike others protocols described previously, SSR select the route based on signal strength between nodes and on node's location stability, in SSR the route select base on route that have stronger connection with other nodes, and SSR combine two features, dynamic routing protocol (DRP) and, static routing protocol (SRP) [35].

The DRP is responsible for the maintenance of the Signal Stability Table (SST), and the Routing Table (RT). In SST record the signal strength of neighbors, after update all entries of SST, DRP pass the packet to SRP. The SPR receive the packet and look for the destination in routing table (RT), if no entry is found in the routing table, the source node initiate route discover process to find a route by forward route request to neighbors but only forward for one hop, if the neighbors receive the packet over strong channel and have not been previous received (to avoid looping), the destination choose the first arrived route-discover packet because the packet is arrived over shortest path, then the DRP send back a route reply message to source node. The DRP update the information about the route in RT. The packet should arrive to the destination over stronger path and stability signal, otherwise the destination will dropped this packet [36].

When the node detect failed link in the network, the intermediate nodes send an error message to the source indicating which channel has failed, The source then initiates another route discovery process to find a new path to the destination, The source also sends an erase message to inform all nodes about the broken link [18].

The advantages of SSR is using stability of links instead of hop numbers as metric for routing path selection is a good solution for reducing control overhead. In SSR the messages are only sent through strong channels to reduce the traffic overhead [36].

2.3.2.6. Summary of On-Demand Ad-Hoc Routing Protocols (Reactive)

In general, on-demand reactive protocols are more efficient than proactive approach. Re-active protocols minimize control overhead and power consumption since routes are only established when required. By contrast, proactive protocols require periodic route updates to keep information updated and available; in addition, maintain multiple routes that might never be needed, adding unnecessary routing overheads.

In table 2.2 shows the main differences in mobile ad-hoc routing protocols (On-Demand) are listed. Each reactive-routing protocol has different properties from others. Some of protocols have same properties.

Table 2.2: Differences of mobile ad-hoc routing protocols (On-Demand)

PROTOCOLS	BEACONS	ROUTE METRIC METHOD	ROUTE MAINTAINED IN	ROUTES	ROUTE RECONFIGURATION STRATEGY	BROADCAST	UPDATE DESTINATION
AODV	Yes, hello messages	Freshest & SP	RT	Multiple	Erase route then SN or local route repair	Full	Source
DSR	No	SP, or next available in RC	RC	Multiple	Erase route the SN	Full	Source
TORA	No	SP, or next available	RT	Multiple	Link reversal & Route repair	Local	Neighbors
ABR	Yes	Strongest Associativity & SP &b	RT	Single	LBQ	Full	Neigh/Source
SSR	Yes	Strongest signal strength	RT	Multiple	Erase route then SN	Limited	Neighbors

RT=route table; RC=route cache; SP=shortest path; SN=source notification;
LBQ=localised broadcast query.
b Route relaying load and cumulative forwarding delay.

2.3.3. Hybrid routing protocols

The ad hoc network can use the hybrid routing protocols that have the advantage of both proactive and reactive routing protocols to balance the delay and control overhead (in terms of control packages). Hybrid routing protocols try to maximize the benefit of proactive routing and reactive routing by utilizing proactive routing in small networks (in order to reduce delay), and reactive routing in large-scale networks (in order to reduce control overhead) [37].

The difficulty of all hybrid routing protocols is how to organize the network according to network parameters. The common disadvantage of hybrid routing protocols is that the nodes that have high level topological information maintains more routing information, which leads to more memory and power consumption [38].

The following section describes Zone Routing Protocol (ZRP) as a hybrid routing protocol.

2.3.3.1 Zone Routing Protocol (ZRP)

Zone Routing Protocol (ZRP) provides a hybrid routing mechanism that locally proactive (table driven) and globally reactive (on demand). In ZRP, the network is divided into routing zones according to distances between mobile nodes, each node advertise its link state a fixed number of hops, called the zone radius. This advertises give each node updates of its routing zone, the routing zone nodes that are at a minimum distance of the zone radius are called peripheral nodes. The peripheral nodes represent the boundary of the routing zone and play an important role in zone based route discovery [26] [37].

ZRP uses the information of routing zone connectivity to guide its global route discovery. Rather than blindly broadcasting route queries from a node to all its neighbors, the source node send a route request directly from a itself to its peripheral nodes via multicast. This mechanism reduces the number of query message that broadcast in others protocol. When a destination receives a route query, then a destination send back a query reply to a peripheral nodes [26][19].

The main advantages in ZRP are helping to improve the quality and survivability of discovered routes. ZRP offer enhanced, real-time, route maintenance. ZRP avoid link failures [19].

And the limitations of ZRP are the performance of a zone based routing protocol is tightly related to the dynamics and size of the network. ZRP in general will incur more overhead because zones heavily overlap [38].

2.4 Summary

This chapter provides description of several routing protocols proposed for mobile ad hoc networks (MANET's), and the classification of these protocols according to the routing strategy, i.e., table-driven, on-demand and hybrid. And described a mechanism for each of these protocols, highlighting their features .finally identified the advantage and disadvantages for each of these protocols and which protocol is well-suited for which situation.

Table 2.3 summarize the main differences in mobile ad-hoc routing protocols mentioned previously.

Table 2.3: The main differences in mobile ad-hoc routing protocols

Protocols	Update Destination	Routing Philosophy	Structure	Hello message requirement	Route Selection	Routes	Broadcast	Availability of routing information
DSDV	Neighbors	Proactive	Flat	NO	Shortest path	Single	Full	Always available
CGSR	Neighbors and cluster head	Proactive	Hierarchical	NO	Shortest path	Single or Multiple	Full	Always available
WRP	Neighbors	Proactive	Flat	YES	Shortest path	Single	Local	Always available
FSR	Neighbors	Proactive	Flat	NO	Shortest path	Single or Multiple	Limited	Always available
OLSR	Neighbors **(MPR)**	Proactive	Flat	YES	Shortest path	Single or Multiple	Local	Always available
AODV	Source	Reactive	Flat	YES	Faster and Shortest path	Multiple	Full	Available when needed
DSR	Source	Reactive	Flat	NO	Shortest path	Multiple	Full	Available when needed
TORA	Neighbors	Reactive	Flat	NO	Shortest path	Multiple	Local	Available when needed
ABR	Source	Reactive	Flat	YES	Signal Strength	Single	Full	Available when needed
SSR	Neighbors	Reactive	Flat	YES	Signal stability	Single	Local	Available when needed
ZRP	Neighbors	Hybrid	Flat	YES	Shortest path	Single or Multiple	Local	Available when needed

The field of ad-hoc mobile networks is rapidly growing and changing, and while there are still many challenges that need to be met.

CHAPTER 3

RESEARCH METHODOLOGY

3.1. Introduction

This chapter describes the methodology of this study. The methodology is a sequence of methodical work done through steps, which is used as guideline throughout a research, in order to accomplish the objectives of the research.

This chapter also illustrating the framework that will be used in this project. The first section shows the project structure, then it is followed by project operational framework, and software requirements.

3.2. Project Structure

In this section, project structure is explained. It gives a clear vision for the work which has been done in this preliminary study. In addition, defining the problem background, determining the problem statement and scopes. This study start by, Data collecting and analysis is very helpful for the researcher to conduct his research. Then, a new switching mechanism for mobile ad-hoc network (MANET) is proposed. It is followed by simulation of the new switching mechanism using Network Simulator 2 (NS2) in order to evaluate the performance.

Finally, is compared the performance of new switching mechanism with previous protocols to see the differences between them. Figure 3.1 illustrates the project structure.

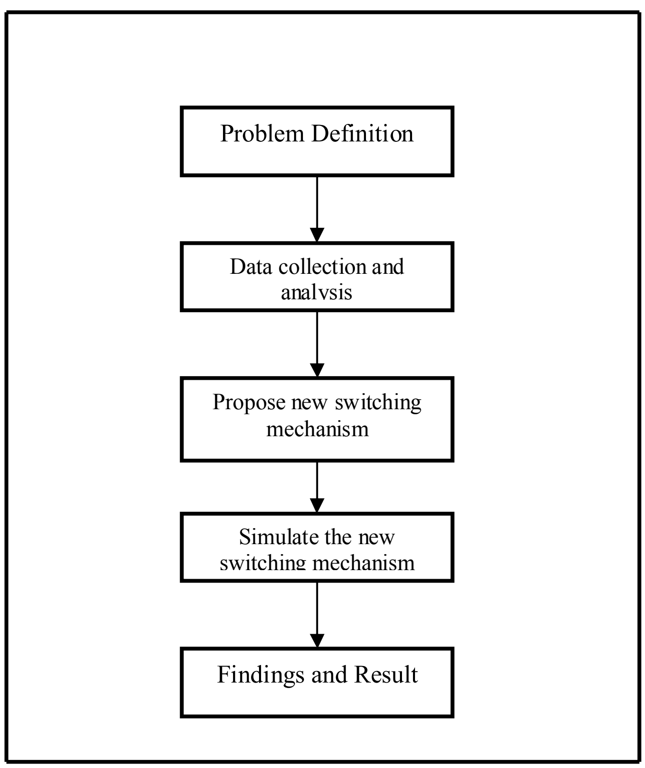

Figure 3.1: Project Structure Steps

3.3. Operational Framework

In this section, the main steps of this project will be explained to achieve the objectives of this project. The goal of this section is to help in developing the project schedule. Figure 3.2 shows the operational framework stages that are used in this project.

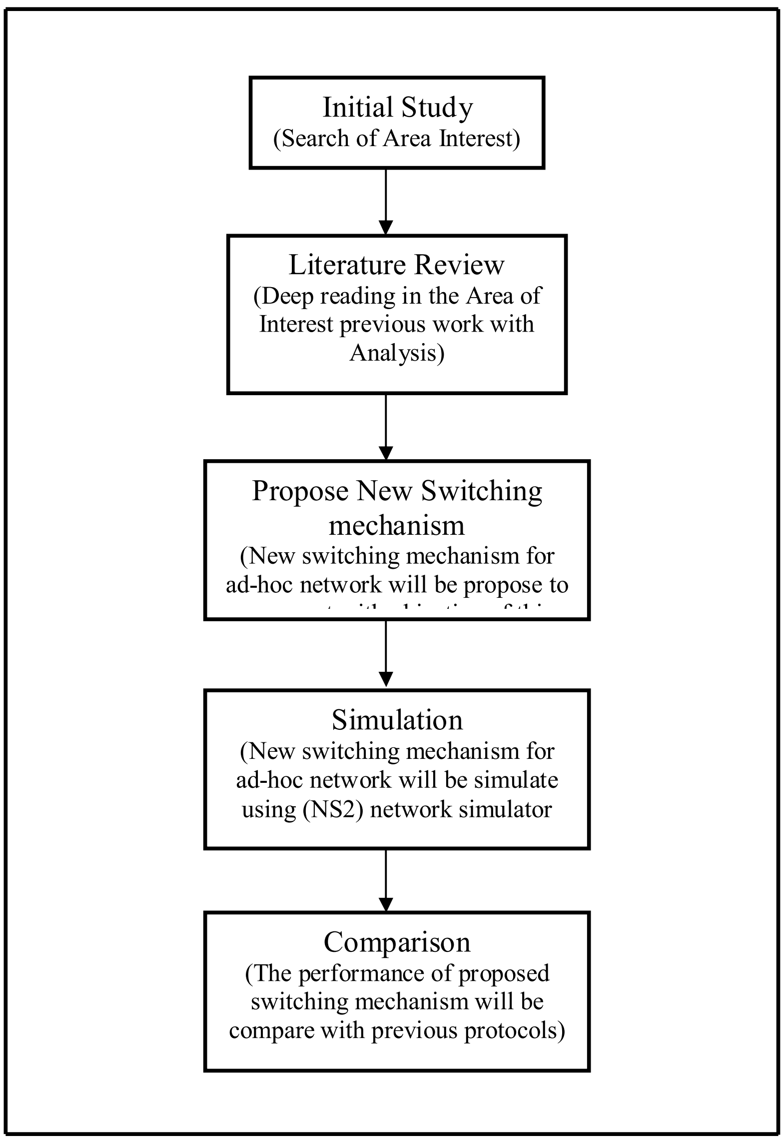

Figure 3.2: Project Operational Framework

3.3.1. Initial Study

Initial study covered mobile ad-hoc network (MANET) sequentially to define the problem statement through deferent altitudes, in order to build a good background in the matter of the research. As a result, the routing was the emerging idea through previous literature review.

3.3.2. Reviewed Related Literature

In this stage of project work, the main related work for routing in mobile ad-hoc network was reviewed from books, journals, and some Internet articles. Reviewing was focused on ad-hoc routing protocols, classification of routing protocols, explanations of mechanism for each protocol, and advantages and limitations for each protocol.

3.3.3. Propose New Switching Mechanism

Proposing a new switching mechanism for hybrid protocol based on previous protocols will be done by developing an operating mechanism, as well as considering the advantages and reduce the limitations. The new protocol is proposed in chapter 4.

3.3.4. Simulation

The new proposed switching mechanism for hybrid protocol will be simulated to present the performance of the overall system. The simulation will be done using, Network Simulator 2 (NS2) tools.

3.3.5. Comparison

Results of this project are measured by comparing the performance of new ad-hoc switching mechanism for hybrid protocol that proposed in this study, and previous routing protocols; to see the differences between these protocols.

The Figure 3.3 illustrates the research methodology steps for this study.

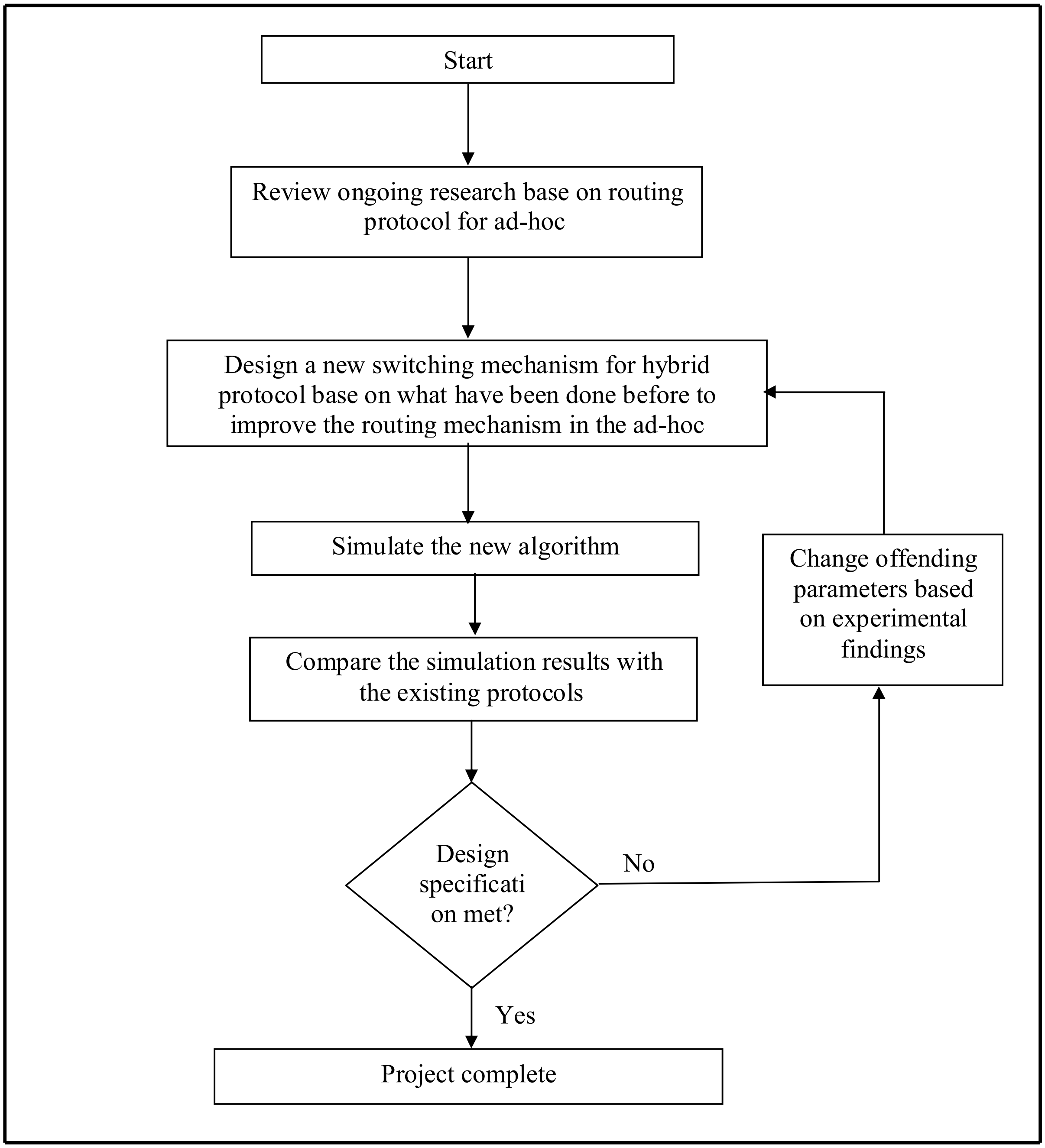

Figure 3.3: Research Methodology Steps

3.4. Software Requirements

In this section, software that will be used through this project are listed. This helps to make the project clear and easy.

Softwares that would be used in this project to propose and analyze the new routing protocol for ad-hoc networks.

- Microsoft Windows XP Professional.
- Linux fedora 8.
- Microsoft Office 2007 Professional.
- Network Simulator 2 (NS2) v 2.30.
- Trace graph.

3.5. Summary

This chapter has explained the methodology that will be followed to achieve project objective. In this chapter also the operational framework was discuss, to make the work process clear and guided. It is hope that this methodology will serve the objectives of this project in best way.

CHAPTER 4

RESULTS AND DISCUSSION

4.1. Introduction

In this chapter, results of this project were presented and discussed in details. Comparison between proposed protocol and existing protocols was discussed and highlighted.

The result explained in the following section have been collected at the end of each simulation, and then they were analyzed and documented.

4.2. Simulation setup

Simulation was done by using network simulator 2 (NS2), v 2.30. Table 4.1 illustrates the simulation setup.

Table 4.1: Simulation setup

Simulation parameter	Value
Channel type	Wireless Channel
MAC type	802.11
Link layer type	Traditional Link Layer (LL)
Antenna model	Omni-directional (unity gain)
No. of nodes	50
No. of sending nodes	1
No. of receive nodes	1
Simulation time	300 Sec
Size of simulation area	800m x 800m
Initial Power level	104000 watt
Power consumption /packet	228.0 joules

4.2.1. Network Simulator (NS-2):

NS (version 2) is an object-oriented, discrete event packet level network simulator. It is usually used by the academic community to evaluate research proposals or ideas in the area of networking. The goal of NS2 is to support networking research and education. It is suitable for designing new protocols, comparing different protocols and traffic evaluations. The NS2 is open-source and users are free to add or modify any sections as they consider necessary. NS2 is capable of supporting both wired and wireless (local and satellite) networks, as well as real support for simulation of TCP, routing, and multicast protocols.

NS2 use both OTcl script and C++ to implement a protocols and setup and direct simulations scenarios. C++ is often used when control is needed at packet level. In the same time OTcl is a scripting language, that allows for the connecting of various other languages (originally C, now also C++, Java, Eiffel, and Prolog). Scripts can be written to allow the bringing together of a number of different sections of code in these languages. OTcl is also programmable and allows for complex commands and structures to be created. Often the programs need to use the procedures, looping command, and conditional commands. OTcl was originally designed with the philosophy that a user would use two or more languages, and OTcl would be used as the intermediary to allow communications between the languages. The languages would make use of the standard tcl syntax, plus this would allow users to issue commands, in tcl, to a program using a language they may not necessarily understand. NS2 support just four ad hoc routing protocols, AODV, DSDV, TORA, and DSR at the moment. OTcl is used to build a simulations scenarios object and control the data flow. Figure 4.1 illustrate the relationship between C++ and OTcl script.

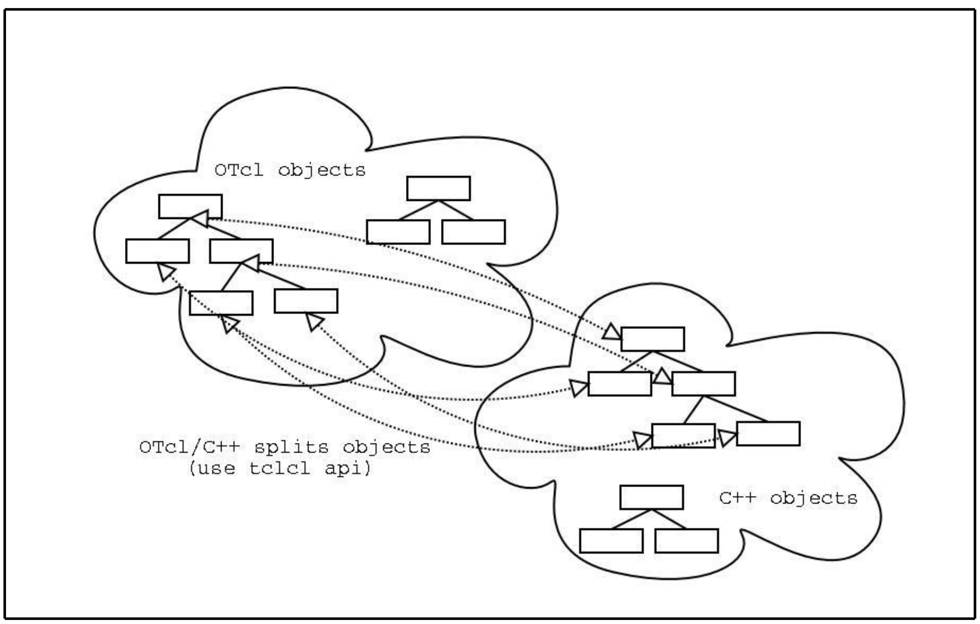

Figure 4.1: Relationship between C++ and OTcl script [33].

NAM (Network Animator) is a tool includes in NS2. It is an animator, designed specifically to visualize the protocols implemented in NS2. NAM is a Tcl/TK based animation tool for viewing network simulation traces and real world packet trace data. The first step to use NAM is to produce the trace file. The trace file should contain topology information, such as nodes, links, as well as packet traces. Usually, the trace file is generated by NS2. During an ns simulation, user can produce topology configurations, layout information, and packet traces using tracing events in NS2.

4.2.2. Protocols Implementation:

The main interest of this project is to propose a routing protocol that has ability
to react on network topology changes (for instance node movement, and node vicinity). Furthermore, this study performed three sets of simulations; in these three sets the three protocols (AODV, DSDV, and Hybrid) were implemented under same scenario. The following subsections will discuss this scenario.

4.2.2.1. AODV Simulation Scenario:

AODV implementation was done with 50 wireless nodes, where the nodes are placed randomly within an 800m x 800m terrain or simulation area. The IEEE 802.11 was used as the MAC protocol. In addition, the mobility model used was the random waypoint model, and the simulation time is set to 300s. The initial battery energy = 104000 wt and, power consumption 228 wt for each send/receive/forward packet from node to node. This setting is done to validate the effect of proposed protocol on throughput and delay ratio. In addition, TCP sending window limit is 20 packets (Max Size Segment). There is only one TCP flow in the simulations to reduce the impact of multiple flows. TCP flow starts from 10s, thus the first 10s is for the warm up. However, after 10 sec of simulation initialization node0 will start send data to node1 in mobile pattern (node0, and node1 start moving randomly), and the minimum speed of node is (0m/sec) and maximum speed of node is (10m/sec).

4.2.2.2. DSDV Simulation Scenario:

DSDV scenario implementation was done same as AODV scenario, with 50 wireless nodes, and nodes placed randomly within an 800m x 800m area. The IEEE 802.11 was used as the MAC protocol. In addition, the mobility model used was the random waypoint model, and the simulation time is set to 300s. The initial battery energy = 104000 wt and, power consumption 228 wt for each send/receive/forward packet from node to node. This setting is done to validate the effect of proposed protocol on throughput and delay ratio. In addition, TCP sending window limit is 20 packets (Max Size Segment). There is only one TCP flow in the simulations to reduce the impact of multiple flows. TCP flow starts from 10s, thus the first 10s is for the warm up. However, after 10 sec of simulation initialization node0 will start send data to node1 in mobile pattern (node0, and node1 start moving randomly), and the minimum speed of node is (0m/sec) and maximum speed of node is (10m/sec).

4.2.2.2.1. Hybrid Simulation Scenario:

Hybrid approach is implemented same as AODV and DSDV scenarios, with 50 wireless nodes, and nodes placed randomly within an 800m x 800m area. The IEEE 802.11 was used as the MAC protocol. In addition, the mobility model used was the random waypoint model, and the simulation time is set to 300s. The initial battery energy = 104000 wt and, power consumption 228 wt for each send/receive/forward packet from node to node. This setting is done to validate the effect of proposed protocol on throughput and delay ratio. In addition, TCP sending window limit is 20 packets (Max Size Segment). There is only one TCP flow in the simulations to reduce the impact of multiple flows. TCP flow starts from 10s, thus the first 10s is for the warm up. However, after 10 sec of simulation initialization node0 will start send data to node1 in mobile pattern (node0, and node1 start moving randomly), and the minimum speed of node is (0m/sec) and maximum speed of node is (10m/sec).

4.2.3. Metrics:

In order to evaluate the performance of the three protocols (AODV, DSDV, and Hybrid) this study considering the following metrics:

- Drop rate: The ratio of lost TCP DATA packets over sent TCP DATA packets.
- Lost rate: Packet loss is the failure of one or more transmitted packets to arrive at their destination.
- Packets generation.
- End –to-end delay: the average delay time of all successfully delivered packets from source node to destination.

A lot of studies have compared the performance of the routing protocols in MANET using the above metrics [32].

4.3. Results:

As already mentioned in section 4.2.2., this study implement the proposed switching mechanism for Hybrid protocol , AODV, and DSDV with fixed network size. In this section the results for the three protocols that obtained from simulation will be presented and conclusions of the results for each protocol were discussed at the following subsections.

4.3.1. AODV:

Most of protocols perform well in small networks (i.e. 10 nodes), when only few hops need to be taken to reach the destination node. However, based on simulation results, AODV has a number of dropped packets equal to 117 packets during simulation time that set to 300s. Figure 4.2 illustrate the ratio of dropped packets. That obviously clear the number of dropped packets in begin of simulation were huge because the rate of mobility is high while at the end of simulation the rate of dropped packets were low because the rate of mobility is too low.

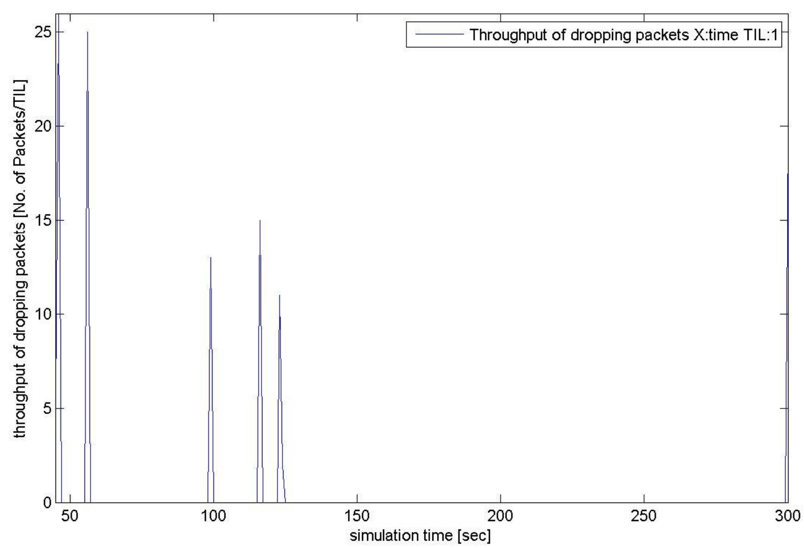

Figure 4.2: packets Dropped Ratio in AODV

In addition, the total packets generation by AODV was 26875 packets during simulation time. Figure 4.3 illustrate the rate of packets generation as the mobility speed is varied and the movement of source node and destination are randomly. That can be noticed the number of generate packets in begin of simulation were huge because the path between source node and destination is short (low number of hops), while at the end of simulation the rate of generate packets were low because the path between source node and destination is long (high number of hops).

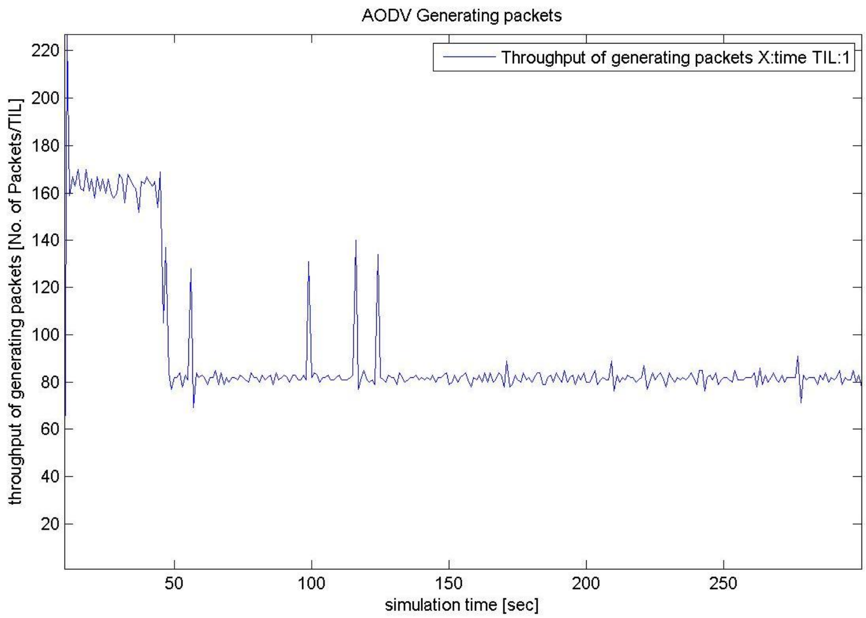

Figure 4.3: Packets generation ratio in AODV.

Next, the analysis of packet loss ratio in AODV is as illustrated in Figure 4.4. The number of lost packets in AODV during simulation is 30 packets. At low mobility (end of simulation), the packet loss is overall quite low. At high mobility (beginning of simulation), the loss rate is increased.

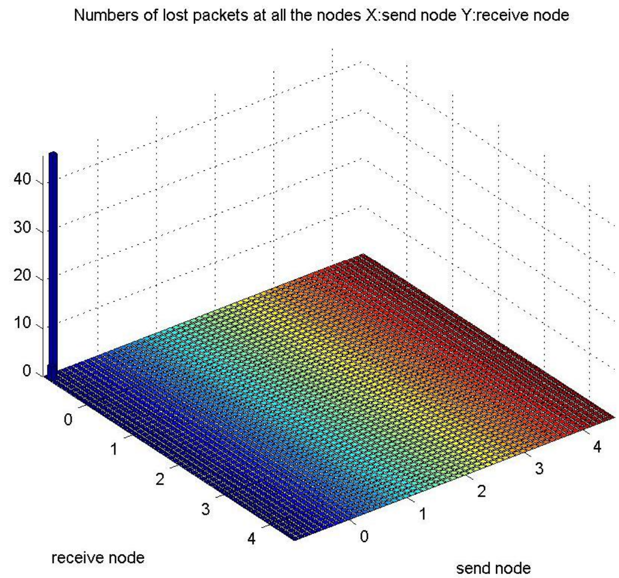

Figure 4.4: Packets loss ratio in AODV.

4.3.2. DSDV

In figure 4.5 the dropping packet during simulation has high average drop rate. This results obtained according to apply same scenario in AODV on DSDV. Therefore, the number of dropping packets during simulation is 149 packets. It is obviously the drop rate is different, depend on mobility pattern. However, in DSDV with low mobility, the dropping rate is low and DSDV with low mobility pattern can maintain routing information effectively. And with high mobility the dropping rate is high**.** That obviously the number of dropped packets in begin of simulation were huge because the rate of mobility is high while in the end of simulation the rate of dropped packets were low because the rate of mobility is too low.

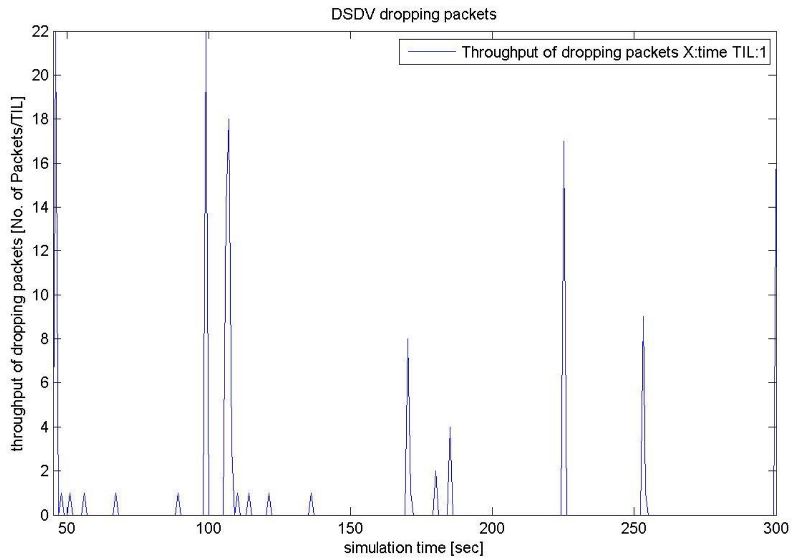

Figure 4.5: Packets drop ratio in DSDV.

DSDV suffer from low generating packets rate, for both stable link and failing link, even with low mobility nodes. However, the number of generating packets in DSDV during simulation time (300s) is 20598 packets. Figure 4.6 illustrate the generation packets ratio. That can easily noticed the number of generate packets in begin of simulation were huge because the path between source node and destination is short (low number of hops) while in the end of simulation the rate of generate packets were low because the path between source node and destination is long (high number of hops).

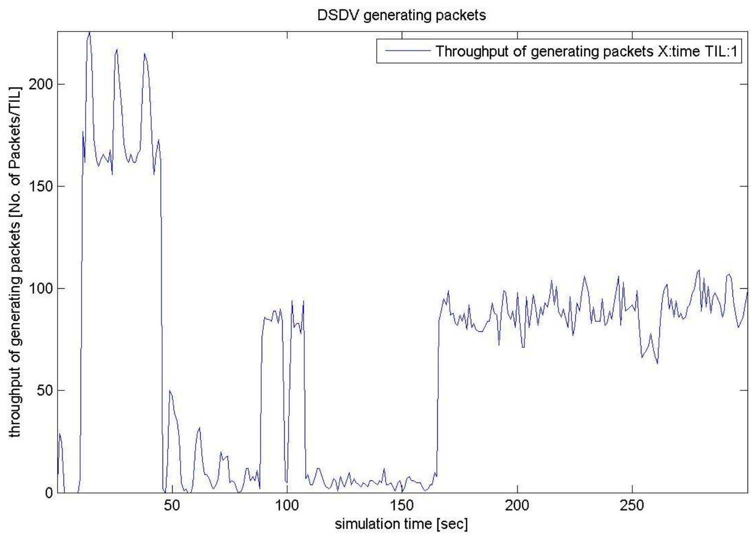

Figure 4.6: Packets generation ratio in DSDV.

The ratio of lost packets in DSDV is too high, because DSDV has a bad performance in mobile pattern. Power level in each node is also effect on packet loss rate, because DSDV consume power during transmitting packets more than others protocols, this is lead to some of intermediate node that involved in route to die during transmission process. The number of loss packets during this simulation is 37 packets. Figure 4.7 illustrate the packet loss ratio.

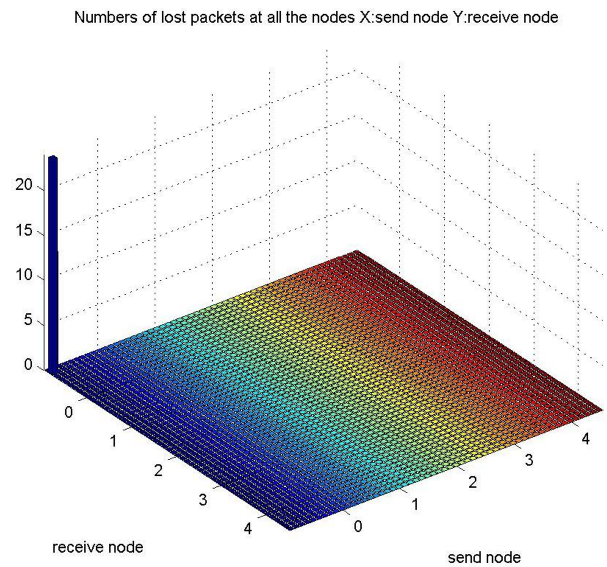

Figure 4.7: Packets loss ratio in DSDV.

4.3.3. Hybrid:

In case of proposed switching mechanism for Hybrid protocol, this study apply same scenario that had applied on AODV and DSDV. However, it is obviously that the dropping packets rate is quite low even with high mobility pattern and high rate of node vicinity; this is because of hybrid mechanism that using AODV and DSDV both and switch between these two protocols according to specific criteria; this study will discuss hybrid mechanism later on. Therefore, the number of dropped packets is just 102 packets. Figure 4.8 illustrate packet drop ratio.

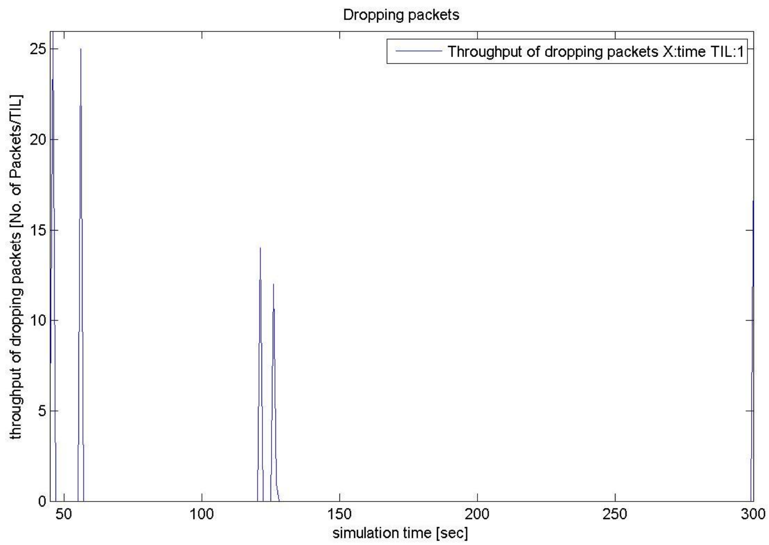

Figure 4.8: Packets loss ratio in Hybrid approach.

The number of generating packets was much higher than others protocols that implemented in the same scenario. However, the number of generating packets is 28507 packets during simulation time (300s). Figure 4.9 illustrate the ratio of generating packets by hybrid approach.

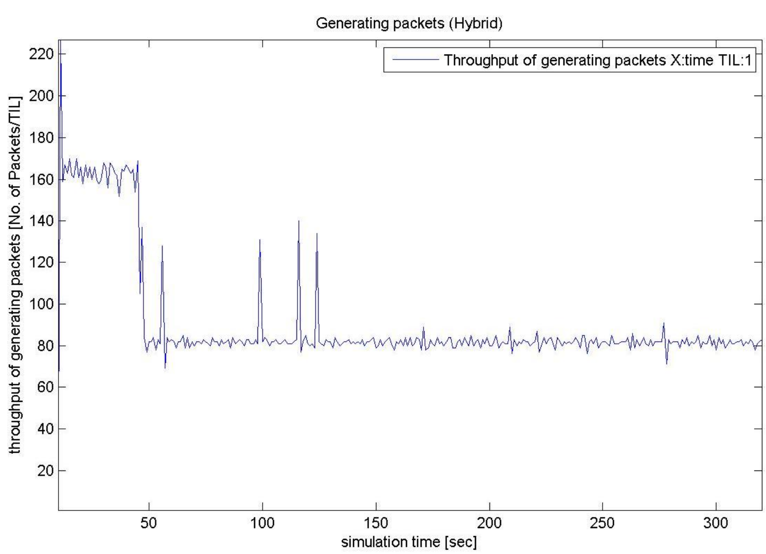

Figure 4.9: Packets generation ratio in Hybrid approach.

Figure 4.10 illustrates that the hybrid approach has low packet loss ratio during the simulation time over all protocols, even with high mobility or low mobility pattern. In addition, in hybrid approach the power consumption not effect badly on packet loss rate. The number of lost packets during simulation time is 24 packets only.

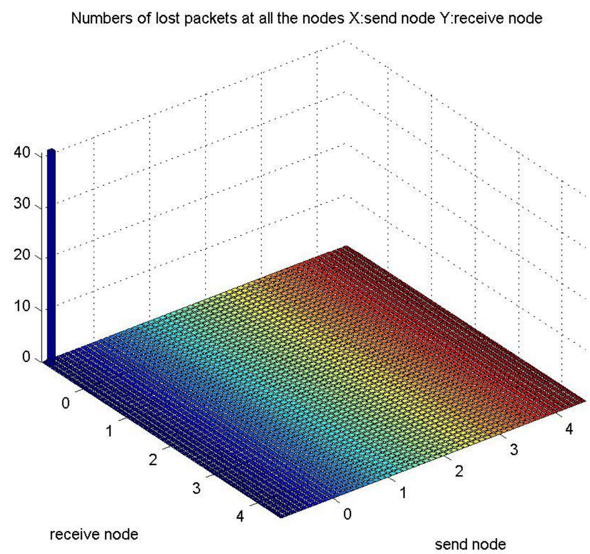

Figure 4.10: Packets loss ratio in Hybrid approach

4.3.4. End-to- End Delay:

In term of end-to-end delay; the best performance was reached by DSDV. DSDV has the lowest ratio when looking at the average of delay of delivering packets to destination node, because of DSDV chose the shortest path from source node to destination, even with high mobility rate.

While AODV shows the higher delay during simulation because; when mobility is high, more packets may travel over non-optimal routes with larger hop counts, which may be stored in a route cache. Therefore, these packets will suffer longer end-to-end delay than the packets transmit over the shortest path.

Hybrid approach gives better performance in terms of end-to-end delay than AODV (reactive), because hybrid approach combines AODV and DSDV specification, and switch or switch back between these two protocols according to some criteria that will discussed later.

In addition, to measure the delay, the packets have been captured during simulation time for each scenario and analysis has been done. Figure 4.11 illustrates the delay ratio for each protocol during simulation time.

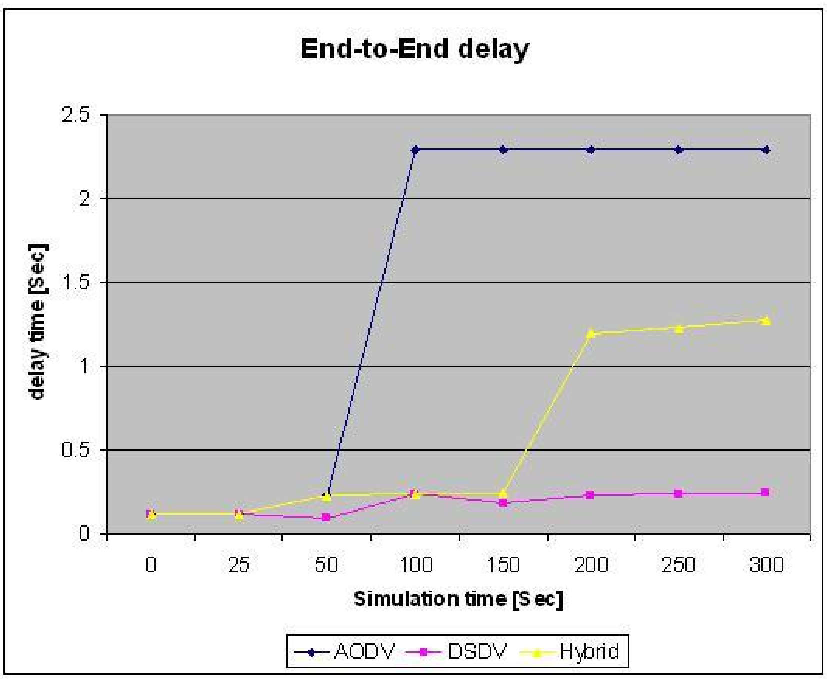

Figure 4.11: End-to-End delay.

The following table illustrates all results that had obtained from simulation scenario for all protocols.

Table 4.2: Results

PROTOCOL	NO. OF DROPPED PACKETS	NO. OF PACKETS GENERATION	NO. OF LOST PACKETS	END-TO-END DELAY
AODV	117	26875	30	More than 2 sec
DSDV	149	20598	37	Less than 0.5 sec
Hybrid	102	28507	24	More than 1 sec

4.4. Discussion:

In this section, the comparison between the performance of AODV (reactive), DSDV (proactive), and hybrid approach has been done, in terms of packets throughput and end-to-end delay.

4.4.1. AODV vs. DSDV:

The simulation experiment uses in this study to compare the performance between AODV and DSDV has the same scenario. However, the observations made in this study shows that AODV has a better drop packet ratio compare with DSDV. The number of packets that dropped during simulation time by using AODV is 117 packets; while the number of packet that dropped using DSDV is 149 packets. Therefore, this rate of dropped packet shows that DSDV perform badly at high mobility pattern, because DSDV cannot maintain its routing table update process with high topology changes.

AODV manages the packets generation rate better than DSDV. The number of packets generating by AODV is 26875 packets, when DSDV generate 20598 packets only. In addition, that obviously clear that the rate of loss packets by DSDV is greater than loss rate by AODV. The number of loss packets in DSDV is 37 packets, when number of loss packets by AODV is 30 packets.

In terms end-to-end delay, DSDV have very low delay compare with AODV because the longer time that been taken to send RREQ and wait for neighbors nodes respond and sent back RREP. AODV suffer from long delay, which effect badly on routing performance. AODV has average delay more than (2s), when DSDV has less than (0.5s). Figure 4.11 illustrates the end-to-end delay ratio between AODV and DSDV.

4.4.2. Hybrid vs. AODV:

Hybrid approach is combine between two routing mechanism (AODV and DSDV). However, and according to specific criteria such as (mobility rate, node vicinity, and power level), will switch or switch back between (AODV and DSDV). Therefore, in order to compare the performance between hybrid approach and AODV, the simulation results shows hybrid approach has better performance than AODV, in terms of drop packet rate. The number of dropped packets by AODV is 117 packets, while in hybrid approach its 102 packets only.

The number of packets that were generated by AODV is 26875, while in hybrid, the number of packets generation was 28507 packets.

In terms of loss packet rate, simulation results show hybrid approach has a better rate compare with AODV. The number of lost packets in hybrid is 24 packets and in AODV is 30 packets.

From the results it shows that the average of end-to-end delay in AODV is much higher than hybrid, because when mobility rate is low hybrid approach used DSDV component to do the routing process unlike AODV its consume much time with send and receive RREQ and RREP therefore, hybrid approach perform better than AODV in large scale networks. Figure 4.11 illustrates the end-to-end delay ration between AODV and hybrid approach. It is easy to notice that AODV has average delay more than (2s) when hybrid has average delay less than (1.5s).

4.4.3. Hybrid vs. DSDV:

The simulation results shows that the hybrids approach perform well in high mobility model. Nevertheless, DSDV fail to maintain its routing table during high topology changes. The simulation results shows that the number of dropped packets by DSDV during simulation is 149 packets, when in hybrid it just 102 packets.

When looking at the number of generating packets, it can be easily seen that hybrid approach perform better than DSDV. The number of generating packets by hybrid is 28507 packets, while DSDV generate 20598 packets only.

In addition, simulation results shows that DSDV completely fail with high mobility model, as DSDV loose 37 packets during simulation, on the other hand hybrid approach shows a good performance with loss packets rate, hybrid just loss 24 packets during simulation, that is because hybrid approach use AODV component when the rate of mobility is high.

Although form previous results show that hybrid approach perform better than DSDV, but in term of average end-to-end delay, DSDV shows a good delay ratio compare with hybrid approach. From figure 4.11 can be noticed that DSDV has an average delay less than (0.5s), when hybrid has less than (1.5s).

4.4.4. Hybrid Protocol Performance:

This study proposed new switching mechanism for hybrid routing protocol for mobile ad hoc network (MANET), the need of hybridization technique comes to utilize the strength of each of proactive and reactive techniques. Therefore, this hybrid routing protocols have the advantage of both proactive and reactive routing protocols to reduce the average end-to-end delay and power consumption. Hybrid routing protocols try to maximize the benefit of proactive routing and reactive routing by utilizing proactive routing protocol when the node vicinity is high (number of neighbors nodes within power range), and reactive routing protocol, when power level is low (in order to extend the battery life), and the mobility ratio is high.

The hybrid routing protocol contains AODV and DSDV combined together with switching mechanism. One of the reasons to choose AODV and DSDV in this study is these two protocols based on Bellman-Ford routing mechanism [10][17], so they work together in compatible way. Apart from that, the most common and popular proactive routing protocol for mobile ad hoc networks in term of number of researchers is AODV [28], because AODV choose the shortest and freshest path from source to destination, and AODV is loop free problem [30], (see 2.3.2.1.). On the other hand DSDV is considered as the most popular proactive routing protocol for mobile ad hoc networks, because DSDV reduce the overhead in the network by providing a single path to a destination and maintains only the best path. DSDV also reduce the average end-to-end delay by choosing the shortest path [7][13], (see 2.3.1.1.).

Hybrid protocol contains two sub protocols, one of them act proactively (DSDV) and the other one act reactively (AODV). Therefore, the switching mechanism between hybrid protocol components should fulfill under specific criteria. In addition, based on these criteria, the hybrid routing protocol decide when to act proactively or reactively. The criteria are as follow:

- **Power Consumption**: the hybrid protocol will calculate the power level for each node in the network topology. Thereafter, base upon this calculation, the protocol will behave as proactive or reactive mode.

- **Mobility**: it defines the rate of movement of the node in the network. If the mobility speed level of the node is high, that means the topology around the node is changing quickly; this lead to the node required to behave reactively.

- **Vicinity**: the last criteria here relate to the node's vicinity density level, and it is define as a number of nodes per square unit. Therefore, when the node's vicinity density level is high, its means that there are many nodes within node range. This leads to the node required to behave proactively.

Figure 4.12 illustrates the switching mechanism code for hybrid routing protocol that used in this study base on the previous criteria. In addition, the hybrid routing protocol code contains DSDV code and AODV code combined together with switching code; this code written in C++.

```
// DSDV source code Part.

void
DSDV_Agent::forwardPacket (Packet * p)
{
      if (NoNB <= 5 && MobileCheck == TRUE)

            AODV::command(2, "start");   // call AODV main function.

            else if (NoNB <= 5 && EnModClass.Energy() < 14.4)

      AODV::command(2, "start");        // call AODV main function.

            else if (MobileCheck == TRUE && EnModClass.Energy() <
14.4)

      AODV::command(2, "start");    // call AODV main function.
      else {

continue with to DSDV

// DSDV source code Part
```

Figure 4.12: Switching mechanism code from DSDV to AODV in C++.

In this part of code, the node initialized in DSDV mode (proactive), however, the node start checking the three criteria that mentioned before. Checking code contain three condition, the first condition check the value of node vicinity (`NoNB <= 5`), number of neighbors nodes <= 5 nodes (10% of all nodes in simulation experiments), (vicinity is low), and node mobility rate (`MobileCheck == TRUE`) (mobility rate is high), mobility ratio of node is obtained from gridkeeper header file that gives coordinates of node's locations, thereafter, calculate the new node's location and compare it with its previous location, if the rate of location changing is high each period time, that is lead to (`MobileCheck == TRUE`) , otherwise, (`MobileCheck ==`

`FALSE`). If the value of this condition becomes TRUE, the node will switch from DSDV to AODV by calling AODV main function.

The second condition check the value of node vicinity (`NoNB <= 5`), number of neighbors nodes <= 5 nodes (vicinity is low), and node's power level (`EnModClass.Energy() < 14.4`), when 14.4 watt consider as a half of battery level, near to 30 minutes (power level is low). If the value of this condition becomes TRUE, the node will switch from DSDV to AODV by calling AODV main function.

The last condition check both node mobility rate (`MobileCheck == TRUE`) (mobility rate is high), and node's power level (`EnModClass.Energy() < 14.4`) (power level is low), if the value of this condition become TRUE, the node will switch from DSDV to AODV by calling AODV main function.

Otherwise, if the value of these three condition = FALSE, the node will stay on DSDV mode, act proactively.

```
// AODV source code Part

void
AODV::sendRequest(nsaddr_t dst) {

if (NoNB > 5 && MobileCheck == FALSE)

DSDV_Agent::command(2,"start-dsdv");      // call DSDV main function.

      else if (NoNB > 5 && EnModClass.Energy() >= 14.4)

DSDV_Agent::command(2,"start-dsdv");    // call DSDV main function.

      else if (MobileCheck == FALSE && EnModClass.Energy() >= 14.4)

      DSDV_Agent::command(2,"start-dsdv");  / call DSDV main
function.

else {

 continue with AODV

// DSDV source code Part
```

Figure 4.13: Switching mechanism code from AODV to DSDV in C++.

In this part of code, the node is already in DSDV mode (proactive), therefore, the node start checking again the three criteria that mentioned before. Checking code contain three condition, the first condition check the value of node vicinity (`NoNB > 5`), number of neighbors nodes > 5 nodes (vicinity is high), and node mobility rate (`MobileCheck == FALSE`) (mobility rate is low), if the value of this condition become

TRUE, the node will switchback from AODV to DSDV by calling DSDV main function.

The second condition check the value of node vicinity (`NoNB > 5`), number of neighbors nodes > 5 nodes (vicinity is high), and node's power level (`EnModClass.Energy() >= 14.4`), when 14.4 watt consider as a half of battery level, near to 30 minutes (power level is high). If the value of this condition becomes TRUE, the node will switchback from AODV to DSDV by calling DSDV main function.

The last condition check both node mobility rate (`MobileCheck == FALSE`) (mobility rate is low), and node's power level (`EnModClass.Energy() >= 14.4`) (power level is high), if the value of this condition become TRUE, the node will switchback from AODV to DSDV by calling DSDV main function.

Otherwise, if the value of these three condition = FALSE, the node will stay on AODV mode, act reactively.

4.5. Summary:

This chapter has presented results derived from simulations experiments conducted at the end of this project. This chapter presents the results in details of simulation that have been done upon hybrid routing protocol, AODV, and DSDV. Thereafter, the comparison between these protocols has been discussed in details, in order to evaluate the performance.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1. Introduction:

This chapter summarizes this project briefly, highlighting all the main features in this project and shows the achievements of this study.

5.2. Achievements:

Proposing a new hybrid routing technique for mobile ad hoc networks This new hybridization routing strategy is combined between proactive (DSDV) and reactive (AODV) techniques. In addition, this new hybrid technique has its own switching mechanism to switch between AODV and DSDV. This switching process fulfill under specific criteria, in which based upon these criteria, the hybrid routing protocol decide when to act proactively (utilize DSDV) or reactively (utilize AODV). These criteria as follow:

- **Power Consumption**
- **Mobility**
- **Vicinity**

This proposed hybrid technique is performing well in large scale networks and high mobility models because; utilize the advantages of both its components (AODV, and DSDV). AODV is suited with high mobility model, and DSDV is suited with large networks.

A performance comparison of three different mobile ad-hoc routing protocols (AODV, DSDV and Hybrid) was presented. As a result of this study, it can be said that AODV performs very poor in larger networks as it shows extreme high delays, but it is perform well with high mobility pattern. The performance of hybrid was very good in large network sizes and high mobility pattern. Additionally, DSDV perform poor in high mobility pattern.

5.3. Challenging:

There are many challenges faced during this study. One of the challenges is how to install NS2 and witch version of NS2 is suited with for ad hoc routing, and witch operating system is compatible with NS2.

The most difficult obstacle that had overcome is learning NS2 from beginning, and how to deal with C++ and OTcl script to create the network topology with 50 nodes and fulfill the simulation setup. In addition, another problem during simulation ending is how to extract the result from the trace files.

5.4. Future Work:

One of the future works that can be done is to extend this study by implementing real world scenario to see the results of having the proposed hybrid mechanism. It is interesting to see if the results suggested from the simulation experiments could match the real world implementation.

As another future work improving and enhancing the routing process is a matter of concern. Hopefully the concept of this study can be a motivation for further investigation in ad hoc networks routing field with considering more switching criteria like connectivity, and so on. By improving the performance of proposed protocol, it will help in finding more exact quality values.

Hopefully this study gives at list an idea and spreading the knowledge in the ad hoc routing field. Also, it is hoped that the proposed protocol can be used by other researchers to develop more researches in mobile ad hoc networks.

REFRENCES

1. Changling L., Jörg K. (2005). *A Survey of Mobile Ad Hoc network Routing Protocols*. : The University of Magdeburg .
2. Anders L., (2004). *A New Routing Strategy for Mobile Ad Hoc communication*. Master Thesis. University of Kalmar, SWEDEN.
3. Bassam H. (2000). *Internet Routing Architectures*. Cisco Press.
4. Krishna G. (2006). *Routing Protocols in Mobile Ad-hoc Networks*. Master Thesis. Ume°a University, SWEDEN.
5. ELIZABETH M. , CHAI-KEONG T. (1999). *A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks*.: University of California. Georgia institute of technology.
6. Tony L. , Nicklas H. (1998). *Routing protocols in wireless ad hoc networks – a simulation study*.: Lulea University of Technology.
7. Charles E. Perkins and Pravin Bhagwat, *DSDV Routing over a Multihop Wireless Network of Mobile Computers*, chapter 3, pp. 53{74, In Perkins [1], 2001.
8. Mehran A., Tadeusz W., Eryk D. (2003). *A review of routing protocols for mobile ad hoc networks*. Australia : Motorola Australia Research Centre, 12 Lord St., Botany, NSW 2525.
9. Perkins C. E., Bhagwat P., *Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers*, Comp. Commun. Rev., Oct. 1994, pp. 234-44.
10. Johnson D., Maltz D. A. (1996). *Dynamic source routing in ad hoc wireless networks in Mobile Computing* : (T. Imielinski and H. Korth, eds.), Kluwer Academic.

11. Daniel L. (2003). *A comprehensive overview about selected Ad Hoc Networking Routing Protocols.*

12. Uwe R. (2006). *Highly dynamic destination-sequenced distance-vector routing*. http://wiki.uni.lu/secan-lab/Highly+Dynamic+Destination-Sequenced+Distance-Vector+Routing.html

13. Danny D. Patel. *Energy in ad-hoc networking for the pico radio*. Technical report.

14. Guoyou H., *Destination-sequenced distance vector (DSDV) protocol*. Technical report, Helsinki University of Technology, Finland.

15. C. Chiang, Tsai T. C., Liu W. and Gerla M. (1997). *Routing in clustered multihop, mobile wireless networks with fading channel* . : The IEEE SICON.

16. Shree M. , Garcia L. J. (1996). *An ecient routing protocol for wireless networks*. : University of California at Santa Cruz, Santa Cruz, CA 95064.

17. Pei G., Gerla M. and Chen T.W. (2000). *Fisheye State Routing in Mobile Ad Hoc Networks*. Taiwan.

18. Atsushi I., Ching-C. C., Guangyu P.,Mario G., and Tsu-Wei C.. *Scalable Routing Strategies for Ad Hoc Wireless Networks*.IEEE (1999).

19. Chen W. , Gerla M., *Global state routing: A new routing schemes for ad-hoc wireless networks* . IEEE ICC'98, pp. 171–175.

20. Guangyu P., Mario G. and Xiaoyan H. LANMAR: *Landmark Routing for Large Scale Wireless Ad Hoc Networks with Group Mobility*.: University of California.

21. Tony L. , Niclas H. (1998). *Routing Protocols in ad-hoc networks*. Master Thesis. Lulea University of Technology. Stockholm.

22. Zygmunt J. H., Jing D. , Ben L. , Panagiotis P., and Sajama S. (2001). *Wireless Ad Hoc Networks*. : Cornell University.

23. Markku M. (2006). *Table-Driven Routing Protocols in Ad-hoc Mobile Wireless Networks*. : University of Helsinki, Department of Computer Science.

24. Victor o. K. Li, Zhenxin Lu (2004). *Ad Hoc Network Routing*. Hong Kong, China: Department of Electrical & Electronic Engineering University of Hong Kong.

25. Jie W. (2002). *An Extended Dynamic Source Routing Scheme in Ad Hoc Wireless Networks*. Florida Atlantic University.

26. David B. J., David A.M., and Josh B. (2006). *DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks*. Carnegie Mellon University.

27. Chai K. T. (1996). Associativity-*based routing for ad-hoc mobile networks*. University of Cambridge.

28. Dube R. (1997). *Signal stability based adaptive routing for ad hoc mobile networks*: IEEE Computer Community.
29. Padmini M. (2000). Routing *Protocols for Ad Hoc Mobile Wireless Networks*: http://www.cis.ohio-state.edu/~jain/cis788-99/adhoc_routing/index.html.
30. Zhijiang C., Georgi G., and Stamatis V. (2006). *Routing Protocols for Mobile Ad-hoc Networks Current Development and Evaluation*. Delft University of Technology. Netherlands.
31. Jose C. R., Tuna V., Mohammad A,. and Raimo K.(2006). *Scalable Hybrid Ad Hoc Routing Approach*. Helsinki University of Technology. FINLAND.
32. Adel Ben Mnaouer, Lei Chen, Chuan Heng Foh , Juki Wirawan Tantra. *A New Polymorphic Multicast Routing Protocol for MANET*. Nanyang technological University. Singapore.
33. The Network Simulator official website - ns-2 website: http://www.isi.edu/nsnam/ns.

APPENDIX A

OTcl Simulation Code

A 50 mobile-node example for ad-hoc simulation with DSDV or AODV or Hybrid

```
# Define options
set val(chan)           Channel/WirelessChannel    ;# channel type
set val(prop)           Propagation/TwoRayGround   ;# radio-
propagation model
set val(netif)          Phy/WirelessPhy            ;# network
interface type
set val(mac)            Mac/802_11                 ;# MAC type
set val(ifq)            Queue/DropTail/PriQueue    ;# interface queue
type
set val(ll)             LL                         ;# link layer type
set val(ant)            Antenna/OmniAntenna        ;# antenna model
set val(ifqlen)         50                         ;# max packet in
ifq
set val(nn)             50                          ;# number of
mobilenodes
set val(rp)             ""                       ;# routing protocol
should be DSDV or AODV or Hybrid
   ;# X dimension of                 set val(x)              800
topography
   ;# Y dimension of                 set val(y)              800
topography
   ;# time of simulation              320           set val(stop)
end
set val(sc)             "home/aws/setdest1"
set energymodel  EnergyModel;
set p_rx        281.0
set p_tx        281.0
set initialenergy 104000
  [new Simulator]            set ns
set tracefd       [open aodv50.tr w]
set windowVsTime2 [open win.tr w]
set namtrace      [open aodv.nam w]
$ns use-newtrace
$ns trace-all $tracefd
$ns namtrace-all-wireless $namtrace $val(x) $val(y)
```

```
# set up topography object
set topo       [new Topography]
$topo load_flatgrid $val(x) $val(y)
create-god $val(nn)
#  Create nn mobilenodes [$val(nn)] and attach them to the channel.
# configure the nodes
        $ns node-config -adhocRouting $val(rp) \
 -llType $val(ll) \
 -macType $val(mac) \
 -ifqType $val(ifq) \
 -ifqLen $val(ifqlen) \
 -antType $val(ant) \
 -propType $val(prop) \

-phyType $val(netif) \
 -energyModel $energymodel \
 -rxPower $p_rx \
 -txPower $p_tx \
 -initialEnergy $initialenergy \
 -channelType $val(chan) \
 -topoInstance $topo \
 -agentTrace ON \
 -routerTrace ON \
 -macTrace OFF \
 -movementTrace ON



for {set i 0} {$i < $val(nn) } { incr i } {
    set n($i) [$ns node]
}

# Provide initial location of mobilenodes

set x 5.0
set y 70.0
set temp 30
set temp1 60
set n($i) [$ns node]
set k 10
for {set i 0} {$i < $k} {incr i} {
$n($i) set X_ $x
$n($i) set Y_ $y
$n($i) set Z_ 0.0
set x [expr $x + $temp]
set y [expr $y + $temp1]

}
set x 140.0
set y 70.0
set temp 30
set temp1 60
set k1 20
set n($i) [$ns node]
for {set i 10} {$i < $k1} {incr i} {
$n($i) set X_ $x
$n($i) set Y_ $y
$n($i) set Z_ 0.0
set x [expr $x + $temp]
```

```
set y [expr $y + $temp1]


}

set x 280.0
set y 70.0
set temp 30
set temp1 60
set k2 30
set n($i) [$ns node]
for {set i 20} {$i < $k2} {incr i} {
$n($i) set X_ $x
$n($i) set Y_ $y
$n($i) set Z_ 0.0
set x [expr $x + $temp]
set y [expr $y + $temp1]
}
set x 420.0
set y 70.0
set temp 30
set temp1 60
set k3 40
set n($i) [$ns node]

for {set i 30} {$i < $k3} {incr i} {
$n($i) set X_ $x
$n($i) set Y_ $y
$n($i) set Z_ 0.0
set x [expr $x + $temp]
set y [expr $y + $temp1]
}

set x 560.0
set y 70.0
set temp 30
set temp1 60
set k4 50
set n($i) [$ns node]
for {set i 40} {$i < $k4} {incr i} {
$n($i) set X_ $x
$n($i) set Y_ $y
$n($i) set Z_ 0.0
set x [expr $x + $temp]
set y [expr $y + $temp1]
}

# Set a TCP connection between node_(0) and node_(1)
set tcp [new Agent/TCP/Newreno]
$tcp set class_ 2
set sink [new Agent/TCPSink]

$ns attach-agent $n(0) $tcp
$ns attach-agent $n(1) $sink
$ns connect $tcp $sink
set ftp [new Application/FTP]
$ftp attach-agent $tcp
$ns at 10.0 "$ftp start"
# Printing the window size
proc plotWindow {tcpSource file} {
```

```
global ns
set time 0.01
set now [$ns now]
set cwnd [$tcpSource set cwnd_]
puts $file "$now $cwnd"
$ns at [expr $now+$time] "plotWindow $tcpSource $file" }
$ns at 10.1 "plotWindow $tcp $windowVsTime2"
# Define node initial position in nam
for {set i 0} {$i < $val(nn)} { incr i } {
# 30 defines the node size for nam
$ns initial_node_pos $n($i) 30
}
# Telling nodes when the simulation ends

for {set i 0} {$i < $val(nn) } { incr i } {
    $ns at $val(stop) "$n($i) reset";
}

# ending nam and the simulation

$ns at $val(stop) "$ns nam-end-wireless $val(stop)"
$ns at $val(stop) "stop"
$ns at 320.01 "puts \"end simulation\" ; $ns halt"
proc stop {} {
    global ns tracefd namtrace
    $ns flush-trace
    close $tracefd
    close $namtrace
    exec nam aodv.nam &
exit 0
}
$ns run
```

APPENDIX B

C++ Hybrid Code

```
// DSDV source code Part.

void
DSDV_Agent::forwardPacket (Packet * p)
{
if (NoNB <= 5 && MobileCheck == TRUE)

AODV::command(2, "start");                  // call AODV main
function.

else if (NoNB <= 5 && EnModClass.Energy() < 14.4)

AODV::command(2, "start");           // call AODV main function.

else if (MobileCheck == TRUE && EnModClass.Energy() < 14.4)

AODV::command(2, "start");     // call AODV main function.
else {

continue with to DSDV

// DSDV source code Part

// AODV source code Part

void
AODV::sendRequest(nsaddr_t dst) {

if (NoNB > 5 && MobileCheck == FALSE)

DSDV_Agent::command(2,"start-dsdv");             // call DSDV main
function.

else if (NoNB > 5 && EnModClass.Energy() >= 14.4)

DSDV_Agent::command(2,"start-dsdv");        // call DSDV main
function.
```

```
else if (MobileCheck == FALSE && EnModClass.Energy() >= 14.4)

DSDV_Agent::command(2,"start-dsdv");  // call DSDV main function.

else {
continue with AODV


// DSDV source code Part
```