Mobile and Wireless Communications
Mekelweg 4,
2628 CD Delft
The Netherlands
http://www.ewi.tudelft.nl

WMC 2009

M.Sc. Thesis

Evaluation of Dependability of MAC and Routing Protocols in Personal Networks

Mohamed Hawas

Evaluation of Dependability of MAC and Routing Protocols in Personal Networks

Thesis

submitted in partial fulfillment of the
requirements for the degree of

Master of Science

in

Mobile and Wireless Communications

by

Mohamed Hawas
born in Giza, Egypt

This work was performed in:

Mobile and Wireless Communications Group
Department of Telecommunications
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology

Delft University of Technology

Delft University of Technology
Department of
Telecommunications

The undersigned hereby certify that they have read and recommend to the Faculty of Electrical Engineering, Mathematics and Computer Science for acceptance a thesis entitled **"Evaluation of Dependability of MAC and Routing Protocols in Personal Networks"** by **Mohamed Hawas** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: 24/7/2009

Chairman: _____
Prof.Dr.Ir. Ignas Niemegeers

Advisor: _____
Ir. Javad Vaziefehdan

Committee Members: _____
Dr.Ir. Ertan Onur

_____
Dr.Ir. Arjan van Genderen

_____
Dr.Ir. R. Venkatesha Prasad

Abstract

In this project we have investigated how to enable dependable routing inside personal network (PN) cluster. Dependable routing among different clusters is out of the scope of this thesis. In practice, personal network cluster is considered as mobile ad-hoc network (MANET) with heterogeneous radio access technologies and nodes with different capabilities (processing power, battery level,etc.) Although a single proposal for dependable routing inside the personal network's cluster could span over all OSI layers, but we considered only the network and medium access control (MAC) layers in this project. We concerned routing protocols because they have the basic task of routing the data packets from source to destination. Routing protocols should be independent of radio technology implemented beneath. And they should support IP addressing scheme, uni-directional links and they should have low software complexity. These metrics are needed to give high performance routing in personal networks with heterogeneous nature and are called qualitative performance metrics [98]. We introduced one of the routing protocols which is designed for mobile ad-hoc networks (MANETs) as suitable candidate for allowing dependable routing in personal networks after some modifications on components design. Those modifications are mentioned in our thesis with possible implementations. We noticed that, default implementation of this protocol has shown best performance interacting with 802.11 MAC layer than other competitive mobile ad-hoc network (MANET) protocols regarding throughput performance. For dependable performance comparison among different protocols we used qualitative performance metrics [98] which are: packet delivery ratio, routing overhead and end-to-end delay. We also considered power consumption as a dependability metric because less power consumption means longer network lifetime. And this will lead to longer time duration the network is able to provide the required service. After studying the effect of MAC layer on the performance, we conclude that default MAC layer designed for IEEE 802.11 wireless technology doesn't guarantee dependability for personal networks. MAC layer couldn't exploit network resources efficiently. The position of MAC layer at the bottom of OSI layer stack has influenced the overall network performance. Inter-operation between medium access control (MAC) and network layer has shown large differences in both delay and throughput performance with different routing protocols. For our investigation on MAC layer we used performance metrics which have remarkable effect on the performance of any application such as: throughput (mainly controlled by MAC layer) , packet delivery ratio (which is routing protocol dependent) and delay (an important performance metric especially for real time applications). Under different simulation scenarios, OLSR routing protocol has shown better performance than DSR in terms of both delay and throughput. Concerning the heterogeneity nature of personal network cluster, a dependable routing solution must include a QoS mechanism. Such mechanism should be able to share heterogeneous network resources among different traffic demands according to individual QoS agreements. This thesis is organized as follow: Chapter one presents an overview of personal networks (PNs) , dependability and problem definition. Because a cluster in a personal network is modeled as an ad-hoc network, in Chapter two we have presented

overview of routing in ad-hoc networks. We have discussed the architecture of ad-hoc routing protocols in general and we presented a quantitative performance analysis for single path and multi path routing protocols. Chapter three introduces routing in a proposed PN scenario. We presented an expected scenario for a personal network with respect to network service area, size and degree of mobility. We made a performance comparison among different ad-hoc routing protocols under different varied conditions. We could choose only three routing protocols which have shown better performance in a typical personal network scenario than other protocols. Also in Chapter three we presented the performance of routing protocols under some realistic traffic scenarios (FTP, HTTP, VoIP and video conference). The performance of routing protocols with respect to power consumption has also been addressed in Chapter three. As a conclusion from Chapter three, we were able to introduce a dependable routing protocol for personal networks which is able to offer dependable routing service inside a single PN cluster. In Chapter four, we introduced our investigation on dependability of 802.11 MAC layer and its influence on overall performance. Chapter five is a summary chapter, which presented our final conclusions and summarized our work.

Acknowledgments

I would like to thank both Prof.Dr.Ir.Ignas Niemegeers and Ir.Javad Vaziefehdan for their kindly support in this project. The output from periodical meeting discussions with Javad and earlier recommendations from Prof.Niemegeers were helpful to me through my research.

Mohamed Hawas
Delft, The Netherlands
24/7/2009

Contents

List of Figures

List of Tables

1 Introduction

A person who needs to carry mobile phone, personal digital assistant (PDA) device, MP4 player and laptop will have multiple display units, keyboards, processors and other redundant facilities. Interconnectivity among these devices could reduce this redundancy. This interconnectivity may also enable sharing of resources and give a single device more capability than when this device is standalone. For example, a large desktop computer connected to the Internet with quad-core processor and very large storage capacity doesn't have to be mobile as long as it has connectivity with small cheap device held in a person's bucket. The small device can get access to huge amount of information and use the powerful processing capability to process data, read/print files and possibly connect to Internet. Introducing the concept of ubiquitous computing to our personal life could build a personalized distributed I/O system which surround us with personalized services and enable us to get access to our personal information anywhere anytime. For individual users this will be very comfortable and joyful. Examples for some services are: permanent access to personal information, multi player gaming and Internet banking. For business this would form a big revenue opportunity for service providers, mobile operators and hand held vendors. They could develop various services for customers with different ages, cultures and living standards. A long relationship with customers can be built to offer multimedia, interactive information services and many other types of services. From this point the idea of designing a personal network sounds very attractive. A personal network (PN) is a network contains all personal devices despite the geographical location of these devices. There is a trust relationship between these devices (further we will call them nodes) inside the PN. Such a network consists of several clusters, each of them works in an ad-hoc fashion. Also PN can lay on other networks (UMTS, Internet, WLAN, etc.) to transfer packets between personal nodes from different clusters as shown in Figure 1. A personal network cluster consists of some devices which surround the person (mobile phone, mp4, PDA, body sensors, etc.). Nodes inside a cluster are self organized and they communicate without need to interconnecting structure or foreign nodes (which are nodes that dont have trust relationship with other PN nodes.) Interconnection among clusters might need an external infrastructure network. In this chapter we introduce an overview of personal networks (PN) and we define the problem we want to solve which is how to enable dependable routing inside PN cluster (intra-routing). We introduce the definition of dependability and we try to show what does dependability terminology mean in the domain of routing service. We conclude that, a dependable routing protocol must satisfy dependability requirements and this can be done by implementing dependability means into components.

1.1 PN architecture

The concept of personal network (PN) differs from earlier introduced personal area network (PAN) in several points:

1. Personal node can communicate only with other personal node or with a node which has a trust relationship with it. This communication happens in heterogeneous radio access medium.

2. Communication between PN nodes doesn't have the border of few meters as it is the case in PAN. A node inside one PN cluster is connected to other node in remote cluster in a self organized way using other interconnecting network.

3. Privacy should not be violated, and personal node must keep its identity and all related personal information from being exposed by other non-trusted entity.

4. Connectivity between personal network devices should be sustained all the time without any user interaction

Personal network (PN) consists of variant of heterogeneous devices and radio access technologies. A single personal node can join a PN cluster after verifying the trust relationship with other PN nodes otherwise it is not allowed to join the cluster. Physical links between nodes are secured to prevent eavesdropping or any attempt to violate user privacy. Nodes could leave and join a cluster dynamically and communication between nodes could happen in multi-hop fashion. This rises the need for a distributed Ad-hoc routing protocol which is able to adapt quickly to dynamic topology changes. This adaptation should sustain dependability state of the network. (in next part we will talk in more details about dependability meaning.) Concerning interconnectivity between clusters, it will be carried out by gateways which are personal devices with higher capabilities (processing power, memory size, multiple interfaces, etc.) And routing information between different clusters will be carried out by IP based core network with heterogeneous radio access ends as depicted in Figure 1.1. We assume that PN cluster will not form a highly dense network (max. 100 node) , and nodes will always be connected as long as they are not out of battery. Also communication among nodes inside the PN cluster will happen in multi-hop fashion. Example of this scenario where a person moves from one room to the other inside his home or office cluster. In further investigation of routing dependability we will consider quantitative metrics mentioned in [6], thus there will be some issues like managing trust relationship, privacy and security which are not covered in our work. In next chapter we will get closer to dependability terminology and clear view of our requirement will be presented.

1.2 Definition of dependability

First use of Dependability as technical term was in 1960 by Hosford [3], and was confined with only two attributes (availability and reliability) . Later on, the term encompasses more attributes like safety, integrity and maintainability. Nowadays variation of definitions exist [2,4]. According to Laprie [4], The term Dependability is defined as: *"the*

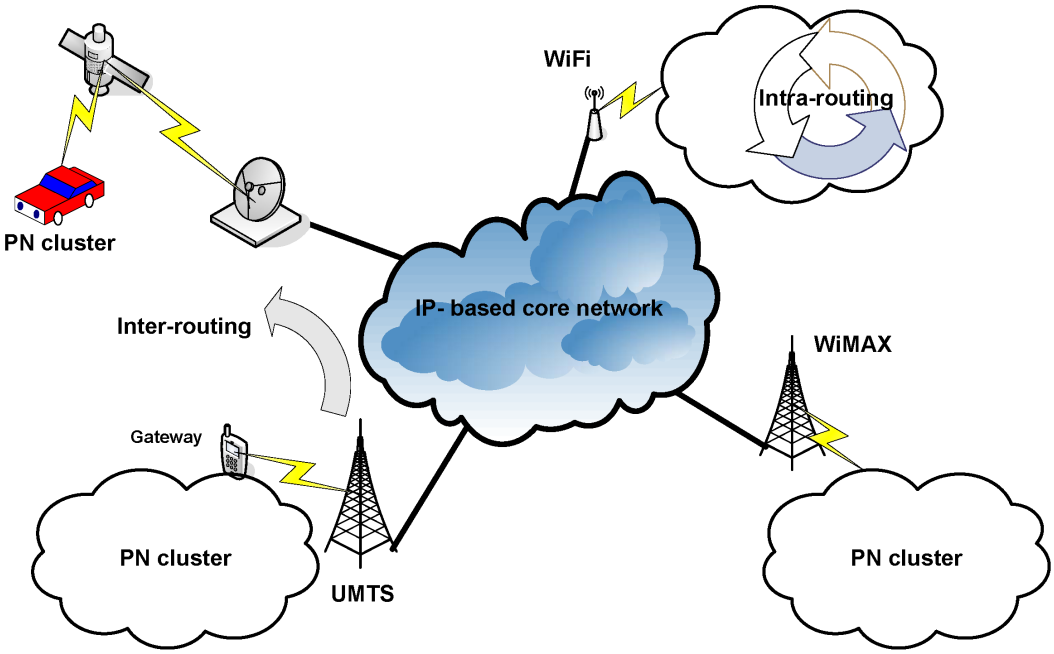


Figure 1.1: Example of routing in PN

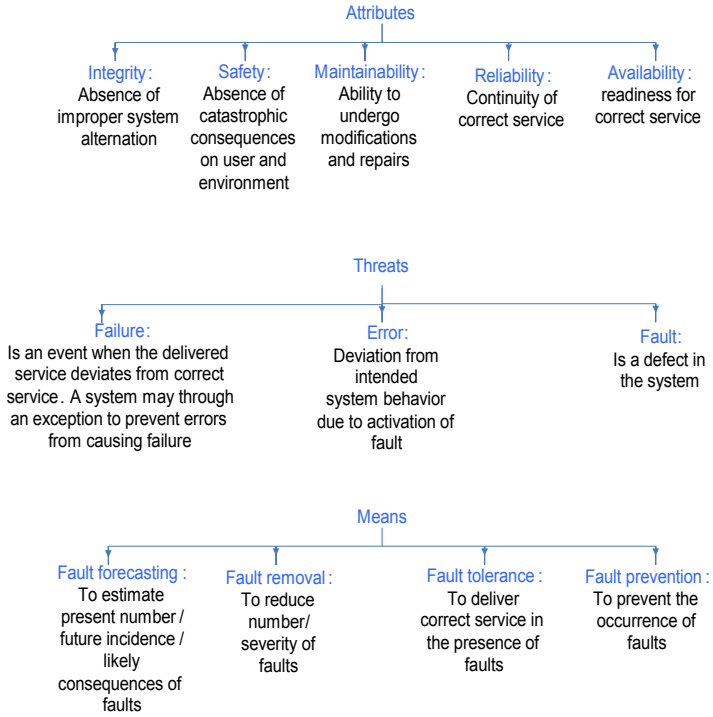


Figure 1.2: Dependability elements

trustworthiness of a computer system such that reliance can justifiably be placed on the service it delivers". The service delivered is considered as the behavior of the system as perceptible by the user, and the user himself is other system which could be human or physical system (In our case it is the application running on PN node.) Thus for different applications there will be different dependability plain. Or we can also say that QoS requirements for an application will determine its dependability requirement. Also the interaction between the system and surrounded environment is from a great deal. For the same dependability requirements, a system works under hostile wireless environment will differ from the same system which is working in wired environment. To extend the definition of dependability we mention that, the term itself is composed of three basic elements :

- **Attributes**: a way to assess the Dependability of a system
- **Threats**: things which could violate the dependability of the system
- **Means**: ways to sustain the Dependability of that system

Each of above mentioned elements is subdivided into sub-elements and we have summarized them all in Figure 1.2. As we can see, a short description for all dependability elements is introduced. That could help us in high-lighting important issues which will be helpful for the problem definition. The term dependability becomes more clear and we can translate the requirements to some technical solution. The main requirement is to enable dependable intra-routing inside a PN cluster. Before going into details, we like to clarify the threats because they are the main sources that could violate system dependability. Figure 1.3 shows the relationship between threats and how they interact with each others. First, a fault is the hypothesized cause of error when it is active, subsequently an error may cause a failure when it propagates through interactive system components. In general, system might throw an exception when an error occurs, preventing failures from happening and allows system to deliver the service. Also system failure has different modes which can be ranked according to failure severities. For our scenario, we consider the failure as the main threat and dependable routing protocol must avoid failures from happening. The failure occurs when personal network is not able to route data packets with QoS agreements. Because we consider a dependable network layer solution, the failure is expected to happen as a result of an error in some link (signal strength degradation, high amount of fading, broken link , etc.) Therefor, to sustain dependability in personal networks routing protocol has to deal with errors before they propagate and cause failures. This could be done by implementing some means of dependability into routing protocol. After introducing previous discussion we present in next section the problem definition.

1.3 Problem definition

In general, main purpose of dependable intra-routing protocol is to facilitate transferring of packets from any personal node inside a cluster to a designated destination node inside the same cluster satisfying QoS agreements during the transmission time. So far we can state the problem definition as:*"An intra-routing protocol is needed for personal*

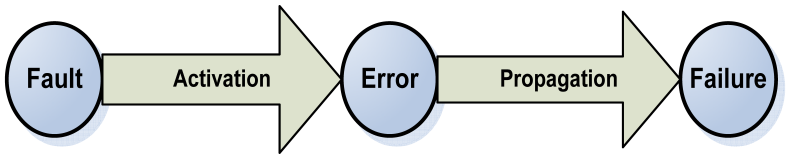


Figure 1.3: Dependability threats

network which will be able to route application data packets satisfying QoS agreements during transmission time and prevent the occurance of faults". We concern in this work intra-routing issues only and other issues which concern inter-routing mechanism are out of the scope of our work. Now we map both dependability attributes and means to our system requirements (support for dependable applications in PN cluster) as follow:

Dependability attributes which are conditions that PN must satisfy to be considered as dependable network

1. **Availability**: PN is always ready to offer a correct service for a specific application and that means:
   (a) Application data packets are able to reach their destination with agreed QoS levels (packet loss, delay, jitter,etc.)
   (b) Nodes are always connected
   (c) Data packets could be routed anytime (or with other words there will be always a valid route to carry data from source to destination)
2. **Reliability**: data transfer process (the service) cannot be hindered by any external event (link breakage, node outage, link quality degradation,etc.) and PN can continuously provide the correct service for specific application. Also routing protocol for personal network (PN) must consider QoS into design components (route discovery, route maintenance, route metrics, etc.) In next chapter we will explain more about design components for ad-hoc routing protocols.
3. **Maintainability**: routing protocol must have the ability to respond quickly to any external event that could generate faults (new nodes are merged/left the cluster, failure of an intermediate node,etc.)

Handling the rest of attributes is out of the scope of network layer and will be handled by other entities like for example research and development department of the device manufacturer or application software developer. Also because maintain and monitor links between nodes is for a great deal, using efficient link quality assessment and route update mechanisms is highly recommended. As we will see in later chapter some examples of those mechanisms implemented as components in some ad-hoc routing protocols. For each mechanism there will be some costs (bandwidth, power consumption, delay, etc.)

Dependability means these are the ways how to deal with threats and keep them

from violating system dependability. We can think of these means and ways to implement them as follow:

1. **Fault prevention** : a routing protocol should keep network connectivity and quality of links at some degree which will satisfy required service levels. Example is a multi-path routing approach which could be used for this purpose and it will do this task in two ways :
   (a) multi-path routing protocol could exploit radio channel diversity (a node could have two different radio interfaces) which will mitigate fading effects and increase throughput performance.
   (b) redundant paths will ensure delivery of packets in case of failures (path/intermediate node failure) .

   Also an efficient technique to assess link quality is considered as a fault prevention tool. Proper information will be fed to routing protocol which in turn will choose among different possible paths the most dependable path from source to destination node. As we will see later, there are so many metrics which are used to assess wireless link quality (HOP count, RTT, ETX , ETT, WCETT) [7], each of them has pros and cons. For example, hop count metric is simple and easy to implement also it doesnt add much overhead packets or processing load to routing protocol. But at the same time it doesnt guarantee optimal performance. A route with minimum hop count could have longer links which are expected to be broken in short time (in a mobile scenario). And data packets will be more suspected to signal variations due to fading effect. Another example for fault prevention is to allow for less power consumption, where as a consequence a personal node will be able to offer the service for longer time. In later chapters we will present these issues in more details.
2. **Fault tolerance**: as we will see in Chapter two, there are some techniques which could tolerate data corruption and losses. Example is an efficient error correction code which is used with multipath routing. The existence of alternative paths will reduce the effect of single path failure (which could be packet loss and/or delay).
3. **Fault forecasting**: this could be achieved by periodic assessment mechanism for service levels (delay, jitter, packet loss,etc) which could report routing protocol when any level drops beyond required value. A simple mechanism could sense signal to noise ratio (SNR) and report the protocol to replace this link before it is totally broken.
4. **Fault removal**: the routing protocol should react quickly enough when link breakage occurs and try to replace it by alternative link/route in timely manner without affecting the running service.

1.4 Summary

In this chapter we introduced the concept of dependability and we were able to define the problem of supporting dependable routing for application packets which are running inside a personal network cluster. This was an essential step before introducing the technical answer. We also introduced a general mechanism for a dependable routing protocol and how dependability attributes should be sustained with possible implementations of dependability means into routing protocol components. We define the main threat to dependable routing protocol as the failure. Failure happens when personal network is not able to route data packets with the required service levels (QoS agreements). Dependable routing protocol must prevent these failures from happening and it must be able to route application data packets satisfying QoS agreements during the whole transmission time period. Errors (link quality degradation, link/route breakage, node failure, etc.) which are the main cause of failures are supposed to co-exist. Dependable routing protocol must mitigate the occurance of errors and prevent them from propagating and causing faults. In the next chapter we will introduce a survey on ad-hoc routing protocols with a quantitative analysis of available single-path and multi-path routing protocols.

Routing Protocols for Personal Network

2

In this chapter we introduce taxonomy of routing protocols and their design components. We also show the differences between available implementations for such components. This could resolve why ad-hoc routing protocols have different behavior under various network scenarios. This part has been done based on collection of simulation based comparison from the literature. We have summarized this work and presented it in a table comparison. This work could be helpful for researchers in designing new ad-hoc routing algorithms and also for us to complete our dependable routing design investigation for PN. We have introduced a quantitative analysis of ad-hoc routing protocols. Interoperability behavior of these set of design components together under different scenarios could be analyzed. The protocol with best performance will indicate the goodness of its component interoperability performance. And a dependable routing protocol for PN cluster is proposed. The performance of different protocols is compared under variation of mobility, number of nodes and offered load. Later we have introduced multi-path routing techniques. The goal of multipath technique is to increase the reliability of data transmission and also to provide load balancing (use network resources efficiently). Different multipath routing implementations have been introduced and we show how they can enhance the performance of different routing protocols. But we mention that, multipath routing doesnt provide optimal performance all the time. Under some limitations single-path routing protocol could perform as same as multi-path routing protocol. Which after considering the cost of multipath routing (overhead, large routing tables and possibly high complexity) , multipath routing might not be recommended.

2.1 Components of routing protocols

No single mobile Ad-hoc routing protocol could fit the emerge of several wireless technologies and applications, instead routing protocols could be decomposed into some building blocks components and for each application we could construct an appropriate protocol using those building blocks [14]. By analyzing design components, their possible implementations and interaction behavior with applications, a so called component-based routing (CBR) protocol could be implemented which will accommodate different application profiles and Link state varying environment parameters at reasonable cost (overhead, delay). Adding new feature to protocol will need only some modification in one of those components instead of designing a new protocol from scratch. We can divide routing protocol components in two types (see Figure 1) , core components which are presented in most of routing protocols and auxiliary components which serve to enhance the performance of routing algorithms to satisfy some target applications or conditions. In the following sections we will introduce in some detail

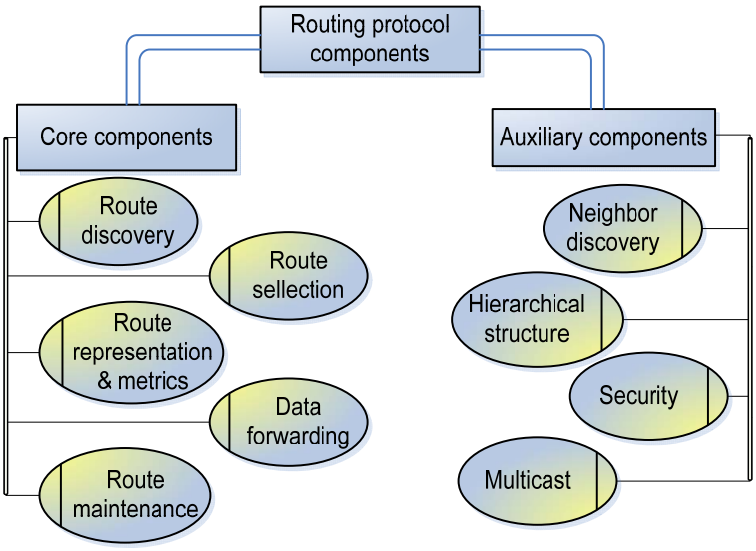


Figure 2.1: Design components of routing protocol

each of core components. We don't concern auxiliary components further in details as they are optional components. An example for auxiliary components a neighbor discovery component is implemented in some routing protocols to enhance the discovery of link breakage in timely manner specially for mobile scenario. While in other protocols it is sufficient to count on feedback from MAC layer.

2.1.1 Core components

1. **Route discovery** it is the first and main component in any protocol where possible routes must be found twoards desired destination(s). There are three approaches to carry out this discovery process and they are as follow:

   (a) *Proactive approach*: every node in the network maintains a routing table contains updated routing information to all other nodes and this happen periodically despite if the node has something to send or not. Traditionally there are two ways to exchange network topology among nodes, one is called distance vector where a node updates her neighbor with the best-known distance to every other node, and these updates are performed periodically where all or part of routing table for a node is sent to all its neighbors to update their tables. The other approach is called link-state approach where every node constructs a map of the connectivity of the network, in the form of a graph showing which nodes are connected to which other nodes and send this information to entire network. It was shown[1] that distance vector protocols have less complexity and overhead than link-state one. Examples on proactive approach are Wireless Routing Protocol (WRP) , Fish eye State

[1] $Http://www.en.wikipedia.org/wiki/Distancevectorroutingprotocol$

Routing (FSR) and Location Aware Routing with Reduced Location Maintenance (LARRLM). The advantage of this type of protocols is that alternate routes could be found quickly upon link failure, which enables short transmission delays, but in other hand there will be unnecessary control overhead because a node must keep update routing information to all destination(s) regardless how frequent these routes will be used, also maintaining unnecessary paths considered as a waste in network resource. If the network topology is changing frequently a significant part of bandwidth will be occupied due to exceeded routing overhead

(b) *Reactive approach*: a route discovery process is established only upon a request from source node which first floods a request message to the whole network searching for the destination and then it searches for optimal path to this destination[2]. Nevertheless; this flooding mechanism causes large amount of control overhead and degrades the overall network performance. Examples on reactive approach are Ad Hoc On-Demand Distance Vector Routing (AODV) , Dynamic Source Routing (DSR) and Associatively-Based Routing (ABR). The advantage of this mechanism is the reduction in overhead comparing to proactive one especially for low traffic low mobility networks. In other hand it causes delay while initiating the path which will affect negatively the service availability especially for some critical applications (time bounded).

(c) *Hybrid approach*: this type of protocols can behave reactively and pro actively at different times and they introduce a hierarchical routing structure to the network to reduce the number of forwarding nodes during route discovery or topology discovery. Each node periodically maintains the nearby topology by employing a proactive routing strategy (such as distance vector or link state) and maintains approximate routes or on-demand routes for faraway nodes. Here, we can mention Intra-Zone Proactive with Inter-Zone Reactive Routing Protocol as a hybrid approach where a network is subdivided into clusters (zones). Proactive routing is implemented between nodes inside a cluster while reactive routing is used when a source node wants to establish a connection with a destination node located outside its own cluster. Other example is Ant Agents Hybrid Multi-path for ad-hoc networks (AntHocNet) routing protocol where path set-up is reactive mechanism and multiple paths are built between source and destination. Traffic is spread stochastically over these paths according to estimated quality of each path and paths are continuously monitored and improved in a proactive way during transmission session. More implementation examples for hybrid approach are Zone Routing Protocol (ZRP) and Zone-based Hierarchical Link State Routing (ZHLS). Finally we mention that for hybrid approach less routing information is maintained than proactive case and less delay than reactive one, but in turn it has higher complexity than both approaches which needs higher computation ability and larger memory.

[2]Task of route selection component

2. **Route selection and update** after discovery of all possible routes, optimal route(s) from potentially discovered route(s) must be selected according to some defined metrics (number of hops, link quality ,link cost,etc.) As mentioned before, most of proactive routing protocols implement two different methods , one is called (distance vector) where the node periodically calculates the cost of outgoing links (according to a specific metric) to all destinations, updates own routing table and sends information to all neighbors to update their routing tables as well. When forwarding data packets, the node chooses the shortest distance (least cost) to destination. The other method is called (link state) where each node should have at least a partial overview of the network topology by periodically broadcasting link state information (state, delay, strength, etc.) of its outgoing links to all other nodes. Upon receiving this information each node updates its topology information and apply shortest path algorithm to select path to each destination. On the other hand, reactive routing has also two methods. First one is called (source routing selection) where destination node replies all route requests regardless of their quality. the source node is responsible for choosing the optimal path after receiving all available paths. The whole path is contained in the packet header and intermediate nodes dont have to calculate any routes. Source routing provides a very easy way to avoid forming loops in the network however, the size of each packet gets bigger as the number of intermediate nodes increases. Second method is called Hop-by-Hop routing where data packet carry only destination and next hop address and each intermediate node uses its routing table to find next hop twoards destination. The disadvantage of the hop-by-hop routing over source routing is that each intermediate node has to store and maintain routing information for each active route and may require sending periodic beaconing messages to its neighbors to be aware of its neighborhood.

3. **Route representation component** route representation is a way to store information obtained after route discovery and selection to use afterwards in forwarding the data. Generally there are three kinds of route representation:

   (a) *Exact route*: which indicates the exact neighbor(s) of destination node; and there are three approaches used in most routing protocols. In the first approach, every node maintains a routing table which contains entries to all destinations so that upon receiving a packet it can forward it to next hop twoards destination using availably stored information (e.g. ad-hoc on demand distance vector routing AODV and Highly Dynamic Destination-Sequenced Distance-Vector Routing DSDV). Second method is to use so called interest cache where routing table contains interest entries instead of destinations (e.g. Directed Diffusion protocol DD). An interest could be a node which evolves with source node in some task or carrying out together with it some function. By this way an interest cache only scales with number of interests This way is useful in large scale sensor networks where some group of sensor nodes relay their data to a central processing node. Such central node is considered as a gateway to some data base center, or even could have interest relation with other central nodes for further aggregation

of data. Last method used is to store the complete path in data packet header so that intermediate nodes dont have to store any route information.

(b) *Route guidance*: where the route twoards destination is described by some guidance information (cost table, a binary tree, geographical information, or a hierarchical structure)

(c) *Hybrid route*: which is combination of last two approaches, and it is often used in sensor networks.

4. **Route metric component** this component is responsible for defining all parameters (metrics) which are used to select optimal route. These metrics must consider some criteria to ensure good routing performance. Which are as follow: First criterion is that routing metrics must not cause frequent route changes to ensure the stability of the network. Second one is to capture the characteristics of the network and ensure that minimum weight paths will satisfy good performance as well. Third is to ensure that minimum weight paths could be found using efficient algorithms with low complexity. Fourth and last requirement from routing metrics is to ensure that forwarding path is loop free. Also we would like to mention some examples for different route metrics which are used with wireless ad-hoc networks routing protocols [15]:

   (a) *Minimum hop count*(HOP) is used in many of routing algorithms, but it could lead to weak signal strength along with high loss ratio because of the probable existence of long links. The primary advantage of this metric is simplicity as computing the hop count requires no additional measurements like the case with other metrics.

   (b) *Expected Transmission Count* (ETX) which is defined as expected number of transmissions is needed for successfully delivering a packet through a wireless link to a neighboring node. The weight of a path is defined as the summation of the ETX weights of all links along the path. Since both long paths and lossy paths have large retransmissions probability and consequently will have large weights under ETX, the ETX metric captures the effects of both packet loss ratios and path length. The drawback of this method that it doesnt consider interference or variation of transmission rates in the links. To calculate ETX, each node exchange some probes periodically with its neighbors. ETX of a link is calculated by [15] $ETX = 1/(D_f * D_r)$, where D_f is the probe delivery ratio on forward direction and D_r on reverse direction. Number of broadcasted probes in a network with N number of nodes is O(N).

   (c) *Expected Transmission Time* (ETT) which improves ETX by considering different transmission rates. The ETT of a link is defined as the expected duration for a successful transmission of a packet at this link, further more the weight of a path(p) is the summation of the ETT weights of the links along the path. Relation between ETX of some link L and the ETT is given by ETT(L) = ETX(L) * s/b, where s is the packet size and b is transmission rate of link L [15].For a network with n nodes and m number of data rates,the ETT computing complexity is O(n * m).

(d) *Minimum loss*(ML) metric where the purpose is to find lowest end-to-end loss probability by multiplying delivery ratios of the links in the reverse and forward directions. The authors of this method argue that using multiplication reduces the number of route changes and consequently improves network performance.

(e) *Weighted cumulative ETT* because the use of multiple non-overlapping channels could increase network throughput, care must be taken to avoid intra/inter flow interference. Intra-flow interference occurs when packets from the same flow interfere with each others. While inter-flow interference happens among concurrent flows. Weighted cumulative ETT (WCETT) metric is a variant of ETT and used where multiple channels with different wireless technologies are used. It deals with both intra- and inter-flow interference. We consider WCETT as a sum of end-to-end delay and channel diversity with some control parameter to combine both components or give a priority to one on the other. The drawback of this metric is that it neither guarantee shortest paths nor avoid inter-flow interference which might lead to choose routes in congested areas.

(f) *Metric of interference and channel switching*(MIC) which addresses the drawbacks of WCETT as each node estimates inter-flow interference by taking into account number of interfering nodes in its neighborhood and uses virtual nodes to guarantee minimum cost routes computation. MIC uses the ETT metric to calculate its value.

(g) *Modified ETX* (mETX) deals with fast link quality variation problem which is making ETX not aware enough of link variations and in addition to that it causes ETX to produce high overhead because ETX is based on average values computed on a time window interval. The mETX solves this problem because it works at the bit level. It calculates the bit error probability using the position of the corrupted bit in the probe packet and the dependence of these errors after successive transmissions (notice that probes are composed of known bit patterns).

(h) *Effective number of transmissions* (ENT) considers the variance of number of successive retransmissions per link, as it broadcasts probes and limits route computations to links with acceptable number of retransmissions. That means links with higher number of transmissions will be excluded from routing computations.

(i) *Interference aware* (iAWARE) uses SNR and SINR to monitor neighboring interference variations. It measures the average time the medium is busy when one or more of interfering neighbors is transmitting and aggregate links with lower measured values to construct the optimal data forwarding path.

In(table2.1) we introduce a summary of routing metrics characteristics.

5. **Route maintenance component**This component maintains the current path and repairs it in case of failure according to node mobility or changing in wireless

Table 2.1: Characteristics of routing metrics

Metric	Quality aware	Data rate	Packet size	Intra-flow	Inter-flow	Medium instability
HOP	no	no	no	no	no	no
ETX	yes	no	no	no	no	no
ML	yes	no	no	no	no	no
ETT	yes	yes	yes	no	no	no
WCETT	yes	yes	yes	yes	no	no
MIC	yes	yes	yes	yes	yes	no
mETX	yes	yes	yes	no	no	yes
ETN	yes	yes	yes	no	no	yes
iAWARE	yes	yes	yes	yes	yes	yes

environment. Generally there are three mechanisms which are also working in accordance with route discovery mechanism in use.

(a) *Route refreshing mechanism* which tries to confirm the validity of current route to destination by using different ways according to route discovery strategy. For example in reactive way, nodes refresh the route upon need of packet transfer using different tools like control packet, data packet, automatic update upon expiration of route lifetime (predetermined or estimated) , or a hybrid of the above. In proactive approach each node updates the whole network topology periodically or at occurrence of link changes and broadcast this information to other nodes. For the hybrid approach like in cluster based protocols, proactive refreshing is used for intra-cluster and reactive refreshing for inter-cluster.

(b) *Route failure handling* where the source or an intermediate node attempts to find a new or alternative route to the destination upon breakage in one or more links in the path , and the way it behaves also depends on routing discovery mechanism is used (see Table 2.2).

(c) *Route invalidation mechanism* which discards unusable routes due to route failure as described above. We can say that reactive route maintenance has more reactive time than proactive one, but the later delivers more overhead.

Table 2.2: Route maintenance component

Maintenance component	Proactive	Reactive	Hybrid
Route refreshing	Periodical	on-demand	hierarchical
Route Failure handling	Update routing table entry	Initiate new route discovery	Both ways
Route invalidation	available	available	available

6. **Data forwarding component** we can describe this component as the mean to forward data packets based on route information. In general there are four major approaches:

(a) *Table based forwarding* which is implemented in two ways; first one is deterministic forwarding where a node selects the next hop node twoards the destination based on a predetermined policy installed in a routing table (routing metric in use) or a packet header. The sender sends address of destination and next hop while intermediate node has to calculate the route from its routing table. Second method is called probabilistic forwarding where upon receiving a packet the node looks up the probability table for the desired destination and forwards the packet to the neighbor with the highest probability

(b) *Self routing* where there are two implementations:

   i. Tree based forwarding where a node specifies link state information to all destinations and send an update to its neighbors upon occurrence of changes (link breakage). In this case a source will be a root in a tree and leaves are the destinations. The source will have the task to specify the forwarding path. Description of the whole path will be included in packet header and intermediate nodes will not have do any calculations (this method is also called source routing). The disadvantage of the hop-by-hop routing over source routing is that each intermediate node has to store and maintain routing information for each active route and may require sending periodic beaconing messages to its neighbors to be aware of its neighborhood.

   ii. Second approach is called position based forwarding where the forwarding decision primarily depends on the position of the packets destination and the position of the immediate one-hop neighbors of the node. In this approach each node may determine its location using one of these methods : GPS, self positioning algorithm (SPA) or time of arrival (TOA). Location service is used by sender to obtain location of destination node and add this information to packet header, so that routing decision is made according to position of destination node and its neighbors. Examples for this kind are The Location-Aided Routing (LAR) , Anchored Geodesic Packet Forwarding (AGPF) and Position Based Ant Colony Routing (POSANT).

(c) *Data broadcast and neighbor filtering* where a node advertises its cost for delivering a message to the destination, and only those neighboring nodes that can deliver the message at a lower cost relay the message. Gradient routing (GRAd) is an example of this forwarding approach which shows low end-to-end packet delays and offers good immunity to rapidly changing topologies.

(d) *Flooding* where the node broadcasts the packets to all neighboring nodes, this considers a robust approach but has very low efficiency.

Table 2.3 presents some examples for different routing protocols and their components

Table 2.3: Example of routing protocol components

Routing component	OLSR	DSR	AODV
Route discovery	Proactive(efficient flooding by means of MPR)	Reactive(flooding with loop prevention technique)	Reactive(flooding with loop prevention technique)
Route selection	Link state	Proactive	Link state
Route maintenance	Periodic HELLO messages	ACK based with no route refreshing mechanism	Periodic HELLO messages
Route representation	Routing table	Complete path presented on packet header	Routing table
Data forwarding	Table based	Self routing	Table based

2.1.2 Auxiliary components

Now after introducing core components we would like to introduce auxiliary ones.As mentioned before, the purpose of auxiliary components is to enhance the performance of routing protocol in a willing to adapt for certain services and applications. Also it would be possible to add some desired features which were not exist before for a purpose of coping with quick technology development. Bellow we mention some of the auxiliary components:

1. **Hierarchical structure** generally speaking the core of any network is formed by physical topology (actual positioning of nodes) and logical structure which in turn is formed by routing protocol. Further we can categorize networking approaches in two types, one is called flat (zero-tier) infrastructure where all nodes have the same role from routing point of view. The other approach is called hierarchical infrastructure (N-tier) where there are nodes having different role than the others and network is divided into clusters. In principal it is not mandatory that logical and physical topology will directly correspond to each others as logically hierarchical routing can be implemented in physically flat topology and vice versa.

2. **Security** [16] is an important issue for ad-hoc networks for some reasons:

   (a) *The wireless network is more susceptible to attacks ranging from passive eavesdropping to active interfering secondly*

   (b) *Lack of an on-line certificate authority (CA) or Trusted Third Party adds the difficulty to establish security mechanisms*

   (c) *Mobile devices tend to have limited power consumption and computation capabilities which makes it more vulnerable to Denial of Service attacks and incapable to execute computation-heavy algorithms like public key algorithms.*

   To enable security in ad hoc network, we must consider the following security attributes: availability, confidentiality, integrity, authentication, and non-repudiation. Generally speaking, security can be maintained either in link layer (encryption for example) or network layer (using IPSec for addressing). Routing protocol should be protected against any attack on any kind of previously

mentioned attributes as we will explain. Concerning authenticity and integrity, if public key is used then both of them would be handled in parallel where digital signature is applied for both confirming the origin of the data and its integrity. Also maintaining authenticity of the routing data is very important because nodes can confirm the source of new or changed routing information; while without integrity protection the attacker is able to destroy messages, manipulate packet headers or even generate false traffic and as a consequence the actions cannot be distinguished from hardware or network failures. In the other hand non-repudiation is somehow related to authenticity, so that routing traffic must leave traces to ensure that any party sending routing information cannot later deny of having propagated the data to other parts of the network. At the end routing protocol must maintain availability because without it part of the network will not be connected and out of any service.

3. **Multi casting** which involves the simultaneous or near-simultaneous transmission of data from one source to many destination nodes or from many sources to many destination nodes. Here we present reliable multicast approaches [13] for MANET and they mainly subdivided in two types:

   (a) *Deterministic approach*: it provides all-or-nothing delivery guarantees for the deliver of messages to a group of nodes in a MANET where it tries to detect and repair failures, either at the source, or locally where the route is broken. As an example we mention Reliable Multicast Protocol (RMA) where link life time (predicted future life of a link) is used as a metric in route creation

   (b) *Probabilistic approach*: it provides predictable probabilistic guarantees with bounded termination time and with relatively low packet overheads. Gossip concept is used where nodes outside the normal message delivery phase exchange information on which messages they have received, thus increasing the reliability of the system. Comparing it with deterministic approach we can say that probabilistic one has fewer guaranties of packet delivery but from other side less restrictive assumptions and constraints associated with it as well as reduced overhead. Examples are Anonymous Gossip (AG) and Route Driven Gossip (RDG) protocols.

4. **Broadcasting** which is considered as a variant of multicasting protocols where the multicasting group contains all participant nodes in the network (source node is excluded)

5. **Neighbor discovery and maintenance** which maintains dynamically neighborhood information such as location, direction, ID, resources, etc

2.1.3 Routing protocol design considerations

After introducing component based routing protocol (CBR) concept, we mention in this section some considerations which must be taken into account while designing any ad-hoc routing protocol [10,12]. We can think of them as challenges where they vary

according to network topology, time, wireless technology, provided services, etc.They are introduced as follow:

1. **The scalability of the protocol with network size**: as number of node grows, the amount of information required for route discovery and maintenance increases exponentially.Making optimal use of network bandwidth (which will be used for transmitting data as well as exchange routing information) will be a challenge.

2. **Frequent topological changes**: nodes may enter or leave the network randomly, when current rout is unusable a new one must be established. This process should happen so quickly to prevent loss of data. Handover between two access points (or gateways) mechanism could be invoked when changing routes and data transmission is in progress.

3. **Unidirectional links**: ad-hoc network consists of different types of devices, some of them with different wireless interfaces, even the devices with the same interfaces could have different ranges due to power constrains. That makes the channel between two nodes sometimes asymmetric. If node A can transmit to node B, that doesn't mean by default that opposite scenario can happen.

4. **Power supply**: as all mobile nodes are battery driven, power conservation should be considered in addition to load balancing and transmission efficiency. Also routing protocol should accommodate sleeping mode (idle) without overly adverse consequences. That may require a close coupling with link layer protocol through integrated interface

5. **Quality of service (QoS) support**: topology of network change frequently and this offers a challenge to maintain QoS. As each time a path will break, routing protocol must find alternative path with same QoS requirements.

6. **Loop freedom**: in some cases some packets will be routing in infinite loops, which will consume links resources and cause remarkably end-to-end delay. Some solutions exist like Time to live (TTL) which can bound this problem, but other well structured approaches are desirable and often times lead to better overall performance.

We consider previously mentioned discussion as a guidance which could help us to find an optimal routing solution for our desired application. But how optimal is this solution depends on routing performance assessment metrics we use. As we like to be able to assess the protocol after design and see if it is really suitable for current application or not. Also we can determine the weak-point of specific implementation and try to suggest improvements.

2.1.4 Summary

In this part we have introduced a survey on ad-hoc routing protocols design components and we gave some examples. We have introduced the concept of component based protocol which gives flexibility in designing routing protocols. By flexibility we mean

the ease of protocol modification to suit some conditions. For our case this could help to design dependable components for PN routing protocol. Also we have introduced possible design for route maintenance mechanism base on the use of multiple paths. In later part we will introduce a survey on multi-path routing techniques. As we will see how these techniques can enhance the performance of routing protocols in a great deal. In that part also we will show that for different protocols there also exist many kinds of axillary components which are implemented together with multi-path method to boost the performance of routing protocols. But first we will start from next part to investigate the performance of single-path ad-hoc routing protocols.

2.2 Single-path routing protocols

Mobile ad-hoc networks are wireless self configuring networks that dont depend on fixed infrastructure to communicate where nodes work as routers to route packets for other nodes. Topology of network is varying dynamically which introduce a challenge for routing protocols.In this paper we study the performance of different routing protocols which are designed for MANETs. There are many performance comparisons are made to evaluate performance for different kind of routing protocols. This has been done under varied network conditions (size of surface area where nodes are moving, maximum speed, number of nodes). In some cases it was also possible to simulate different MAC layer protocols (RTS/CTS CSMA/CA) to study the layer interaction effect between network and link layer during different load and mobility conditions. Mostly, UDP protocol was preferred on TCP in simulations because the later offers a confirmation load to the network. That means, due to TCP mechanism a node will send a packet when it receives a confirmation of no network congestion. For some mobile scenarios, this might result in variation of time the packet will be sent and also variation of source location for each protocol under comparison. This in turn could lead to indirect comparison. But in later chapter we will show that TCP protocol in general will show better performance than UDP protocol for client/server kind of traffic. In this part, most of studied simulations were using random way point mobility model where a node starts to move twoards a randomly chosen destination with uniformly distributed velocity between (0 and maximum velocity) , upon reaching destination the node stops for a particular period of time (pause time) before continue to move again to other randomly chosen destination. In section bellow we define protocols under comparison and the metrics used for measuring the performance.

2.2.1 MANET routing protocols

A mobile ad hoc networking (MANET) working group has been formed within the Internet Engineering Task Force (IETF) to develop a routing framework for IP-based protocols in ad hoc networks. There are three main types [22]of routing protocols categorized on the route discovery mechanism they apply:

1. *Proactive protocols*: nodes maintain a route to all other nodes no matter if there is a packet to send to some of them or not, further they react to topology changes

even if there is no traffic affected by the change. They require periodic control messages to update freshness of routes.

2. *Reactive protocols*: where route discovery is initiated only on demand when there is a packet to send.

3. *Hybrid protocols*: It was found that there is no optimal protocol suits all networks, but it is somewhere between proactive and reactive one. Hybrid protocols try to combine advantages of both proactive and reactive approaches.

Our goal is to study differences in performance of different protocols which are based on simulation experiments. We will show how the performance of routing protocols could vary under different scenario conditions (mobility pause time, number of nodes, burst time). First we start by defining our comparison metrics which we use to measure the performance of routing protocols as described bellow:

- **Routing packet overhead**: the total number of control packets transmitted during simulation. For packets sent over multiple hops we consider each transmission of packet from each hop as one transmission. This metric measures the scalability of routing protocol and the amount of congestion it will generate in low bandwidth environment and power consumption efficiency.

- **Routing byte overhead**: it presents the sum of all bytes used to control traffic including both bytes in control packets and the bytes in data packets (source routing scheme) which is stemming from IP layer

- **Average delay**: it indicates in average the time taken by a packet generated at application layer of source node to arrive at the application layer of destination node. We call it also average end-to-end application delay.

- **Throughput**: the throughput is defined as the total amount of data a receiver R actually receives from the sender S divided by the time it takes for R to get the last packet. This metric was replaced in some simulations by other one called packet delivery ratio. Packet delivery ratio (PDR) measures the efficiency of a routing protocol.

- **Packet delivery ratio(PDR)**: the difference between the number of packets originated by the application layer in source node and number of packets received by destination. This metric characterizes the completeness and correctness of a routing protocol.

As we saw in previous section, hybrid routing protocols divide the network into clusters with cluster-head nodes working as bridge to transfer packets between clusters. In most of cases, a proactive strategy is implemented for intra-routing while reactive one for inter-routing. In [21,24] it has been shown that Hybrid protocols have less WCC(worst case computational complexity, which is the number of messages needed to perform route discovery or update operation in worst case) and also less WTC (worst case time complexity ,which is number of steps involved to perform a route discovery or update operation in worst case) than reactive protocols.This has also been proofed

via simulations [22,23] as hybrid protocols (DZTR,ZRP) have less overhead ($< WCC$) and delay ($< WTC$) than reactive one(AODV) in stressful scenarios (larger number of nodes with more flows). But from other side, reactive protocols have performed better in packet delivery ratio in low to moderate mobility than hybrid protocols with respect to overhead. For proactive routing protocols, they have performed poorly in terms of packet delivery ratio in all of simulations comparing to other two kinds. As they couldnt react fast enough to update routing tables in high mobility scenario and they produced more overhead than reactive and hybrid protocols. In terms of delay, proactive protocols perform well as they send the packet via an established route and there is no extra time wasted for route discovery process as in reactive protocols. In following sections we will introduce simulated scenarios, where each time a factor is changed to see its response on performance metrics.

2.2.2 Varying pause time (mobility)

As we mentioned before, pause time in random way point simulation is the time a node has to wait before moving again to next way point twoards final destination. This metric shows the effect of varying mobility in the network which is an important factor in mobile ad hoc networks as topology is dynamically varying all the time. Bellow we show the effect on performance metrics:

1. **Overhead**:
   (a) *Proactive protocols (DSDV, OLSR)* have nearly constant overhead in low mobility because of their proactive nature, but in high mobility where more routes will have to be fully updated they will have the highest overhead.
   (b) *Reactive protocols (AODV, DSR, TORA)* [26] have increasing overhead with increased mobility because they will have to react on more link breakage and as a result number of control packets will increase. DSR outperforms AODV in terms of packet overhead due to aggressive cashing property as it maintain multiple routes to destination in the cache, while AODV uses periodic HELLO messages to recognize neighbors. Also DSR uses promiscuously overhearing which will allow him to get information from one route discovery in subsequent route discovery and it uses also aggressive non-propagating route request mechanism which will limit flooding process. In total DSR will have significantly less packet overhead than AODV. But in terms of byte overhead, DSR has higher overhead than AODV because of source routing. It was also shown that DSR generates less RREQ packets (broadcast) up to order of magnitude than AODV does, and more (2-4 times) RREP /RERR packets (uni-cast). That means if MAC layer overhead is factored in, DSR will generate more overhead than AODV in high mobility scenario, as the cost to acquire a medium to transmit a packet is higher than the incremental cost of adding few bytes in an existing packet in terms of power and network utilization. TORA showed the worst performance as the overhead in this protocol is sum of constant mobility-independent overhead caused by periodic HELLO messages and variable mobility-dependent overhead caused

by routing packets used for creating and maintaining routes multiplied by number of retransmission and ACK packets IMEP uses reliable and in-order delivery of packets.

(c) *Hybrid protocols* (ZRP, DZTR) [22,23] have the best performance in overhead metric over both previously mentioned protocols. For high number of nodes DZTR outperforms AODV in high mobility (8 times less overhead), but in low rate of mobility they have close performance. ZRP has more overhead in low mobility than reactive protocols due to proactive intra-routing;but in high mobility scenario it was outperformed by DSR.

2. **Packet delivery ratio**:

(a) *Proactive protocols* (OLSR, DSDV) have worst performance in medium to high mobility, OLSR will have to update entire set of multi point relay MPR and generate more overhead which will lead to network congestion and packet loss, while DSDR will not be able to respond on time to topology changes.

(b) *In the case of reactive (AODV, DSR, TORA) and hybrid (DZTR, ZRP)*[23,29] protocols, we saw that DSR outperforms AODV and TORA because of multiple route cashing while AODV stores only the best path. TORA performs well in low to moderate mobility, while packets are dropped because of short lived routing loops which are created in link reversal process. Further more AODV has better performance in low mobility than ZRP which will have better performance in higher mobility. Also AODV will have better performance than DZTR at medium number of nodes (100 nodes) but when number of nodes become higher (200 nodes) DZTR will have slight better performance in high mobility where channel contention will be lower because the utilization of different location tracking mechanisms, and as a result it will have less queuing times and packet losses.

3. **Average delay**:

(a) *Proactive protocols* (OLSR, DSDV) have the least delay as they use routes already in the routing table.

(b) *For reactive(AODV, DSR, TORA) and Hybrid protocols (ZRP, DZTR)* DZTR shows the worst performance with low number of nodes (100 nodes), but best performance when number of nodes increased (200 nodes). That is because DZTR uses iterations to invoke number of location tracking strategies to find the route , so in dense network the probability to find a route during first iteration is high. In other hand ZRP shows less delay than reactive protocols in high mobility according to proactive part, but AODV [31] outperforms in low to moderate mobility. Also AODV has less delay than DSR because the later might use stale routes which will cause retransmissions. Other problem in DSR mechanism is that intermediate node could replay with invalid route (stale) when it receives a route request message which will cause a pollution of some other caches.

2.2.3 Varying number of nodes

This metric measures the scalability of a routing protocol as it has influence on (hop count, path length, convergence) of routing algorithm. In our case for PN, clusters could merge or split dynamically. Bellow we introduce different protocols behavior with respect to previously mentioned metrics.

1. **Overhead**: for all protocols overhead increases with increasing number of nodes. In proactive protocols more routes will be maintained and as a result there will be higher number of periodically update messages and larger routing tables (OLSR will limit flooding by using MPRs), while in proactive protocols number of RREQ and RREP will increase. It was mentioned in [25] that asymptotic overhead for proactive, reactive and hybrid protocols are $o\left(N^{1.5}\right), o\left(N^{2}\right)ando\left(N^{1.66}\right)$ respectively, where N = number of nodes. Hybrid /hierarchical protocols will have more flexibility (less delay, more delivery ratio) in high dense networks because the overhead produced for discovering intra-cluster routes and inter-cluster routes is restricted inside the cluster instead of the whole network as the case of other protocols. Reactive protocols outperform in medium dense scenario [25], where DSR will have zero overhead in stationary scenario.

2. **Packet delivery ratio**: increasing number of nodes decreases PDR for all protocols because longer and fragile routes will be established which will lead to packet losses. Hybrid protocols like DZTR will outperform in dense networks because it utilizes different location tracking techniques and in turn fewer data packets will be dropped. At next step come proactive protocols as they perform better than proactive protocols under same scenario. DSR will perform better than AODV because of aggressive caching.

3. **Average delay**: proactive protocols have the best performance with respect to delay metric among all protocols because data packets will be sent via already maintained routes. Hybrid protocols will outperform reactive ones in large scale networks because they introduce less overhead which will lead to less network congestion. Among reactive protocols AODV will have better performance than DSR because of choosing best path criteria, while in DSR the first path will be chosen according to first RREP received.

2.2.4 Varying number of source nodes(applied load)

This metric examines different protocols when the load on network increases. It was shown in [20] that under high stressful situation (heavy data load), the delivered throughput to application was 2 to 5 percent of total network capacity and that was due to significant B.W is consumed by MAC control packets (RTS/CTS/ACK) and additional B.W is consumed by dropped packets, as their B.W cost didn't pay off.We will reserve talking about MAC layer to later chapter where we will introduce the effect of MAC layer on dependability. Bellow we will mention the effect of varying load on the performance:

1. **Overhead**

(a) *Hybrid protocols* will have the best performance as they implement proactive mechanism in intra-routing and reactive mechanism for inter-routing which will deliver best overall performance (the cluster maintenance is fixed portion of routing overhead, with increasing number of connections the overall overhead becomes relatively smaller).

(b) *For proactive protocols* the overhead for constant number of nodes will be constant and independent of number of source according to proactive nature, by increasing data rate congestion will occur which will lead to retransmissions and in some cases initiate alternative routes.

(c) *In reactive protocols* overhead will increase as number of source nodes increases, by increasing data rate congestion will happen and DSR will outperform AODV because of caching while AODV will initiate route discovery process.

2. **Packet delivery ratio**

(a) *For all protocols* PDR decreases when network load increases because of network congestion and nodes will suffer from queue overflow which in turn will cause packet loss, more retransmissions, and more congestion.

(b) *It has been shown* in [26,28] that hybrid protocols have the best performance in stressful situations because less overhead they produce which will lead to less network congestion and packet loss, where AODV outperforms DSR because the absence of route refreshing mechanism in DSR which will result in stale routes (packet sent via these stale routes will be lost) .

3. **Average delay**

(a) *At high traffic load* all protocols will have higher delay at low mobility thats because routes live longer and more traffic will be maintained over the same path during longer time, which will cause longer queues formation and as a result higher delay. When mobility increases, routes will live shorter time and establishment of fresher route will occur more often so as a result the traffic will spread over larger number of routes (We could think of it as load balancing.)

(b) *Hybrid protocols* have slightly less delay than AODV[28] when average connections per node exceeds 0.17 for 200 node network because of less overhead produced, while AODV has very good performance with small curve slope when number of connections increases.

2.2.5 Summary

In this section we have introduced a simulation-based comparison between MANET routing protocols. Performance comparison metrics were packet delivery ratio, average delay and routing overhead. We conclude that there is no global protocol will have optimal performance with all different scenarios, but according to network characteristics

Table 2.4: Performance of routing protocols under different scenarios

	Packet delivery ratio (PDR)	**Average delay**	**Routing overhead**
Varying pause time (mobility)	- Reactive protocols have the best performance, while proactive protocols have the worst. - Among reactive protocols we found DSR outperforms AODV and TORA in low mobility because of caching multiple routes - In higher mobility scenario AODV will have better performance.	- Proactive protocols have the best performance in higher mobility, on second degree come reactive protocols - AODV has significantly better performance than DSR in higher mobility , while in lower mobility the difference become smaller	- Hybrid protocols have less overhead than both reactive and proactive ones in high mobility , and the difference become smaller in lower mobility - proactive protocols(OLSR) have better performance than reactive protocols (AODV,DSR)except for fixed scenario - DSR has less overhead than AODV in terms of frequency of route discovery
Varying number of nodes (scalability)	- For small to moderate number of nodes(50 nodes) reactive protocols have better performance where DSR is slightly better than AODV - For dense networks Hybrid protocols performs better (for 100 node scenario DZTR outperformed AODV)	- Proactive protocols have the best performance for small to moderate number of nodes - AODV has less delay than DSR because the optimal path is chosen - Hybrid protocols have better performance than reactive protocols in dense networks	- With increasing number of nodes difference in performance becomes higher and Hybrid protocols have the best performance -reactive protocols have better performance with increasing number of nodes than proactive protocols - DSR has less overhead than AODV
Varying number of sources (offered load)	- Hybrid protocols have best performance in more stressful situations due to less overhead - AODV has better performance than DSR when number of flows increases, as in stressful situations DSR faces stale routes problems - DSDV has better performance than AODV in low mobility with increasing number of sources, while in higher mobility AODV wins	- Hybrid protocols have less delay than reactive protocols when number of flows is high - OLSR has better performance than reactive protocols with increasing data load - AODV outperforms DSR and DSDV when number of flows increases, while in less stressful situations DSR will have slightly better performance.	- Hybrid protocols have best performance with increasing number of flows - proactive protocols have less overhead than reactive ones with increasing number of sources - DSR has lower overhead than AODV always

(mobility, number of nodes, traffic load) a suitable routing protocol could be designed for supposed scenario. Table 2.4 summarizes the results in a table which contains performance parameters horizontally and varying scenarios vertically. After assuming PN conditions we can narrow the list of possible routing protocols which will have good performance with our personal network in Table 2.5 . As we can see form table above that according to assumed PN scenario conditions we have three potentially suggested protocols (OLSR, AODV, and DSR). We would like to remind that above mentioned simulation comparisons were based on some conditions which could not fit with our PN case(large network service area, number of nodes, traffic,etc.) As these comparisons

Table 2.5: Performance of routing protocols under PN scenario

Assumed PN scenario condition	PDR	delay	overhead
Low to moderate mobility: typical mobility pattern for persons moving within the PN cluster (office, home)	reactive (DSR)	reactive(AODV)	proactive(OLSR)
Low to moderate number of nodes (max. 100 nodes)	reactive(DSR)	proactive(OLSR)	reactive(DSR)
Low to moderate number of source nodes	reactive(AODV)	proactive(OLSR)	proactive(OLSR)

were carried out to investigate the behavior of ad-hoc routing protocols in general case. Thus further investigation must be done considering our PN scenario. This exactly is the work which we will introduce in Chapter 3, as we will simulate PN scenario using OPNET 14.5 simulation environment. Next part we will introduce different multi-path routing techniques which as we will see could improve the reliability of routing protocols.

2.3 Multi-path routing protocols

Multipath routing technique is used to exploit available network resources by utilizing multiple paths from source to destination. There are many benefits of using this technique like enhancing data transmission reliability, minimizing end-to-end delay, increasing fault tolerance, B.W aggregation and load balancing [33]. The idea of multipath routing was applied for first time on traditional circuit switching network where an alternate path route was used when the primary shortest path fails or congested. Thus by using multipath routing the probability of call blocking was minimized. Later the application of multipath routing was extended to include data networks as well. For example, ATM PNNI standard [35] supports multipath routing in so called the crank back and alternate routing mechanism. This process starts upon a call failure on the main route and tries to send the traffic via alternate path to provide fault tolerance. As another example is the fault tolerance in the biggest data network (Internet) where there are some routing protocols like OSPF and RIP are implemented which both support multipath mechanism. In this paper we concern about mobile ad hoc networks (MANETs), which forms a flexible non structural network containing variety of mobile hosts with possibly different capabilities and movement patterns. To be able to transfer data among mobile nodes, a reliable routing protocol has to be designed which will be able to adapt to dynamical topology and transmission condition changes. Routing protocols for MANETs [34] were designed to achieve some goals including minimal control overhead, minimal processing overhead, dynamic topology maintenance and loop prevention. Most used protocols for MANNET were single-path, where the protocol tries to find single optimal path from source to destination satisfying some performance metrics. Further on there were many strategies developed to

enhance the performance of these single path protocols [37, 38, 39]. Recently it was shown that multipath routing protocols which provide multiple routes from source to destination are able to compensate for dynamic and unpredictable nature of MANET. It was shown that using multipath routing in ad hoc networks of high density results in better throughput than single-path routing [34]. A lot of multipath routing protocols have been proposed for MANET where many of them are based on the famous on-demand routing protocols DSR and AODV [33]. These protocols inherit their characteristics from DSR and AODV when we compare their performance together. We can say that DSR-based protocols have the advantage of simplicity where there is no need to maintain routing tables in forwarding the information like the case in AODV-based protocols. However the DSR-based protocols produce more overhead because of using source routing mechanism where the entire route is specified in the data header. Also there are some multi-path protocols built on OLSR proactive protocol [34,35] which show better performance than default OLSR with respect to PDR. Further on we also like to mention that multipath routing technique has drawbacks compared to single-path, which are complexity and overhead [36]. For example, in AODVM protocol larger routing tables have to be maintained in the intermediate nodes. That was not the only problem but also traffic allocation mechanism used in multipath routing can result in packet reordering problem, where in single-path routing this is not an issue. The allocation granularity scheme is defined as the mechanism determines the smallest unit of information to be sent along each path. There are two types of allocation granularity which are: per-connection granularity which allocates all traffic for one connection to one path and per-packet granularity which divide the traffic into packets and distribute them amongst multiple paths. It was shown that a per-packet granularity has better performance than per-connection granularity [40]. In the following section we discuss the components of multipath routing.

2.3.1 Components of multipath routing protocols

In general the multipath routing protocol consists of three main components: route discovery, route maintenance and traffic allocation. Bellow we discuss each of these components in details.

1. **Route discovery**: it is defined as the process of determining the available paths from a source to a destination node [33, 36]. There are different criteria that a multipath routing protocol should follow to choose the set of multiple paths in route discovery phase. According to our knowledge we mention here two criteria which are path disjointness and transmission independence as they will be explained bellow:

   (a) *Path disjointness*: there are three types of path disjointness which are

   - Node/ totally disjoint paths where paths dont have any nodes or links in common
   - Link-disjoint paths where there is no common links between paths but possibly some common nodes.

- Non-disjoint paths where there could be common links and nodes exists between both paths

The advantage of disjoint paths is that they offer resource aggregation because the use of different network resources at the same time. We can see that node-disjoint paths offer the most aggregation of network resources because paths share neither nodes nor links in between. Also disjoint routes offer fault tolerance because a fault in a link or a node will cause only a single path to fall. In the other hand non-disjoint paths have the advantage that they are easier to be discovered because there are less restrictions on path selection. It was shown that for route discovery algorithm it is difficult to find node disjoint paths especially in case of sparse networks [43] , even in moderate dense networks the number of node disjoint paths will be small. In some cases multipath routing protocol may inevitably use longer paths because there is no shortest node-disjoint paths exist, which will waste more B.W and increase end-to-end delay. Other problem may also arise as a result of length difference between shortest and alternative node-disjoint paths which is the need of more buffer space in destination node to handle disordered packets. Link disjoint paths provide a suitable compromise. In some protocols they use the notion of maximally disjoint paths which defines the paths with minimum link or node joint. We can mention some example protocols which use disjoint paths strategy [36]:

- Split multipath routing (SMR) is similar to DSR and use the metric of maximally disjointness, and select only two maximally disjoint routes.
- Ad-hoc on-demand multipath distance vector routing protocol (AOMDV) which has the ability to find either node-disjoint or link disjoint paths
- source routing based multi-path OLSR (SR-MPOLSR) [68] where MPRs are used to define network topology effectively. And Dijkstra algorithm is used to calculate multiple routes and allocate the loads in a wighted round robin fashion.

(b) *Transmission independence*: communication among nodes in MANET happens under the control of MAC layer. That has arisen a cross layer issue which must be taken into account in designing a multipath routing protocol. When nodes in different paths are located within the transmission range of each others and they transmit data simultaneously, a collision will happen and some of them will have to postpone their transmission. These nodes are said to be in the same collision domain. This scenario could happen when traffic is split over the multi paths simultaneously. As a conclusion we say, while node-disjoint paths provide failure independent paths but they will not be able to ensure transmission independence. Further more there exist some metrics which are used to calculate the relative degree of independence among paths which are:

- Correlation factor of two node-disjoint paths η [44] which is the number of the link connecting two paths. Two paths are uncorrelated if there is no links connecting them. We define the total correlation factor of a

set of multi paths as the sum of correlation factors between each pair of paths. It was shown that the larger the correlation factor between two paths the larger the end-to-end delay will be in both paths.

- Route coupling [45] between two routes is defined as the average number of nodes that are blocked from receiving data along one of the paths when a node in the other path is transmitting. The advantage of using this metric is that it is applicable for both disjoint and non-disjoint paths. In multi-channel networks coupling happens between paths which share a common intermediate node (reserved channel) , while in single channel networks coupling is more serious (MAC layer collision). It was shown that alternate path routing provide 20 percent reduction in end-to-end delay for bursty data streams in multi-channel networks, while it offers small improvement in QoS for single channel networks [45]. So, multi-channel multi-user scheme could be considered as a solution to prevent route coupling in multipath routing protocols.

As a conclusion we can say that choosing paths with low correlation factor and/or implementing multi-channel-multi-user scheme can improve the performance of multipath routing protocol.

2. **Route maintenance**: in mobile ad hoc networks wireless links between mobile nodes are borne to breakage, even nodes themselves could come down due to battery failure or any other malfunctions. Route maintenance is the process of regenerating paths after the initial path discovery process has finished. Depending on the multipath routing protocol, a new route discovery will be initiated upon the failure of one route or the failure of all available routes. In [48] a performance comparison between both strategies has been introduced in terms of frequency of route discovery (which considers as main source of overhead for on-demand routing protocols). It was shown that supplying the intermediate node with alternate route to the destination will decrease the overall routing overhead. But as a drawback, waiting for all routes to fail will cause a delay before a new route will be available and this may degrade QoS of the application. In the other hand triggering route discovery process each time a route fails will cause more overhead [36]. A good compromise could be performing route discovery when N routes fail and N is less than the number of available paths. In some multipath protocols a dynamic maintenance algorithm is used to constantly monitor and maintain the QoS metric for available paths.

3. **Traffic distribution**: there are different strategies for traffic allocation in multipath routing protocols [33]. Some protocols forward the traffic via the path with best metric and keep other alternative paths as backup in case of failure of the main path. A second strategy is to use single path at a time in round-robin fashion. It was shown that employing alternative paths only when the primary path fails/overloaded or when the queue length exceeds a certain threshold is reactive process in nature and prone to oscillations [47]. This oscillation happens because the main congested path (with optimal metric) will be under-utilized after a while from switching the traffic to alternative path. Other mechanism for multipath pro-

tocols is to use all multiple paths concurrently. One of the expected advantages of this method is distributing the traffic efficiently among multiple paths to achieve load balancing as it was the case for wired networks. But in [46] it was shown that if the node density is not high or the path number is small the load balancing will not be remarkable. To make a conclusion we can say that there are mainly two kinds of traffic allocation [47] for multipath routing protocols:

(a) *Per connection allocation* where all packets from one connection follow the same path. The drawback of this method arises when there will be several connections as it will be difficult to ensure uniform traffic distribution over multi paths since connections could vary widely in their rates of flow. Accordingly the efficiency of network resources utilization depends on the relative duration of the connection.

(b) *Per packet allocation* where packets from a single connections take several paths. It has been shown that [47] per packet allocation is more reliable than per connection allocation, as it adapt faster to traffic variations and failures. A drawback of this method is the need of addition reassembly process by the destination when packets arrive out of order. However, the delay for reconstructing the message at destination depends heavily on the elapsed time between receiving the first and last packet of the connection. Further more it was shown that in-order arrival of packets doesnt ensure minimization of reconstruction delay [47]. The main source of reconstruction delay has shown to be variable sizes of messages and mismatches in link capacities.

In case of path failure, per connection allocation responds by either switching the traffic to alternative path or by aborting the connection and waits till a new route is discovered which in both cases will cause service degradation. In the other hand per packet allocation adjusts the fraction of traffic routed among multiple paths such a non-reliable path is bypassed completely. Other strategy which per packet granularity follows is to use redundant information (diversity coding) which will be described later. Thus per packet allocation is more robust scheme than per connection allocation because less traffic will be affected by path failure, and practically it has shown efficiency in bursty networks like ATM. Till now we mentioned multipath routing protocol components, and we introduced different implementations and a comparison between these implementations (pros and cons). Next part, we will discuss the benefits of using multipath routing protocol and we will give some implementation examples to show how these benefits could be achieved. Further on last part we will introduce some limitations on using multi-path techniques.

2.3.2 Advantages of multipath routing

As we previously mentioned in the introduction part, multipath routing protocol provides a range of benefits like increasing data transmission reliability (fault tolerance) , minimization of end-to-end delay, load balancing, congestion avoidance, decreasing overall routing overhead and bandwidth aggregation. In this section we will show dif-

ferent routing techniques used to boost the performance in a specific way. Bellow we mention those multi-path techniques and the enhancement in the performance they could achieve :

1. **Increasing reliability** multipath routing protocols can increase reliability by providing a fault tolerance capability where redundant information is routed to the destination via alternative paths. This technique will reduce the probability of communication disruption in case of link failure. There are different kinds of multipath routing strategies to maintain reliability of data transmission and we will mention some of them bellow:

   (a) *End-to-end reliability*: one of the protocols that provide QoS in terms of end-to-end reliability is the multipath dynamic source routing protocol (MP-DSR) [42]. End-to-end reliability is defined as the probability of sending data successfully within a time window, and it can be calculated as a product of availabilities of the links constituting the path. Link availability is defined as the probability that there is an active link between two nodes at time $T = t0 + t$, where $t > 0$ given that there was an active link between them at time $T = t0$ [41]. One way to calculate link availability is from nodes movement model. A mathematical expression for end-to-end reliability is defined as $P(t) = 1 - \prod_{k \in K}(1 - p(k, t))$, where K is the set of node-disjoint paths and $p(k, t)$ is the availability of link k. So we can see that $P(t)$ is the probability that at least one path stay connected for duration of t. By this way, data transmission fails if and only if all disjoint paths fail at the same time, thus the probability that transmission fails is less than probability that any path fails individually. We can demonstrate that by showing the same example mentioned in [36]. Probability of link are calculated as $PPSXD = 0.6 \times 0.8 = 0.48$,$PSYD = 0.7 \times 0.6 = 0.42$,$PSXD = 0.6 \times 0.5 = 0.30$, then End-to-end reliability $= 1 - (0.52 \times 0.58 \times 0.7) = 0.78$. In (figure 2.4) link availabilities are written on each link. On the left hand side path reliabilities are calculated for alternative paths and total end-to-end reliability is also calculated. We can see that end-to-end reliability is higher than individual path reliabilities which achieve our goal. Guarantee of QoS with respect to end-to-end reliability is provided during route discovery phase, where the application supplies the protocol with end-to-end reliability requirement Pu. Having this metric as input, the protocol determines two parameters: the number of disjoint paths it needs to discover (m0) and the minimum link availability τ requirement that each search must maintain to be able to satisfy the required Pu for each single discovered path. These two parameters are inversely proportional, for lower m0 we need higher τ and vice versa. In some scenarios less number of paths will be needed (good utilization of network resources, less maintenance overhead, avoid route coupling). Further more the protocol is detached from data forwarding scheme as it can carry out any data forwarding scheme (per - packet / connection granularity).

   (b) *Packet salvaging*: the idea of caching has been introduced in 1965[M.V. Wilkes,*"Slave memories and dynamic storage allocation"*] to increase the per-

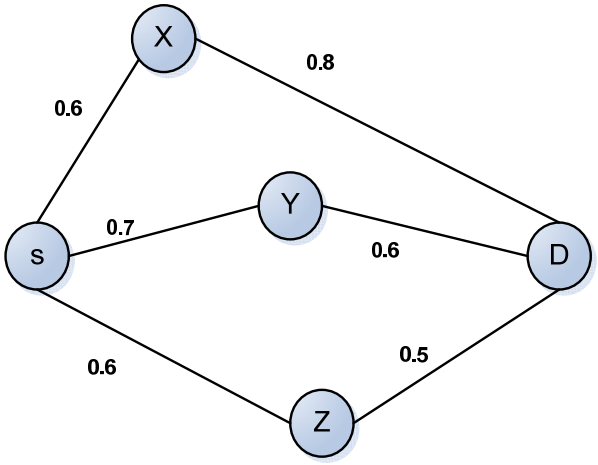


Figure 2.2: End-to-end reliability analysis

formance by introducing a cache memory as a bridge between the processor and main memory which both have different speeds [49]. A cache is defined as a small fast memory that stores the data to use it in near future in order to reduce latency and increase memory bandwidth. The cache could exploit two kinds of locality:

i. Spatial locality: where an access to a memory location indicates that a nearby location will very likely to be accessed in the near future. It has been shown that [64] in low to moderate mobile scenario the change in node position during small period of time is not large. And we can exploit this property to reduce the overhead resulting from flooding the network with RREQ packets upon route discovery. The RREQ messages are flooded only to a limited region which was previously a part of valid route instead of flooding the whole network.

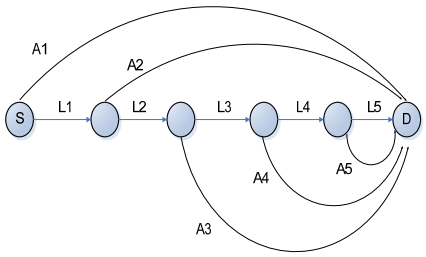
ii. Temporal locality: which states that accessing a location in the memory will give higher probability that this location will be accessed again soon.

Packet salvaging exploits the temporal locality phenomena as lost packet is probably will be the recently sent packet. Nodes keep all recently sent packets in a cache, and for optimal performance it has been shown that cache size should be not more than 5 packets [49]. Two different multipath routing protocols [48,49] present the idea of packet salvaging in different ways which will be introduced bellow:

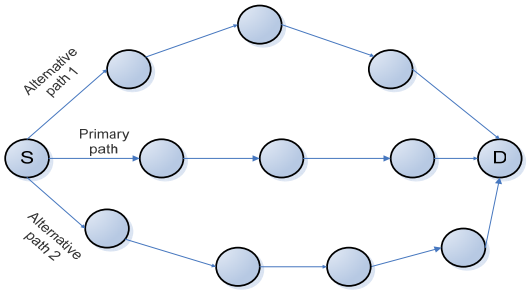
- The multipath extension of dynamic source routing protocol DSR [48]. In original DSR protocol each intermediate node cashes all possible alternative routes to destination. Upon failure in an intermediate link in the primary path, the intermediate node switches the traffic to alterna-

tive path and sends RERR packet to source node. The failed route will be deleted from upstream nodes cashes upon receiving RERR packet. The contribution in [48] is to choose link-disjoint paths from the primary path. By doing this, a link failure in one path mostly will have no effect on other paths, and also number of RREP packets will be reduced (so, less overhead). This scheme has two different variations and both define a primary route which results from the first RREQ packet reaches the destination and it should be the shortest path. After defining that primary route, the destination will be able to replay to RREQ messages for other routes which are link-disjoint to primary route. Initially the traffic will be routed via the primary route till this route fails then the traffic will be switched to alternative shortest path. Also a new route discovery will be initiated only when all available backup routes are broken. For this scheme there are two variations. In protocol variation 1, only the source node is able to choose an alternative path as it is the only node equipped with alternative paths. Upon link failure intermediate nodes send ERR packets back to source which will choose the alternative path. This variation has a drawback that there will be a temporary loss of data upon a link breakage until the source receives an error message and switches to other route. While in variation protocol 2 case, there is some technique to handle this problem by simply allow the intermediate node to have one alternative path to destination where it can switch the traffic upon the failure of proceeding link in the primary path. When a failure in a link occurs, the intermediate node consumes the error message and forward the traffic to the alternative path, subsequently only when an error message reaches the source a new route discovery process starts. The author claims that any form of multipath routing outperforms single path routing in terms of route discovery frequency. Also alternative paths should be chosen under some hop-count constraints as longer alternative routes tend to break so early, and in addition to that they will cause more end-to-end delay. A trade-off between overhead and end-to-end delay must be considered in the design and it will depend on the actual load on the network plus the application requirements. Further more it was shown that the performance advantage of using more than two alternate routes is minimal.

- Previously mentioned protocols use only one path at a time which might result in a scenario where unused routes become stale. Switching to stale routes will degrade the performance as it results in packet loss and more delay. Further more, if the intermediate node couldnt find an alive link to forward the packet it will just discard it. To overcome these drawbacks a multipath routing protocol called caching and multipath routing (CHAMP) is introduced [49]. In this protocol a round-robin per- packet traffic allocation is used to examine the freshness of routes in real time. Besides a cooperative packet caching mechanism is used to prevent losing un-forwarded packets. The previously mentioned packet salvaging



(a) protocol 1



(b) protocol 2

Figure 2.3: Different multipath techniques

scheme is used in this protocol and gives the ability for intermediate node to store more than two routes to destination. Further more, a node saves the packet in its cache and sends it through the least used route. If there is no available routes and the node is not a source node it removes the broken routes from the cache and sends back RERR message to upstream node. Some packet information will be stored on the packet header. When RERR message arrives to source node it initiates route discovery process. CHAMP protocol in this way takes advantage of temporal locality principle. Also in CHAMP protocol, routes are as much as possible of equal lengths (hop count) in order to reduce out-of-order packet problem. As we saw in this protocol, nodes relay on data link layer acknowledgment to determine the state of the link in place of classical keep alive/beacon messages used in other protocols which will reduce route maintenance overhead. It was also shown after simulation that the optimal cache size to achieve optimal performance with respect to packet delivery ratio should be maximum five packets and the optimal number of routes to each destination should be no more than two. As having two paths achieved 10 percent increase in packet delivery ratio, while adding more routes up till five delivered negligible enhancement. Also there will be 25 percent reduction by in overhead due to less route discoveries. Generally, CHAMP outperforms (AODV and DSR) protocols

in terms of packet delivery ratio and overhead in high mobility and network congestion scenarios respectively. Although it might suffer from out of order packets at receiver but it has been shown that will not degrade TCP performance [49].

(c) *Chanel coding*: this approach is used to achieve self healing and fault tolerance for wireless networks. An instantaneous recovery process is done at destination without any need for a feedback channel as in the case of other recovery techniques. Also in case of failure there is no need for rerouting which will save complexity and delay. The channel is treated as erasure channel, either data is sent (no path failure) or no data is sent at all (path failure). We concern about route failures and our protocol has to compensate failures. We mention here Dispersity routing [51] protocol as an example. This protocol has two variants:

i. Non-redundant routing where a message is sub-divided into equally length sub-messages where each of them is sent over one of available paths. Adaptive mechanism equalizes the load of network by rerouting packets through less congested routes. This scheme provides less delay in each single path because the queue will serve large number of smaller packets. But upon a failure in a path the traffic will be switched to one of alternative paths. But there will be no guarantee to reduce overall application delay because destination node will have to wait more for reception of all dropped packets. An approach to decrease delay is to use equally length paths for data forwarding which is not available all times.

ii. Redundant routing where parity check code is used to achieve erasure correction. In a simple representation where there are four paths available, data message is subdivided into three blocks of sub-messages sent over three disjoint paths. Fourth path is used to transmit parity message which consists of number of parity check bits for the three data sub-messages. All sub-messages blocks must have the same number of bits. By using this method we could recover data in case of sub-message loss due to overflow or route failure. This is possible because the protocol allows destination node to recover the message upon reception of first few segments. Notice that, this simple implementation could correct only one erasure. More sophisticated scheme could be implemented by using so called Hamming codes. A Hamming code [65] takes the form (n, k, 3) , where $n = 2^m - 1, k = n - m$ and d=3 (the hamming distance). The hamming distance indicates the ability of a code to correct or detect errors. Hamming codes have ability to detect up to 2 errors and correct single error. The factor m can control the size of sub-message blocks. For m=3, 4, 5 ,6 we have the following Hamming codes: (7,4,3), (15,11,3), (31,26,3), (63,57,3). We illustrate this idea in Table 1 bellow where each data message is subdivided into 4 equal m length sub-messages and sent over 7 different paths. The columns represent the sub messages transmitted through a single path whose number is written on the head of the column. Each row represent a code word in (7,4,3) Hamming code

where last three elements (parity sub-messages) are determined by the first four(K). Also we would like to mention another example for linear block codes which is called Reed Solomon code (RS) = (Z, K, d)[65]. Where Z is the number of simultaneously transmitted packets including number of K data packets ($Z > K$). It is also called a cyclic code over Galois field $GF(2^m)$, where $m \geq 2, Z = 2^m - 1; K = Z - 2t$. The term d (Hamming distance) is defined as $d = 2t + 1$. The factor t is number of errors a code can correct, and also it determines the efficiency of the code (number of information bits to number of codeword bits) . Advantage of RS codes that they are robust against burst errors also they are flexible and have low complexity in implementation. While the disadvantage is that RS codes are not robust against random errors where more symbols could be affected than the code can correct. Generally we can assume that route failure would cause burst errors. From previous discussion,

path 1	path 2	path 3	path 4	path 5	path 6	path7
I1	IN/4+1	IN/2+1	I3N/4+1	P5.1	P6.1	P7.1
I2	IN/4+2	IN/2+2	I3N/4+2	P5.2	P6.2	P7.2
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
IN/4	IN/2	I3N/4	IN	P5.N/4	P6.N/4	P7.N/4

Table 2.6: Implementation of (7, 4, 3) Hamming code with multi-path routing

we conclude that non-redundant protocol has an advantage over redundant protocol because extra sub-messages are not transmitted until a path failure occurs. And dispersity routing could enhance the packet loss ratio and delay performance for MANETs if multiple disjoint paths are maintained [51].

(d) *Reliability in sparse networks*: it was found that number of node-disjoint paths depends on node density in ad hoc networks. Also the more the distance between source and destination increases the less number of paths could be found between them [43]. In some situations it might be difficult to find multiple disjoint paths between two nodes because there may exist sparse areas between source and destination nodes. These areas act as bottle neck in forming reliable paths. One of the solutions is to divide the network to some segments where reliable multi paths could be found and position reliable nodes in the bottle neck areas to form reliable bridges between these segments [43]. In other words reliable nodes must be placed so that the probability of finding a reliable path between source and destination is acceptable. These reliable nodes must be more powerful than other nodes in order to adapt to topology changes in timely fashion. But unfortunately, this scheme has a

drawback because large amount of overhead must be generated to get updated topology information about the entire network. This step is done in order to extract all available node-disjoint paths between source and destination nodes. We define a reliable segment by a part of network which contains a reliable path between end nodes. So far a network might consist of several reliable segments connected by reliable nodes. We can see from Figure 2.6 an example of a network consists of 3 reliable segments with threshold $\zeta = 2$(minimum number of node disjoint paths per segment). An example of such deployment could be in a battlefield where there are number of heterogeneous nodes with different capabilities. Low powered nodes with lower capabilities like hand-helds and sensors could be deployed in the field and highly reliable nodes could be deployed on mobile vehicles.

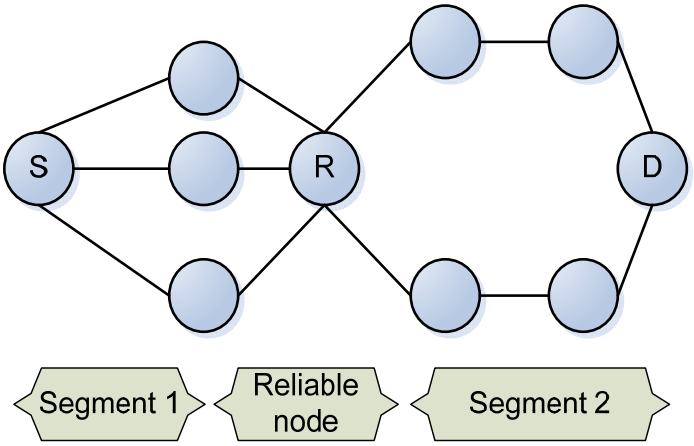


Figure 2.4: Network segmentation

(e) *Minimizing End-to-End delay*: one of the important elements that could degrade the performance of MANET is end-to-end delay. As some applications are not able to tolerate for this delay like multi-media applications. Multipath protocols could decrease this delay to an acceptable value by using appropriate techniques. It was shown in [42] that node-disjoint paths dont guarantee the optimal performance for multi-path routing protocols. Subsequently, the authors proposed zone-disjoint paths in order to avoid the effect of route coupling. Further to reduce radio interference and isolate different transmission channels a directional antenna is used. The result of their simulations show that average end-to-end delay is substantially reduced using directional antennas comparing with omni-directional antenna case. Also it was shown that length of path (more hop count) directly proportional to end-to-end delay, as longer paths even with low coupling coefficient are not efficient enough to decrease end-to-end delay and additionally more B.W eventually will be consumed. Therefore maximally zone disjoint shortest paths with directional antenna will show best performance with respect to end-to-end delay and

load balancing. Different approach is introduced in [54] called split equal cost multi-path routing (SEMR) which is a variant of split multi-path routing (SMR) and both are based on DSR protocol. This protocol is proposed to avoid congestion resulting from some situations where a single intermediate node is involved in two different sessions simultaneously. As a counterpart to DSR where a shortest path metric (hop count) is used, in SEMR the least congestion path metric is used. Path congestion (PC) is calculated by the summation of so called node congestions NCs (processed data in each node) of the intermediate nodes as follow $PC = \sum_{n}^{i=1} NCi$ where $NC_i$is the node congestion of node i. As we can see from Figure 2.8 that source node S2 in SEMR protocol (to the left hand side) has chosen a primary path ($S2/I5/I6/D2$) to destination which is longer (in hop count metric) than the same node could choose in SMR protocol to avoid node congestion. While a secondary node-disjoint path ($S2/I1/D2$) is used as a backup path. It was shown by simulations that SEMR has superior throughput and end to end delay performance over SMR

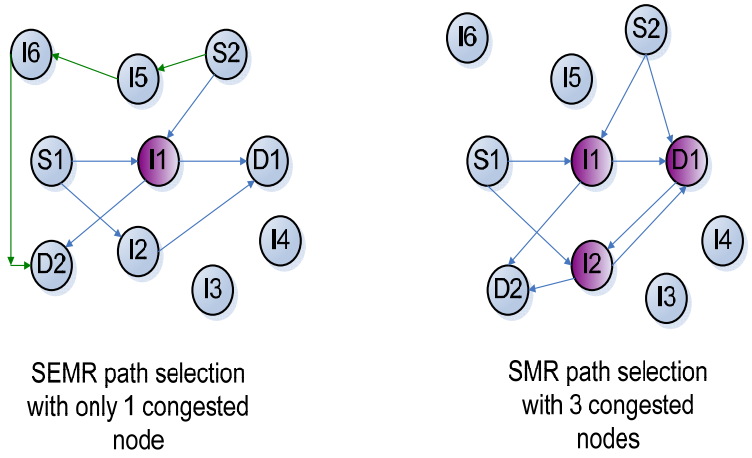


Figure 2.5: Path selction based on congestion metric

(f) *Exploiting network resources and satisfying QoS bandwidth requirement*: in wireless networks bandwidth is a scarce resource and using multiple paths for data forwarding could help to satisfy a real-time application QoS (B.W) requirement. In [45] a ticket-based routing protocol is proposed to confine the overhead of flooding mechanism in route discovery process and provide bandwidth QoS requirement. It is an on-demand protocol where source node S sends some probe packets carrying number of so called a ticket in route discovery process. The main purpose is to search for a path with bandwidth B to destination node D. Each of those tickets has the responsibility to define links along the path which at the end must satisfy the required bandwidth B. To save number of probing packets (decrease overhead), several tickets are carried by one probe packet which also may split (some times merge) in

midway into multiple probes each contain several tickets. This mechanism allows a ticket to split into sub-tickets when it reaches an intermediate node and there is no outgoing link with sufficient B.W. Each of these sub-tickets will search for multiple paths with partial B.W from the required bandwidth B. In this case the original B.W value will split into smaller multiple sub-bandwidth values. Each sub-path will carry partial information. If no links could be found to satisfy B.W requirements, the ticket will be dropped. Figure 2.8 demonstrates the way how tickets search for aggregated B.W. In the

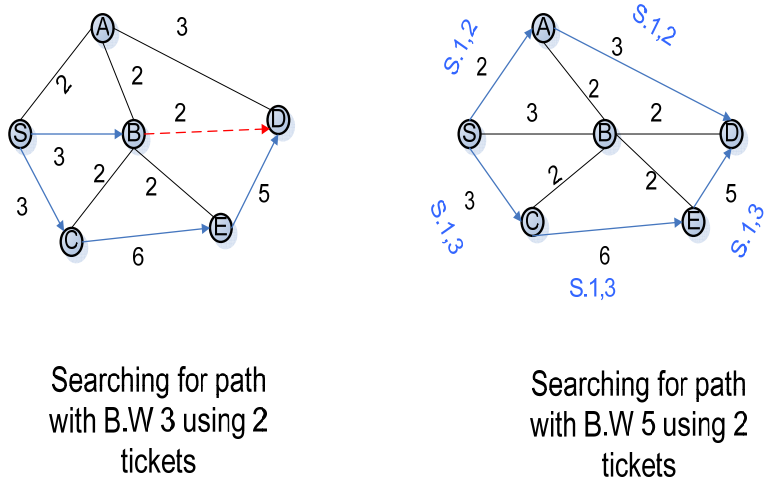


Figure 2.6: Ticket based QoS routing example

figure above B.W for each link is written in black, further the routes are drawn in blue color. We can see in this example that two tickets are split to search for multiple paths which satisfy B.W requirement. For the first example on the L.H.S a path with B.W of 3 MHz is required and a single route (S/C/D/E) is found which is the only path satisfies QoS (B.W) requirement. In the other example on R.H.S where B.W of 5 MHz is required, two paths are found and the traffic will be dispersed among these paths with different rates (2, 3 MHZ subsequently). Simulation results [55] showed the flexibility of this protocol where:

- higher probability in finding satisfactory QoS paths is shown than in case of single path protocols under limited B.W condition
- In the situations where there are no B.W limitations in the network; this protocol performs almost the same as single path protocols in terms of routing overhead.

The drawback of this protocol is that it doesnt specify a method to determine link B.W and it doesnt deal with radio interference problem

2.3.3 limitations on Multi-path protocols

As we saw from previous discussion how multipath routing protocols could drive more enhancements to network performance. Although multipath routing strategy in general provides higher reliability for data transmission, but there are some important issues should be considered in designing such protocols. As in some cases single path routing would be more reliable to implement than multipath one. In the following we will introduce some discussion to make this idea clear and show which critical factors we should consider while designing a multipath routing for a specific network.

1. **Overhead**: in [60] a comparison between reactive single shortest path routing and reactive multipath routing with load balancing is introduced. Also an analytical model is introduced to analyze the overhead for reactive protocols (overhead as it is proportional to number of paths). Also throughput and traffic distribution for general case were introduced. It was mentioned that overhead in reactive protocols is generated from three main sources: route discovery, route maintenance and data transmission. If we look further at route creation frequency, multipath scheme has mostly less frequency compared with siglepath one. In [48] 25 percent reduction is gained for multipath routing in 3-4 hop routes, because route discovery was initiated only upon the failure of all routes. This in turn will reduce the frequency of route discovery floods (RRQ packet flooding). Generally, we can say that at the start up phase both singlepath and multipath protocols will have the same number of RREQ packets which are produced from initial route discovery phase. While multipath protocol will produce more RREP packets overhead, but on long run there will be less route discoveries (less RREQ flooding overhead). One drawback of this multipath strategy [48] is high probability of packet loss due to forwarding the traffic through staled routes (as mentioned in 3.1.2). Also it was mentioned that longer routes will deliver more delay, as more data processing delay (more intermediate nodes are involved in transmission) will be added. In the other hand, multipath routing protocol produced more RREP packets proportional to number of paths (Nu)[60]. It was shown in [48, 60] by increasing number of routes there will be also an increase in the overhead. Also, it was found that the optimal number for paths Nu = 3, because crossing this limit will cause significant rise in overhead. The simulation result shows that in multipath protocol case for Nu = 3 the excess in overhead was approximately 10 percent more than single path protocol when link breakage was less than 10% ; and it was 20% when link breakage rate was higher 50%. Second source of overhead is route maintenance process where an ERR packet is sent back to the source upon a link failure. Where multipath routing will have more overhead packets (RERR) because there will be more routes, therefore more number of path breakages (assuming that probability of path failure among paths is iid random variable). Last source of overhead is data transmission process, where the produced amount of overhead depends on the data forwarding scheme. For example DSR protocol uses source routing, where the entire route is added on the data packet header (longer routes=more overhead). While in other protocol like AODV there will be less overhead because the packet header contains only the address of next hop. Another interesting

notice we found from [60] is that the average length of a route L_m has a big effect on the average number of packets in a queue (Npacm). According to their analysis, it was found that for multipath routing protocol with load balancing mechanism to be more beneficial than single shortest path routing the following condition must hold:
$L_m < (N_{pacs} \times \eta) \div [(N_{pacs} + 1) \times (\pi \times \delta \times R^2 - 1) \times \lambda_m]$, where
L_m: the average length of the route
N_{pacs}: the average number of packets in a queue for single path case
η: the node processing rate
$\pi \times \delta \times R^2$: according to the network model this is area of the circular disc where nodes exist
δ: node density and λ_m is route discovery frequency for multi-path routing. This upper limit value can be used in route selection process in multipath routing protocols. In the other hand we can look at the effect of average length on connection throughput (average transmission rate of the connection [60]). As mentioned before, multipath routing with load balancing mechanism has a great advantage on performance as it spreads the traffic among the network to achieve congestion avoidance. The longer the route length the more nodes are involved in the transmission which in turn will distribute the traffic more among networks nodes. At some limit, more problems will arise because intermediate node will be involved in more data processing load, which might form longer queues inside the nodes buffer. This problem may degrade the network performance significantly as it could reduce the transmission rate and in turn increase end-to-end delay. The upper bound on route length L_{max} to achieve optimal connection throughput was found to be [60] $L_m < \pi \div \beta$, where β is a positive real number which depends on network density. For dense networks β is small and thus there will be less constrains on L_m , while in low dense network care should be taken to the value of L_m in route selection process. At last we can say, in dense networks multipath always outperforms single path routing with respect to connection throughput.

2. **Success probability of data transmission**: the probability of successful transmission is discussed in [61,62] for the scenario where the traffic is uniformly distributed through all available multiple paths. Also M-for-N diversity coding scheme is used to enhance the reliability of data transmission and the paths in this case were node-disjoint. It was found that probability of success P_{succ} (no more than M packets are lost from $N + M$packets) depends on number of paths n and the way how equally sized packets are distributed over multiple paths (traffic allocation). It was found that when the probability of route failure is the same for all routes [61], the probability of success increases with number of used paths (traffic should be distributed over all available paths evenly). In the other hand when a probability of route failure differs among the paths [62], a method is proposed to define the number of multiple paths in order to achieve optimal value for P_{succ}. To satisfy QoS requirement for an application we need to keep P_{succ} within a certain value. A mechanism should determine number of paths and traffic allocation that will lead to the required value of P_{succ}. As a price of high connection end-to-end reliability which multipath protocol will achieve [62], more overhead will be pro-

duced (due to diversity coding scheme). As an overall result, fewer connections will be supported under the same QoS requirements and network capacity.

3. **Load balancing**: the best case of load balancing which a multipath routing protocol can provide is to distribute the traffic evenly among the nodes in the network. Multipath source routing has shown an excellent performance in balancing the traffic load evenly in wired networks case [47]. An analytical model was

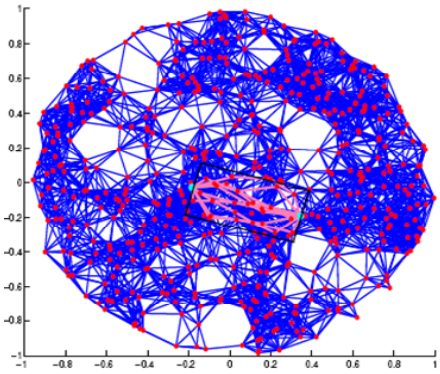


Figure 2.7: First K shortest multi paths

introduced in [63] to evaluate load balancing performance for multipath routing in Ad-hoc networks. The novelty of that work which differentiates it from [60] was the introduction of the effect of multiple paths number and network topology on traffic distribution. The proposed network model consist of a circular disk contains large number of uniformly distributed nodes in a dense network. It was claimed that in dense network, the shortest path between a source and destination will be very close to the straight line connecting these nodes. Also in multipath routing case, first K shortest paths will form a rectangle area around the same straight line. Considering the circle topology, nodes in the center of the circle will be congested and so multipath routing will not distribute the traffic evenly on nodes. This contribution has been derived by both simulation and analytical model.A conclusion was that multipath protocols will have the same performance as single path protocols in terms of load balancing if number of paths is small. To make a significant enhancement in load balance, huge number of paths must be used (was shown to be more than 100). Because this will not be practical, a suggestion has been introduced to solve this problem by choosing the paths which will forward the traffic away from the center of the network. In other words they should be as far as possible from each others.

2.4 Summary

We introduced routing protocols for ad-hoc networks from component design perspective. For different scenarios we have shown which type of these protocols has superior performance and we have summarized this result in Table 2.4. Because we concern Personal Networks in our work, we have presented among different PN scenario conditions which protocol suppose to have best performance. We have summarized this result in Table 2.5. We can see from that table that, further investigation will be focused only on two types of routing protocols which are reactive and proactive routing protocols. Because those types has shown better performance under our proposed PN scenario. Also in this chapter different multipath techniques have been introduced. Each has improved the performance under some specific metric (reliability, delay, PDR). In next chapter we will refer to these techniques again to improve the performance of some proposed protocol for PN. Also we will verify the performance of protocols under PN scenario with different kinds of traffic. But now we would like to mention some important conclusions on multipath techniques:

1. Although node-disjoint paths considered as optimal solution to provide fault tolerance and network resource aggregation, but in low dense networks (PN scenario) it will be difficult to find short node-disjoint multiple paths. So longer disjoint paths might be inevitable which will cause longer delay and out of order packet delivery at receiver node. Notice that longer routes will be less reliable as they have higher probability of breakage (more links will be involved) . But using more efficient route selection metric instead of hop count could overcome this problem. As then reliable paths will be chosen and not shortest paths.

2. Packet granularity in general performs better than per connection granularity as it is more adaptive against failures and traffic fluctuations, but in low dense networks or when there is little number of multiple paths, load balancing will be degraded. This in turn may cause congestion in some intermediate nodes. An efficient forwarding mechanism could solve this problem [68,70].

3. Using diversity coding scheme achieves reliability against route failure, but it delivers more overhead to the network. Also to increase code efficiency, more overhead and more disjoint routes are required.

4. To get best performance out of multipath routing protocol, number of multiple paths shouldn't be more than three (we will investigate the effect of this number on performance in chapter four)

Performance comparison in a PN scenario

3

3.1 Constant bit rate (CBR) traffic pattern

Last chapter we concluded on simulation-based survey that, performance of both reactive and proactive routing protocols are comparable under assumed PN scenario considerations. In this chapter we tried to refine our choice for a dependable routing protocol (either proactive or reactive). We simulated a PN cluster where nodes are mobile except one node (gateway/access point). Source node could be any of mobile nodes and destination node is always the fixed node. We have investigated performance comparison between reactive (AODV, DSR) and proactive (OLSR) ad-hoc routing protocols. We focused on the behavior inside a single PN cluster (intra-routing). The metrics we used in our comparison were derived from dependability requirements from PN which is in general case the ability to offer correct service. We mention these metrics as follow:

- *Packet delivery ratio (PDR)*: this is total number of packets received by destination node to total packet sent from source node. Here we measure the performance of different routing protocols under different conditions in how many data packets will be lost. For some applications (where delay doesn't count) , and it is important to receive all data sent by source node (also think of the case where (FEC) error correction scheme is applied and there is no retransmission channel). The protocol with higher PDR value is considered to be more dependable

- *End to end application delay*: for time bounded applications like voice and multimedia applications, the time a packet takes in the route from source to destination node. This term could affect the quality of some service running on PN node and thus degrades the dependable value of such node. As opposite to PDR metric, a protocol which achieves less delay under simulation conditions is considered to be more dependable.

- *Normalized overhead*: we define it as ratio between total routing traffic sent to total data traffic sent (or in other words the amount of routing traffic generated for routing a single data packet). This metric measures the efficiency of different routing protocols in sending data packets w.r.t generated overhead. Routing protocol which will generate more routing overhead, will consume more resources (bandwidth, battery) and also will cause more delay as there will be more contentions at MAC layer.

In the following section we will introduce simulation environment we used with different varied simulation conditions. Explanation of different behavior is introduced and at the end of the chapter, also we introduce an assessment for each protocol.

3.1.1 Simulation environment

We use OPNET 14.5 as a simulation environment, first we investigate mobile scenario. During simulation we choose only one node as a destination node, which presents a gateway in real PN scenario. As we have introduced in previous chapters, data communication in PN could be between two nodes from different clusters through gateways. For simplicity, traffic pattern used is CBR with 512 byte packet size and a constant rate of 4 packets/sec. Notice that, this traffic presents application traffic (raw data) , before adding any overhead of proceeding OSI layers. Simulation duration is set to 15 minutes (900 sec). 802.11b wireless technology is applied in the simulation environment with 11 M bps data rate and DSSS PHY layer technology. Also CSMA/CA scheme is used at MAC layer. Further we are interested only in mobile scenario, as this is mostly the case in PN environment. We also consider moderate mobility inside a PN cluster (a person is moving, taking the lift, going up/down stairs,etc.) but we don't consider high speed mobility (person in a train, driving a car in highway,etc.) because this is out of the scope in this project.

3.1.2 Simulation scenario

We simulate ad-hoc network which contains 30 mobile nodes in an area of 100 x 100 square meters. Random way point mobility pattern is used with uniformly distributed speed [uniform (0,4) m/sec]. First we investigate how routing protocols will behave when number of source nodes increases. Increasing source nodes presents inject more data traffic load into the network, and we are interested in the behavior of different routing protocols and the ability to manage these loads. In second part we investigate performance of routing protocols with varied mobility patterns and in the last part we investigate the behavior of routing protocols under different network densities.

3.1.3 Effect of number of source nodes on the performance

In this scenario we fix mobility pause time (a factor which determines degree of mobility) [75] to 300 seconds and we change number of source nodes to (5, 10, 15, 20, 25) respectively. A source node is an initiator for a CBR traffic to one fixed destination node (e.g. gateway). Bellow we analyze results from this scenario with respect to previously mentioned metrics.

1. **PDR**
   Figure 3.1 shows the behavior of routing protocols with respect to PDR metric. OLSR as a proactive routing protocol has the worst behavior, as there will be higher collisions at MAC layer between data and routing packets. Notice that all nodes in OLSR case are sending periodic control messages. Increasing source nodes lead to increase of data packets routed in the network and thus higher probability of collisions. DSR has better performance than AODV when number of source nodes increases. This is because of aggressive route caching mechanism. DSR routing protocol allows a source node to cache all available routes to destination node, thus when a primary route is congested or not available the traffic will be switched to a secondary route without invoking route discovery process. This

will prevent in some degree the packet loss. Also packet salvaging mechanism in DSR allows an intermediate node to switch the traffic to alternative route if it is available in the cache upon link breakage which could guarantee higher PDR than AODV case. By increasing number of source nodes, more nodes will have the opportunity to obtain routing information. They will be source nodes and intermediate nodes at the same time. Because destination node is fixed for all sources, then this will work in enhancing the performance of packet salvaging in DSR. Notice that low mobility pattern is used (proposed PN scenario) , but with higher mobility the cached routes in DSR case could be stale. This would degrade the performance as we will see in later section.

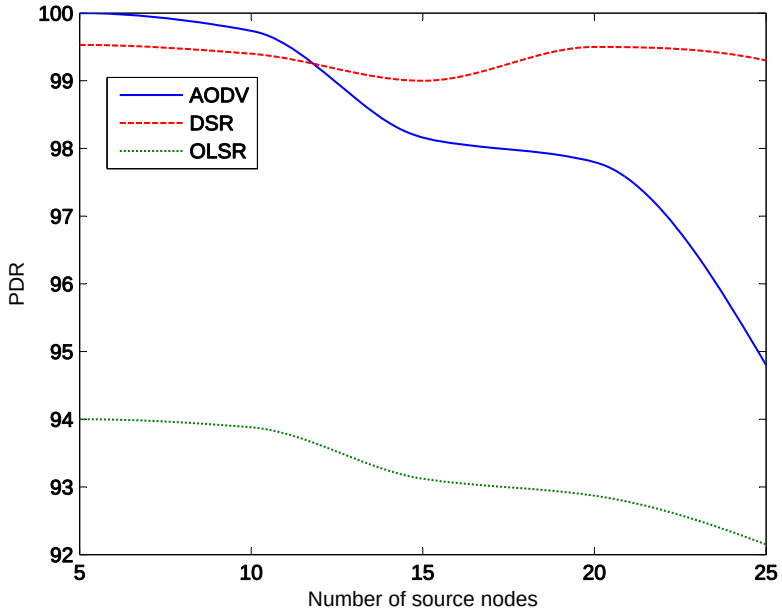


Figure 3.1: Effect of increasing number of source nodes on PDR performance

2. **Application end-to-end delay**
First we like to mention that this delay presents the time elapsed between sending a packet from application layer at source node and receiving the same packet at application layer of destination node. We see from Figure 3.2 that, when number of source nodes increases the delay also increases because the total load on the network will increase and more of network resources will be used. An intermediate node could be responsible for routing two different traffic packets into different directions at the same time which will increase the processing and queuing time at this node and thus will increase end-to-end delay in total. OLSR has better performance than reactive protocols (AODV, DSR) with respect to delay when number of source nodes increases. This is shown in Figure 3.2 at region of x-axis where number of source nodes exceeds 10. Reactive protocols before this region, especially AODV, have better delay performance than OLSR. As we will see in Chapter 4 the application end-to-end delay is influenced by route discovery time

for reactive protocols. Remind that in our scenario source nodes start to send traffic at the same time (at time stamp 100 sec.) When number of source nodes is higher than 10, the network will be flooded instantly with large number of RREQ packets in case of reactive protocols. Therefore both AODV and DSR will show larger delay than proactive OLSR protocol when number of source nodes exceeds 10. Contrary, when number of source nodes is small (less than 10) periodic control messages for OLSR (HELLO and TC) will have higher contention with data packets at MAC layer than the contention caused by control packets in reactive protocol case. DSR has lower performance than both protocols, because of source routing mechanism. Data packet in DSR case carries the description of whole route on its header, that presents more byte overhead processing time than in AODV case.

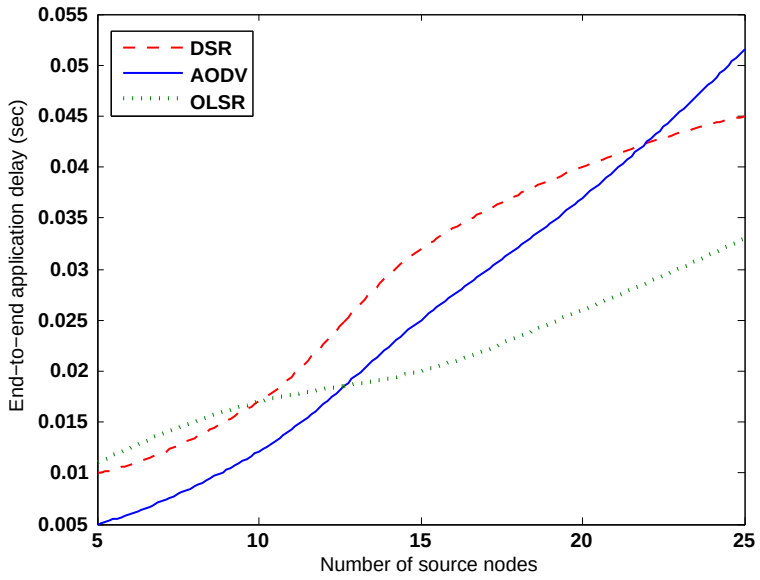


Figure 3.2: Effect of increasing number of source nodes on end-to-end delay performance

3. **Normalized overhead**
   As OLSR has constant amount of routing overhead (proactive protocol) , thus by increasing amount of data traffic sent we see that normalized overhead will decrease. We like to mention that, OLSR as link state protocol has a unique mechanism where nodes advertise only the links which represent multi point distribution relay (MPR) selections and thus reduce routing overhead to less amount than other link state protocols(OSPF,IS-IS). Also periodic topology control (TC) messages in OLSR case are broadcasted only by (MPRs) nodes while RREQ messages in case of reactive protocols are broadcasted by all nodes. This reduces the amount of packet flooding in OLSR case. AODV generates more routing overhead than DSR when number of source nodes increases because all intermediate nodes

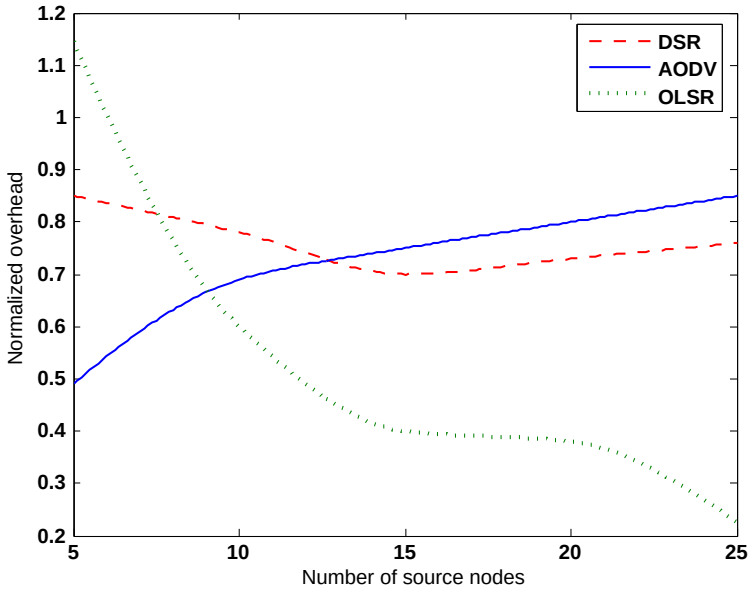


Figure 3.3: Effect of increasing number of source nodes on normalized overhead performance

through one source-destination route will generate periodic HELLO packets for discovering of local connectivity. While in DSR case, only RERR messages upon link breakage will be exchanged, and route discovery process will be invoked less often than with AODV case.

3.1.4 Effect of mobility pause time on the performance

In this scenario we investigate the behavior of routing protocols with different mobility patterns. Lower pause time presents higher mobility and vice versa. We fix number of source nodes (10 nodes) under varying pause time (0 50 100 200 250 300 sec). This scenario presents different rates of link breakage. For higher mobility there will be higher rate of link breakage, each protocol with different route maintenance mechanism will have different behavior. As explained in earlier chapters, OLSR generates periodic messages for topology information exchange (HELLO message within MPR set and TC messages for all network). Wherever link breakage happens, the intermediate node informs its MPR node with this event (using HELLO packets). MPR in turn will inform the rest of nodes in the network to update routing tables (by broadcasting TC messages). In reactive protocol case, AODV implements periodic HELLO messages between intermediate nodes which are involved in transmission to update local connectivity information. Upon link breakage, (no HELLO received during threshold time period, or no ACK received at MAC layer) the node invokes a route discovery process locally. If this process failed, the node will drop the data packet and send RERR message to source node which will pass through intermediate nodes and update their routing table. Upon receiving RERR message, the source node starts route discovery to search for alternative route. In DSR case, there will be no periodic HELLO messages,

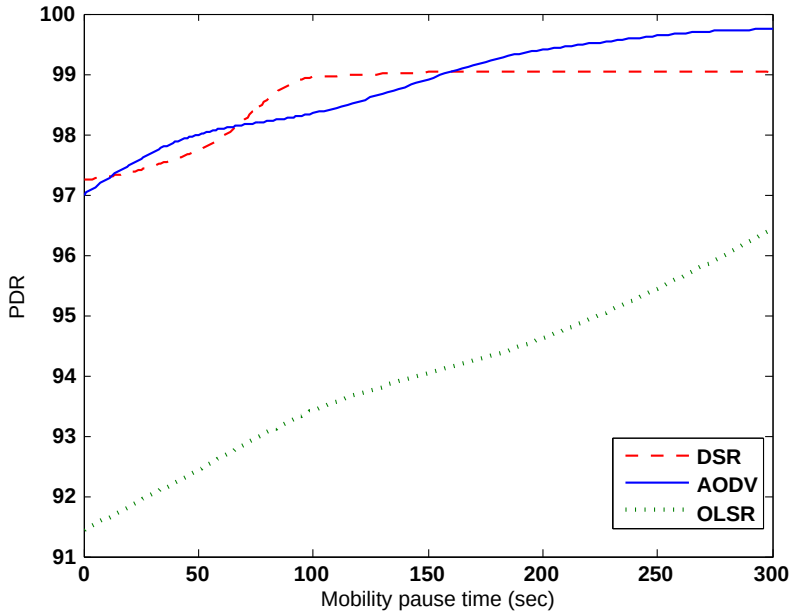


Figure 3.4: Effect of increasing pause time on PDR performance

and a new route discovery will be invoked only when all cached routes become invalid. Source node is the only node which invokes route discovery when all routes in the cache are deleted. When intermediate node in DSR case discovers link breakage (no ACK received upon sending data packet) , either it sends the packet through alternative cached route (if available) or simply it drops the packet and sends RERR message to source node. Bellow we analyze the results of varied mobility scenario with respect to our previously mentioned metrics.

1. **PDR**
   In general and as we see from Figure 3.4, PDR is inversely proportional to mobility degree for all routing protocols. Zero pause time indicates that nodes are continuously moving from one point to the other in random walk model. DSR has lower performance than AODV in higher mobility because of stale route problem. As in DSR there is no mechanism to check the validity of cached routes (which in mobile scenario probably will be stale). Packets could be routed to invalid routes and will be lost. OLSR has lower performance than reactive protocols especially for high mobility case where more links will be broken. Because higher number of routing tables must be updated in high mobility scenario. Data packet could be sent to incorrect entry and lost. While local route discovery in AODV case and aggressive route caching in DSR case will help to achieve less packet loss. The performance of OLSR in high dynamic scenario could be enhanced by changing time interval between successive HELLO messages to enable updating topology information in timely manner, but this could consume more resources (bandwidth, energy).

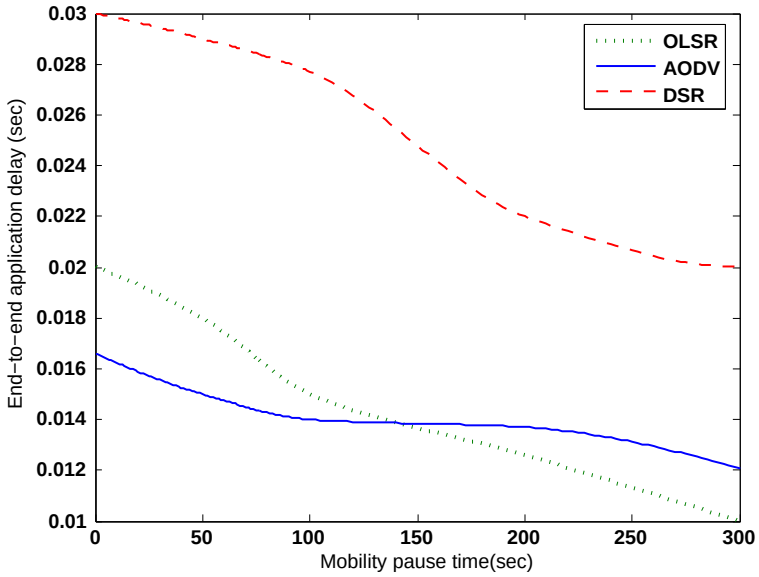


Figure 3.5: Effect of increasing pause time on end-to-end delay performance

2. **Application end-to-end delay**
   From Figure 3.5 we see that end-to-end delay increase with higher mobility. Data packet could be successfully sent after multiple retransmissions at some intermediate nodes. AODV has the best performance when mobility increases. Also OLSR considered as a competitor with small difference of delay in high mobility and better performance in low mobility pattern. Table driven mechanism in both AODV and OLSR shows better performance than source routing in DSR. Data packets with DSR record in average 10 m.sec delay time more due to byte overhead processing.

3. **Normalized overhead**
   Figure 3.6 shows that routing overhead is inversely proportional to mobility. At high mobility case, more links will be broken and thus more messages have to be generated to report these events and update network topology. This process is called route maintenance process as we explained earlier. OLSR has lower amount of overhead comparing with reactive protocols due to his proactive nature. For high mobility (zero pause time) both reactive protocols generate more overhead and DSR has less amount of overhead because of aggressive route caching while AODV will invoke route discovery process more often which means generating more routing overhead packets.

Till now, number of nodes was fixed (30 nodes) for all previous scenarios. In next part we will investigate the scalability of routing protocols (AODV, DSR, OLSR) when number of nodes increases. In typical PN cluster, we assume that number of nodes doesn't exceed 100 personal nodes and thus we will investigate the performance of routing protocols when size of network will be incremented from 30, 50, till 100 nodes.

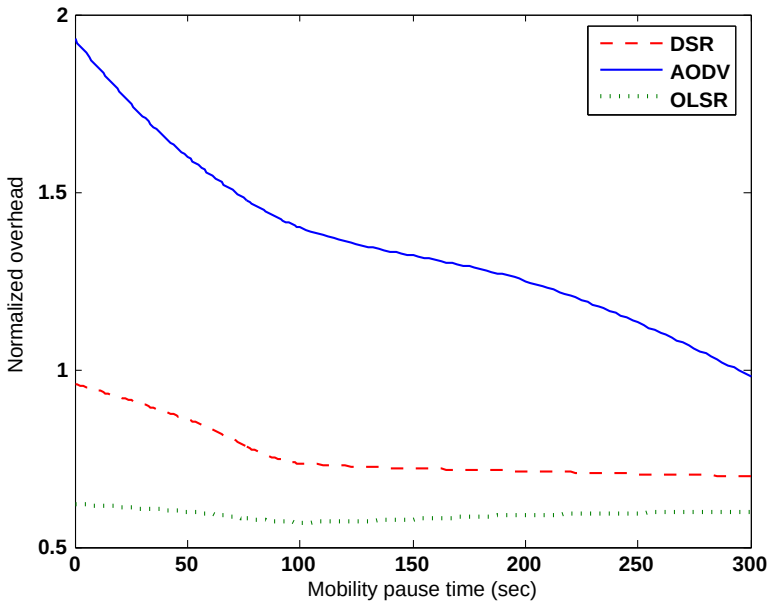


Figure 3.6: Effect of increasing pause time on normalized overhead performance

3.1.5 Effect of network size on the performance

In this scenario we investigate the scalability of routing protocols. As network size increases, time to discover a route to remote destination might increase. With mobile scenario this could form a challenge to routing protocol, as topology is dynamically changing. We can think of a real scenario where a person (carrying some personal devices, PDA, mobile telephone, body sensors,etc.) enters his office cluster (which contains some other personal nodes: printer, fax, access point, number of PC's,etc.) This increase in network size could affect the overall dependability performance of PN. Bellow we will introduce the behavior of routing protocols with increasing number of nodes in a network with fixed service area (100x100 meter square) .

1. **PDR**
   For mobile scenario, increasing network size has slightly degraded the performance of all protocols. For ad-hoc networks in general, increasing node density will increase the degree of connectivity[72]. In worst case (OLSR) , increasing the network size by 333.33 % caused degradation of 1.08 % in PDR performance. From Figures (3.1,3.4,3.7) we conclude that, OLSR protocol has the worst performance with respect to PDR metric in mobile scenario under all conditions. This we refer it to proactive nature of OLSR, as increasing number of nodes will increase the amount of routing tables which must be updated due to dynamic topology changes. Aggressive route caching mechanism enables DSR to record higher PDR values when network size increases than AODV protocol. Backup routes are stored in the cache to piggyback the traffic in case of primary route failure. Also for DSR protocol, the ability of intermediate node to replay on route request with a valid

route from the cache helps to enhance the performance of DSR.

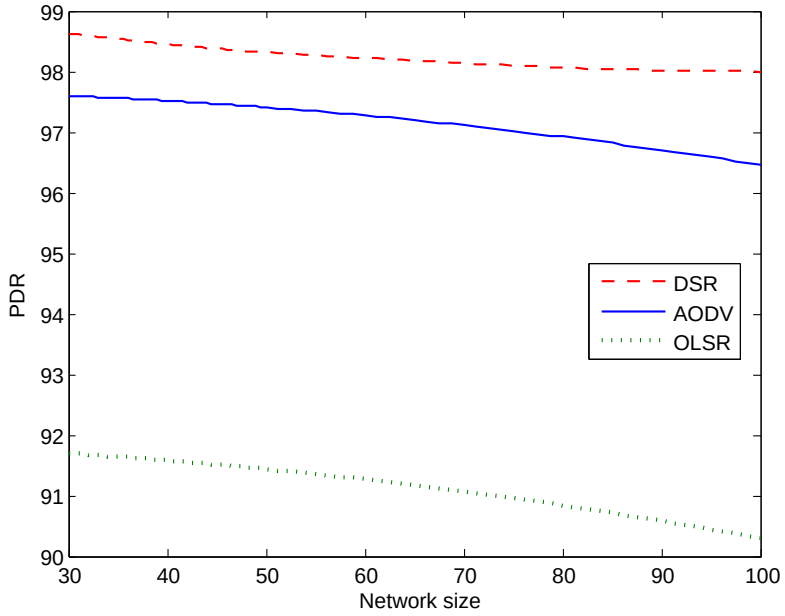


Figure 3.7: Effect of increasing network size on PDR performance

2. **Application end-to-end delay**
   Increasing network size causes an increase in end-to-end delay for all cases. With OLSR protocol, less amount of delay is recorded. A route to destination is always available for data packet in the routing table, thus no need to wait for discovering a route. And in general because OLSR has efficient mechanism for broadcasting routing packets as discussed earlier, data packets will encounter less delay due to contention with routing packets at MAC layer level. Also AODV has worse performance than DSR when network size increases because of more generated overhead, which will cause more collisions with data packets and as a result more retransmissions and longer delay for data packets will occur.

3. **Normalized overhead**
   Because of route caching mechanism we see from Figure 3.9 that DSR curve has a very small slope comparing with other protocols. As we explained earlier in DSR case , after the discovery of multiple routes to destination node the control traffic is reduced to the minimum (zero value in static scenario). AODV has the worst performance as with increasing network size the number of broken links will also increase. More route discoveries will occur whether locally or at source node which will increase the amount of routing overhead. For OLSR protocol with proactive nature, increasing number of nodes will cause increase of total amount of periodic routing packets which are generated by all nodes in the network.

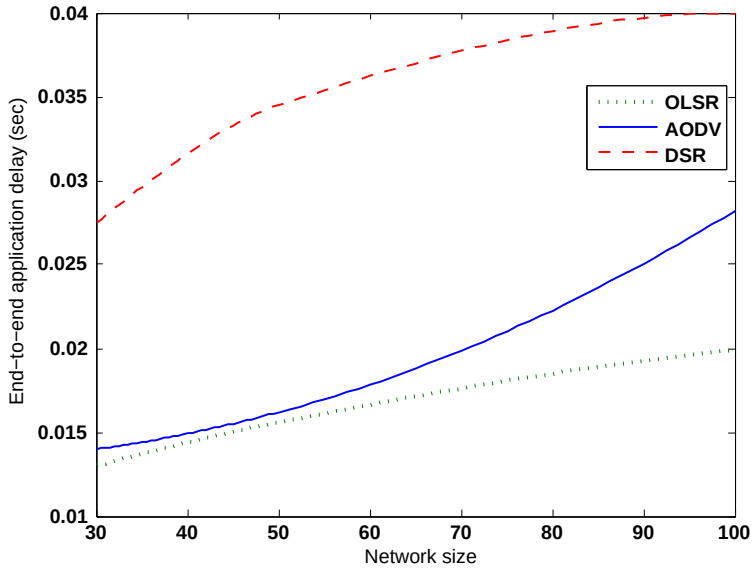


Figure 3.8: Effect of increasing network size on end-to-end delay performance

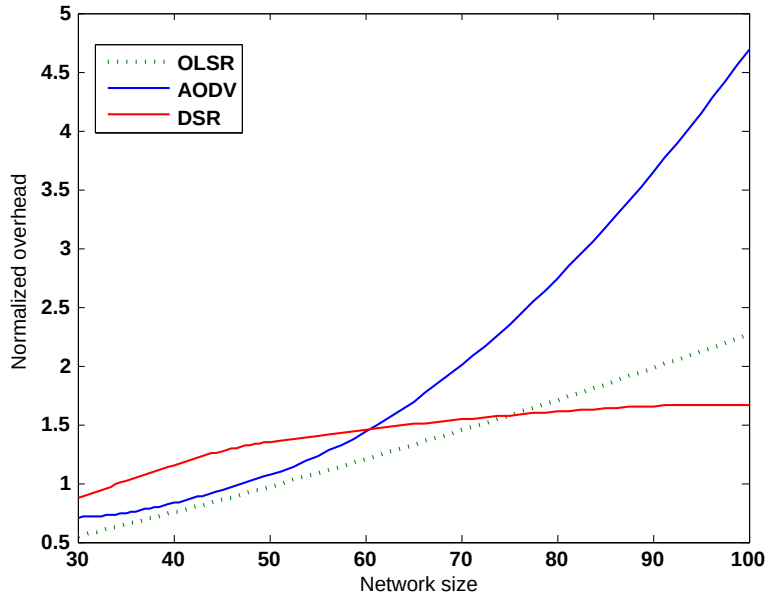


Figure 3.9: Effect of increasing network size on normalized overhead performance

3.1.6 Summary

In this part we have investigated the behavior of routing protocols under different scenarios. In Table 3.1 we summarize the performance of routing protocols under different scenarios. According to observed performance results for routing protocols,

a star sign (*) is given to protocol who has shown best performance under specific scenario. We can consider only both DSR and OLSR in further investigation towards

Table 3.1: Average performance comparison

Routing protocols	Increasing source nodes	Increasing mobility	Increasing number of nodes
AODV	-	*	-
DSR	*	-	*
OLSR	-	-	-
PDR performance			
AODV	-	-	-
DSR	-	-	-
OLSR	*	*	*
Delay performance			
AODV	-	-	-
DSR	-	-	*
OLSR	*	*	-
Normalized routing overhead performance			

dependability. The reason is that:

- DSR guaranties less data packet losses for various scenarios
- OLSR has the best performance with respect to application end-to-end delay
- Both protocols have nearly comparable performance with respect to routing overhead. OLSR has better performance due to efficient flooding mechanism.

We can see from above discussion that we still need more investigation regarding other criteria on dependability to refine our search for dependable routing protocol. We notice a comparable performance between OLSR and DSR regarding amount of generated overhead. Because overhead performance has an effect on the amount of network resources consumption especially energy and bandwidth, we need to investigate the performance of previously mentioned protocols with respect to energy consumption behavior. But before this step we will investigate the behavior of these routing protocols under commonly used realistic traffic patterns. That what we will introduce in next part, as we will introduce the performance comparison of these protocols with variant traffic patterns like (HTTP, FTP, VOIP, video conferencing).

3.2 Performance comparison under realistic traffic patterns

In last part we couldn't get a clear view on which protocol definitely could provide dependability for PN. We conclude that constant bit rate (CBR) traffic model which is used in our previous simulation study might not reflex the real behavior for protocols under investigation. OLSR protocol from one side records less delay and overhead generation, while DSR records high PDR. Therefore we decided to test the performance

of these protocols under some realistic traffic. By realistic traffic we mean some expected applications that will probably run in PN. Most of these applications require interaction between two entities (server/client or client/client) e.g. web browsing, file transfer, voice over IP call. By interaction we mean there will be request and response messages between these entities excluding data packets. We are interested in delay and PDR metrics only in this scenario because they can measure the performance under real time applications. Bellow we introduce simulation scenario.

3.2.1 Simulation of realistic traffic

We present PN cluster with area of 100x100 square meter contains 50 mobile nodes. IEEE 802.11b wireless technology is implemented with CSMA/CA MAC layer presentation and DSSS PHY layer technology. Random way point mobility model is used with pause time = 100 seconds and velocity = uniform(0,4). We compare the performance of three different protocols (OLSR, AODV, DSR) with different traffic flows for 15 minutes (900 seconds) simulation run time .In our scenario there will be one mobile source node and one fixed destination node (could be a gateway in real scenario). Bellow we present these flows and resulting performance for routing protocols:

1. **Web browsing traffic**
   These are HTTP1.1 applications where the user downloads a page from a server (destination node). This page contains text and graphic information. Transport protocol used is TCP and further details about the traffic are presented in Table 3.2. First comparison metric is delay which is presented by Object response time

Table 3.2: HTTP traffic parameters

Attribute	value
Page inter-arrival time(sec)	exponential(720)
Size of text object per page (bytes)	500
Number of small images per page	5

   (response time for each inline object from the HTML page) and Page response time (time required to retrieve the entire page with all contained inline objects) , and second metric is PDR. Table 3.3 illustrates delay and PDR response for different protocols As we expected, OLSR has the less delay performance followed

Table 3.3: Performance of protocols under HTTP traffic type

Metric	OLSR	DSR	AODV
Object response time (sec)	0.01	0.02	0.03
page response time (sec)	0.02	0.12	0.32
PDR	100	100	100

   by DSR and AODV has the worst performance. PDR performance is excellent for all protocols with HTTP kind of traffic. Now we need to test other kind of traffic because we still don't have much difference.

2. **Video conferencing traffic**
   This application allows users to transfer streaming video frames across the network. This application is from the type client/client applications. By default, UDP transport protocol is used. Further details about this traffic is presented in Table 3.4.

Table 3.4: Video conferencing traffic parameters

Attribute	Value
Frame rate	10 frames/sec
Frame size	128 x 120 pixels
Type of service	best effort

   Delay in this scenario is presented by (Packet delay variation) which is the variance among end-to-end delays for video packets, and Packet end-to-end delay which is the time taken to send a video application packet to a destination node application layer. In Table 3.5 we present the results for different protocols. Average packet delay variation is the variance among end to end delays for video packets. End to end delay for a video packet is measured from the time it is created to the time it is received and is presented in lower part of the graph.

3. **VOIP application**
   This application enables two users to communicate together using digitally encoded voice signal (client/client). UDP transport protocol is used. The voice data arrive in bursts followed by silence period. Table 3.6 presents detailed application configuration

   Delay metric in this case is presented by: Jitter which is the difference in time between packets received at destination and packets generated at source node. Negative jitter indicates that time difference between packets at destination node is less than time difference between same packets at source node. Further we are also interested in packet delay variation and end-to-end delay (which have been previously defined in video conferencing part). We introduce simulation results in Table 3.7.

Table 3.5: Performance under video conferencing traffic

Routing protocol	PDR	Average packet delay variation(sec)	Average end-to-end delay(sec)
OLSR	1.7	.001	.06
AODV	2.8	.013	.2
DSR	4	60	18

Table 3.6: VOIP traffic parameters

Attribute	Value
Silence length (sec)	incoming/outgoing = exp(0.65)
Talk burst length (sec)	incoming/outgoing = 0.352
Encoder scheme	G.711
Voice frames per packet	one
Type of service	interactive voice
Signaling	H323
Compression delay (sec)	0.02
Decompression delay (sec)	0.02
Conversation environment	closed room

Table 3.7: Performance under VOIP traffic

Routing protocol	PDR	Average packet delay variation(sec)	Average end-to-end delay (sec)
OLSR	67.43	.12	.6
AODV	39.6	.035	.18
DSR	20.24	4	60

4. **File transfer FTP**
   An FTP application is a client/server application which enables transferring a file between a client an a server. In our scenario the mobile node will send a request to the fixed (gateway) node to download a file. The client (mobile) node first will send a request message (512 bytes) to the server (gateway), and the server will replay with the required file. TCP protocol is used to open a conversation channel between both nodes. This channel is used for both control and data packets. Our delay metric is download response time, which is the time elapsed between sending a request and receiving the response packet. And upload response time which presents time elapsed between sending the file and receiving the response. Table 3.8 introduce traffic parameters and Table 3.9 presents the performance of different protocols.

Table 3.8: FTP traffic parameters

Attribute	Volume
Inter-request time (sec)	exp (50)
File size (byte)	1000
Type of service	best effort

5. **Mixed traffic**
   As in previous section there was one source node sends a traffic to the destination, in this section five source nodes will send traffic to the same destination (fixed node). Each source has different kind of traffic. From this we want to investigate the performance of routing protocols (especially the delay) under more realistic scenario. Table 3.10 summarizes the behavior of different protocols:

Table 3.9: Performance of routing protocols under FTP traffic

Protocol	Average download time (sec)	Average upload time (sec)	PDR %
OLSR	0.03	0.63	100
AODV	1.1	20	100
DSR	0.06	5.3	100

Table 3.10: Performance of routing protocol under mixed traffic

Application	PDR %	Average delay (sec)
HTTP	100	object/page response time = 1.044/2.387
FTP	100	download time = 0.034
VOIP	4.54	jitter = 1,925 ; delay = 74.127
Video	1	delay = 26.45
DSR		
HTTP	100	object/page response time = 0.0116/0.02544
FTP	100	download time = 0.0127
Voice	6.94	jitter = 0.1 ; delay = 2.96
Video	3.57	delay = 11.74
OLSR		
HTTP	100	object/page response time = 0.288/1.78
FTP	100	download time = 7.173
Voice	9.32	jitter = 0.0313 ; delay = 31.69
Video	1.45	delay = 12.755
AODV		

3.2.2 Summary

From results of several traffic patterns which are introduced in above sections we can give the following conclusions:

1. TCP protocol has a major effect on dependability and must be used as transport protocol for PN applications. We see that all applications which use UDP show very bad performance in terms of both PDR and delay. While in case of TCP, all protocols could record 100% PDR value.

2. In general case, OLSR has the best performance under different realistic traffic patterns and mobile scenario. Regarding to results achieved from first part (best delay, overhead with CBR traffic pattern), we conclude that structural design of OLSR routing protocol is suitable to achieve dependability for intra-routing in PN cluster. Some further modifications are needed to enable QoS routing and to enhance PDR performance. We will introduce these modifications after investigating the power consumption performance in next part.

We also summarize the delay performance of routing protocols under different realistic traffic scenarios in Figure 3.10.

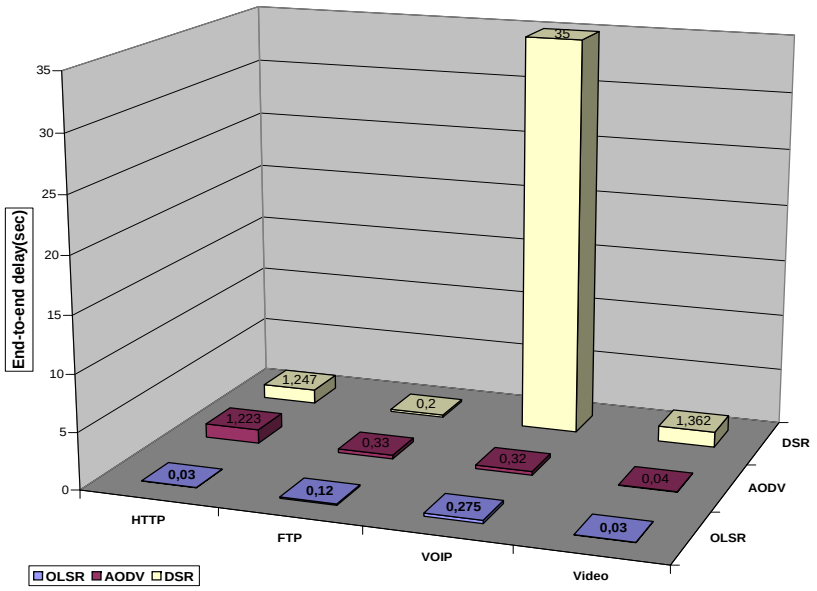


Figure 3.10: Delay performance for routing protocols under realistic traffic

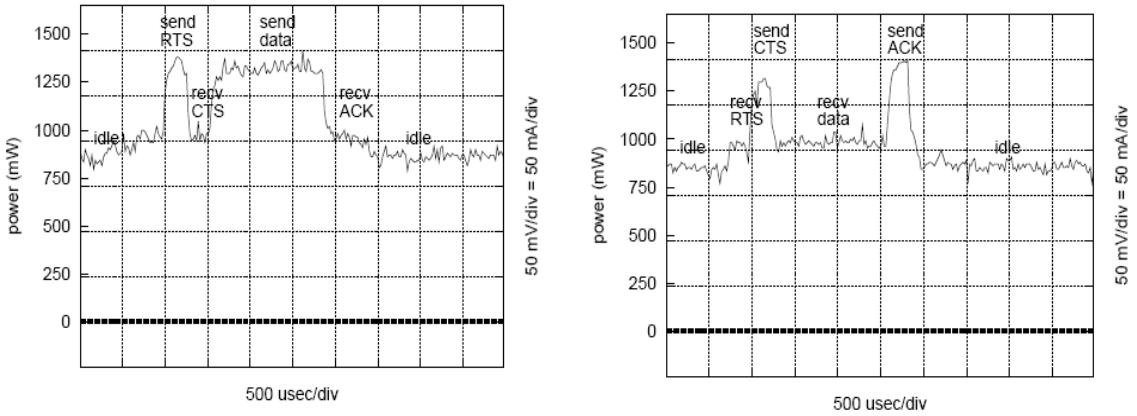
3.3 Evaluation of energy consumption

Because energy consumption for wireless nodes using IEEE 802.11 wireless technology is relatively high. As nodes in idle mode consume relatively the same power comparing to receive/transmit mode. In this chapter we will investigate energy consumption performance of wireless nodes with different routing protocols (OLSR, AODV, and DSR). We will not try to implement a new technique to reduce power consumption, but we will see which routing protocol with its default implementation allows for less energy consumption. Beside this main subject we will also mention some other conclusions observed during simulation which are summarize bellow:

- The effect of HELLO message on AODV performance
- Effect of increasing data rate on delay
- Impact of MAC layer on the delay of data packets

We also will conclude on the performance of OLSR regarding energy consumption and we will give some recommendations to enhance its performance at the end of this chapter. In Figure 3.13 we show outcome of measurements [88] which have been carried out to investigate energy consumption behavior of lucent Wave LAN IEEE 802.11 wireless

network interface operating in ad-hoc mode. It has been mentioned that, because nodes in ad-hoc mode doesnt perform sleep mode there will be high power cost for idle mode which is measured to be slightly less than power consumed for reception. Thus, routing protocol designers should consider the portions of traffic which include broadcast and point-to-point transmissions used by the protocol (HELLO, ACK, RREQ,etc.) Also



(a) sending data

(b) receiving data

Figure 3.11: Receiving and sending 2Mbps point-to-point UDP/IP traffic (256 bytes)[88]

it has been shown that [76] for wireless interface the ratios Idle: receive: transmit are 1 : 1.05 : 1.4 respectively. As we see, both idle mode and reception mode consumes comparable amount of energy comparing to transmission mode. We would like to investigate which routing protocol has the least power consumption with its default implementation. Therefore we will develop a cost function to calculate the energy cost per sending Q data packets with B bits per packet through route K in a network with N number of nodes. We propose the cost function to be as follow:

$$C_k = S_k + R_k + idle \ , \tag{3.1}$$

where

- C_k: aggregated energy cost needed to send Q data packets through route K
- S_k: total energy cost for sending data packets
- R_k: total energy cost for receiving data packets
- $idle$: energy consumed in idle state for all nodes

In our static scenario, default hop count was the route selection metric for all protocols. And the ratio between active to idle nodes was also the same for all protocols. Thus, we have normalized our calculation to idle value (omitted from further calculations). To calculate energy consumption we will use OPNET simulation program and gather total number of packets (transmitted/received) for all nodes in the network. In our scenario we assume a wireless network with homogeneous type of nodes IEEE 802.11b.

For further energy consumption calculation we will refer to the energy model which is introduced in [78]. This model states that, energy consumed for transmission is given by

$$E_t = (e_t + e_d \times d^n) \times B_t \ ,$$

And energy consumed for reception per second is given by

$$E_r = e_r \times B_r \ ,$$

where e_t and e_r is the energy consumed to transmit/receive one bit in transducer electronics. The term e_d is the energy consumed by the amplifier after transmitter stage to ensure acceptable received SNR at distance d from the transmitter, B is total number of bits and n is path loss exponent and it is equal to two in free space propagation (path model in our simulation). Further we will probe packets at MAC layer level, and the total energy consumed for transmission could be calculated as:

$$S_k = \left(E_{td} \times \sum_{N}^{i=1} D_{ti} \right) + \left(E_{tc} \times \sum_{N}^{i=1} C_{ti} \right) + \left(E_{tz} \times \sum_{N}^{i=1} Z_{ti} \right) \ , \tag{3.2}$$

where C_{ti} is total number of MAC layer control packets transmitted (ACK, RTS, CTS) ,D_{ti} is total number of data packets transmitted taking into account number of re-transmissions and Z_{ti} is total number of routing packets transmitted per node(i) during simulation time. E_{td} is energy consumed per data packet transmission and $E_{tc} and E_{tz}$ is energy consumed per MAC/ routing overhead control packet transmission. Similarly, total amount of consumed energy for reception is given by:

$$R_k = \left(E_{rd} \times \sum_{N}^{i=1} D_{ri} \right) + \left(E_{rc} \times \sum_{N}^{i=1} C_{ri} \right) + \left(E_{rz} \times \sum_{N}^{i=1} Z_{ri} \right) \ , \tag{3.3}$$

where C_{ri} is total number of MAC layer control packets received,D_{ri} is total number of data packets received and Z_{ri} is total number of routing packets received per node(i) during simulation time. E_{rd} is energy consumed per data packet reception and $E_{rc} and E_{rz}$ is energy consumed per MAC/ routing overhead control packet reception. From [78] we have

$$e_t = e_r = 50 \ nJ/bit; e_d = 100 \times 10^{-12} \ J/bit/m^2 \ , \tag{3.4}$$

In our case we have grid topology with fixed distance d=20 meters between nodes. Then we can calculate total energy consumed by gathering all packets (transmitted/received) with different sizes for each node in the network and substitute Formulas (3.2,3.3) in Formula (3.1). The size of different packets 802.11 MAC level is standard and as follow:

- MAC protocol data unit($MPDU$) = 34 × 8+payload size(either raw data or routing control)
- MAC control packets:
  - $RTS = 20 \times 8 = 160$ bits
  - $CTS = ACK = 14 \times 8 = 112$ bits

In the following part we will introduce a comparison between three routing protocols (OLSR, AODV, and DSR).

3.3.1 Related work

The work in [81] has investigated the energy consumption for both DSR and OLSR routing protocols. NS2 simulation program was used and energy driven from mobile nodes battery using IEEE 902.11g (NIC) wireless interface was modeled and implemented in software. It has been shown that reactive protocols consume less energy than proactive ones in low data rate scenario. Proactive protocols could perform well in higher data rates if more energy efficient route refresh mechanism is used. Further more, the overhearing mechanism and the idle mode independently from routing protocol affects the performance substantially and has a dominant rule in energy consumption. New techniques should be investigated to reduce energy consumption for wireless interfaces in idle mode to make it more dependable. Our work in this chapter differs by adding AODV routing protocol to comparison. In general case AODV has more routing overhead (HELLO messages) than DSR, but the later one has more byte overhead (source routing). Thus it was not obvious to know which of them consumes less energy.

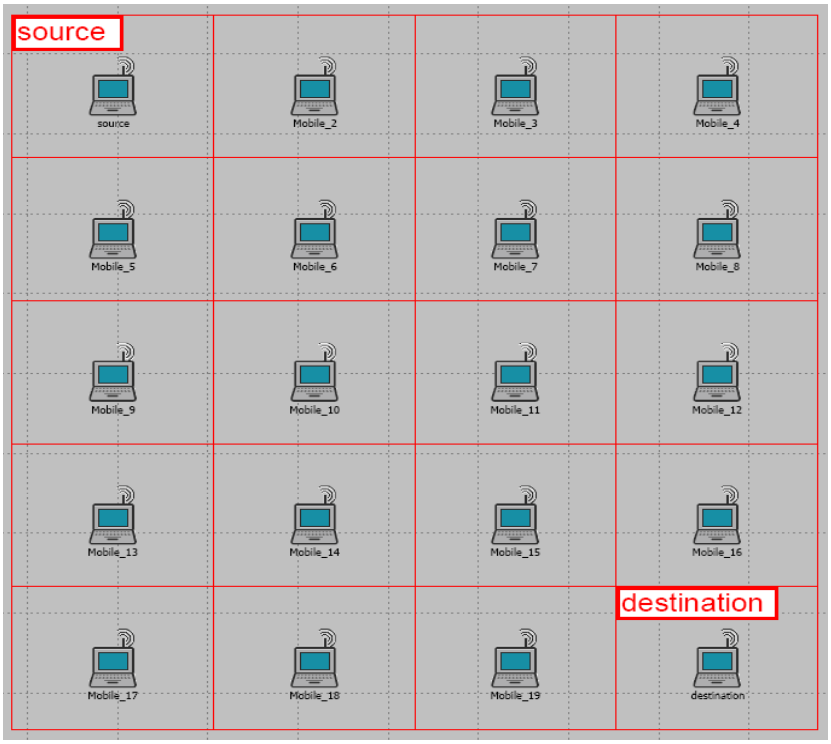


Figure 3.12: Network topology

3.3.2 Simulation scenario

In order to compare energy consumption behavior among different ad-hoc routing protocols (AODV/DSR/OLSR), we deployed fixed network consists of 20 ($manet-station$) nodes which have capability of generating raw data and relaying this data using ad-hoc routing protocol. Grid topology with dimension of 80 x 100 square meters is used.

Transmission power was set to 2e-005 watt (was suitable to establish two hop communication with low BER) . Five different simulation runs were carried out, each with different data rate to investigate the energy consumption behavior vs. offered load for each protocol. Application data was sent with CBR as follow [run1 = 1, run2 = 10, run3 = 100, run4 = 300, run5 = 600 packets/sec], with 1024 bits per packet. We have chosen this range of data rates to simulate typical rates which are used in Internet video streaming [86] and health care applications [85]. The traffic starts at time 100 seconds and total simulation time 5 minutes(300 sec). Nodes are working in ad-hoc fashion, and there is no access point to communicate with outside world (only DCF is used). IEEE 802.11 b wireless technology is used with 11 M bps link capacity. Statics were collected in bucket/sum mode with 300 seconds bucket size (simulation period), this allows gathering total number of packets transmitted/received for all nodes. Figure 3.14 illustrate network topology used in our simulation using OPNET 14.5 [83].

3.3.3 AODV routing protocol

Referring to AODV mechanism, one could expect that AODV protocol will consume more energy more than DSR due to the broadcasted HELLO packets (notice that we are using fixed topology in this chapter). But as we mentioned in previous chapter, DSR has more byte overhead than AODV which lead to more number of bits per data packet transmitted. Thus it is not obvious which protocol will consume more power. For AODV protocol, periodic HELLO packets are used to perform local connectivity measurement. And all nodes in an active route will send periodic HELLO messages during HELLO interval time. We found that broadcasted HELLO messages affect the overall performance as it causes collisions in MAC level with data packets. We will explain this later after introducing HELLO packet format. HELLO packet takes the same format as RREP packet with TTL field set to one [82]. This is described in Figure 3.15. The field parameters are set as follow:

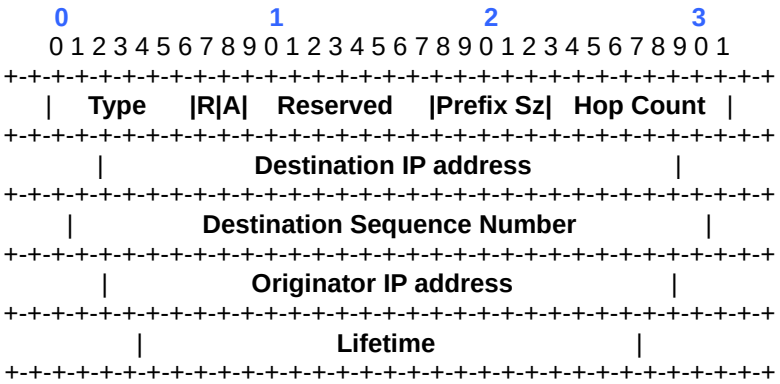
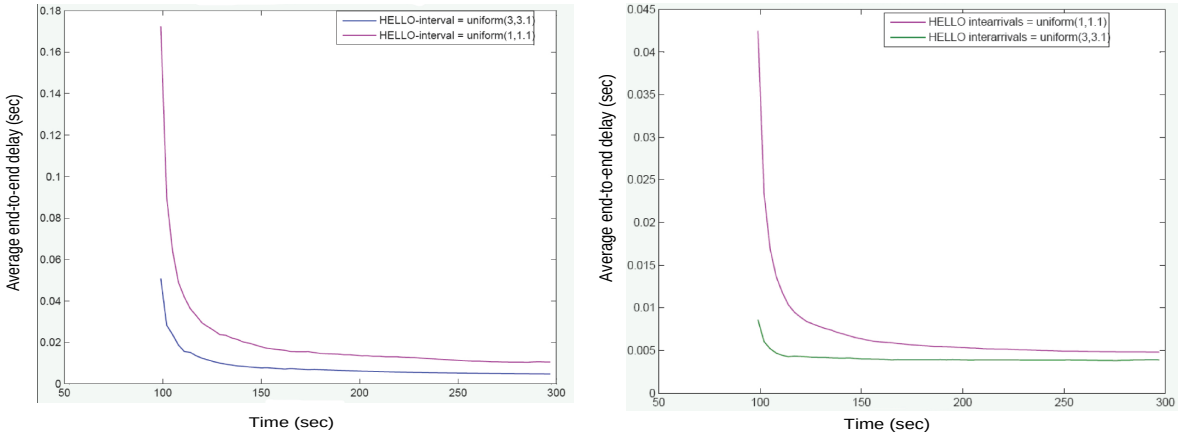


Figure 3.13: AODV HELLO packet format [82]

1. Type = 2
2. R = repair flag (used if local repair is enabled ; broadcast fashion)

3. A = acknowledgment required flag is set in order to ask for RREP acknowledgment .This ACK is necessary to ensure route discovery cycle completion if there are some unidirectional links exist
4. Destination IP address = sets to nodes IP address
5. Destination sequence number = sets to latest sequence number
6. Hop count = zero
7. Lifetime = Allowed HELLO loss × HELLO interval time = $2 \times uniform(1, 1.1)$

As we see from Figure 3.15, the number of bits in HELLO packet = $5 \times 32 = 160$ bits. During the first run simulation (data rate = 1 packet/sec), we noticed that HELLO packets generated with default inter arrival time (uniform(1,1.1)) caused many MAC layer collisions .This caused the source to initiate 38 RREQ packets in total during simulation time upon wrong assumption that default route was not valid. Afterwards we changed the parameter HELLO interval time to uniform(3, 3.1). This action led to decrease in number of RREQs to only nine packets in total. This shows effect of HELLO packets on the performance. For this scenario we like mention that local repair property for AODV is disabled. That means upon link failure (intermediate node sends RERR to source) the only node replies by RREP message is destination node. From previous observation we could conclude that, although channel capacity was high enough (11 M bps) for such low data rate but local connectivity monitoring technique which is used with AODV doesn't allow for exploiting network resources and caused collisions. This in turn led to several route discoveries and longer delay as illustrated in Figure 3.16. We can see from Figure 3.16 how increasing of HELLO



(a) data rate = 1 packet/sec

(b) data rate = 10 packets/sec

Figure 3.14: Effect of inter arrival time of HELLO packets on delay performance

interval can decrease end-to-end delay, but care must be taken because this will affect the performance dramatically in dynamic scenario. As less inter-arrival time between successive HELLO packets will decrease the resolution of local connectivity monitoring. Also the same figure illustrates how end-to-end delay is decreased with increasing data

rate. This happened because data packets will keep the medium busy longer time, so it wins the MAC layer contention with HELLO packets. Also with increasing data rate still longer inter-arrivals of HELLO packets has better performance. From here we mention how useful it is to consider cross layer design for routing protocols. Now we continue our work with energy consumption investigation. Parameters to be collected in OPNET were as follow:

1. **WLAN data traffic sent**: these are packets come from network layer which are encapsulated by adding a 14 byte header (Protocol Control Information (PCI))before the data and appending a 4-byte (32-bit)CRC after the data. The payload data field could include the following values:
   - AODV routing protocol overhead:
     - $RREQpackets = 192$ bits/packet
     - $RREPpackets = 160$ bits/packet
     - $RERRpackets = 160$ bits/packet
     - $HELLOpackets = 160$ bits/packet
   - Data packets = 1024 bits/packet
   - Data retransmissions due to no ACK received
2. **WLAN data traffic received**: which are data packets received from physical layer and has the same format
3. **WLAN control traffic sent/received**: which take the following format:
   - $RTSpacket = 20 * 8 = 160$ bits
   - $CTSpacket = 14 * 8 = 112$ bits
   - $ACKpacket = 14 * 8 = 112$ bits

Using energy model introduced in previous part, we were able to calculate energy consumed per packet (TX /Rx) as shown in Table 3.11. Also Tables (3.12,3.13) illustrate the AODV routing protocol and MAC parameters (default implementation in OPNET 14.5). In the First run (1packet/sec) we observed that there were five RERR messages sent (link failure detection due to collisions with HELLO packets), and packet delivery ratio was 100%. MAC layer could send undelivered packets successfully after performing retransmissions. Total aggregated energy consumed during this run was 0.261473264 joule. Table 3.14 summarizes the outcomes of simulation runs. Notice that, in third run (100 packets/sec) PDR has been dropped to 99.76% with total amount of 1151 retransmissions occurred from both source and intermediate nodes with total number of six RERR messages sent during the simulation time. MAC layer drop packets which were stored in the buffer after maximum retransmissions attempts threshold (7 in this scenario). As we will see in the 4th and 5th runs the amount of dropped packets will get larger which will decrease PDR. In next part we will introduce DSR performance.

Table 3.11: Energy consumed per packet in AODV case

MAC packet type	Energy consumed per packet transmission(n joule)	Energy consumed per packet reception(n joule)
Data packet with raw data as payload = 1168 bits	90*1168 = 105120	50*1168 = 58400
Data packet with routing overhead as payload = 304 bits	90*304 = 27360	50*304 = 15200
Data packet with RREQ as payload = 336 bits	90*336 = 30240	50*336 = 16800
Control packet(ACK/CTS) = 112 bits	90*112 = 10080	50*112 = 5600
Control packet(RTS) = 160 bits	90*160 = 14400	50*160 = 8000

Table 3.12: Transmission parameters

MAC parameters	Value
Transmit power	2E-005 watt
Packet reception threshold	-95 db
Retry limit	7
Data rate	11 Mb/s
Buffer size	256 Kbits

Table 3.13: AODV routing protocol parameters

AODV parameters	Value
Active route time out	3 sec.
Hello intervals	Uniform(1,1.1)
Timeout buffer	2 sec.
Packet queue size	infinity
Local repair	Enabled
TTL (start/increment/Threshold)	1/2/7 hops
Node traversal time	0.04 sec.

Table 3.14: AODV performance

AODV runs	Run 1	Run 2	Run 3	Run 4	Run 5
PDR %	100	100	99.76	99.55	58.73
Delay (sec)	0.00400	0.00170	0.00860	0.02290	1.36690
Energy consumed (joule)	0.26	0.92	7.57	23.81	35.79

3.3.4 DSR routing protocol

Because DSR use source routing mechanism, packets will have different format than in AODV case. Here a special header which caries some control information will be included in any existing IP packet. This header is called option header (4 octets) and follows the IP header immediately as shown in Figure 3.17. From [86] we illustrate

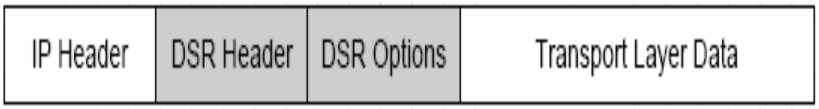


Figure 3.15: Data packet header with DSR protocol

different DSR packet sizes as follow:

- DSR header = 32 bits
- RREQ option = 64 + 32×(number of traversed hops)
- RREP option = 32 + 32×(number of hops)
- RERR option = 128 bits
- ACK request (attached with each data packet) = 32 bits
- Source route option (attached with data packets to define the route) = 32 + 32×(number of hops)

We illustrate in Table 3.15 the total size of MAC layer packets which will be transmitted /received and related consumed energy.

Table 3.15: Energy consumed per packet in DSR case

MAC packet type	Energy consumed per transmission (n Joule)	Energy consumed per reception (n Joule)
Data segment = 1328 bits	110880	61600
Routing overhead = 240 bits	21600	12000
RREQ overhead = 528 bits	47520	26400
Control packet(ACK/CTS) = 112 bits	10080	5600
Control packet(RTS) = 160 bits	14400	8000

In this scenario we have noticed that for different simulation runs, the data was transmitted through two hop route. Also we like to mention that, DSR protocol gets feedback from MAC layer (ACK) for data acknowledgment. But also it has ACK option field which can be used if no ACK is implemented in MAC layer which we have disabled in this scenario. Further, we illustrate DSR parameters which have been set for this simulation in Table 3.16. Using the same method of calculating energy consumption for AODV case, we found in first run (1 packet/sec) PDR was 99.5% which is due to one

Table 3.16: DSR routing protocol parameters

DSR parameters	Value
Max. cached routes	Infinity
Route expire time	300 sec
Send buffer size	Infinity
Send buffer expiry	30 sec
ACK time	0.5 sec
Packet salvaging	Enabled
Routes replies using cashed routes	Enabled
Max. maintenance retransmissions(to confirm neighbor reach ability)	2 times

Table 3.17: DSR performance

DSR runs	Run 1	Run 2	Run 3	Run 4	Run 5
PDR %	99.9500	99.7000	100	100	12.3660
Delay (sec)	0.0092	0.0026	0.0014	0.0015	18.6112
Energy consumed (joule)	0.0833	0.7179	6.93	20.7410	27.9650

packet drop out of 200 packets from source node. This happened because the first route had longer link ($source \rightarrow node15$) which caused several retransmissions and caused the loss of one data packet on time stamp 106 sec. DSR reacts by switching to second cached route ($source \rightarrow node7$) which has shorter link and enabled stable transmission till end of simulation time. Table 3.17 illustrates the results In next section we will present the behavior of OLSR routing protocol with respect to energy consumption and after that we will summarize the differences between the three protocols in last section.

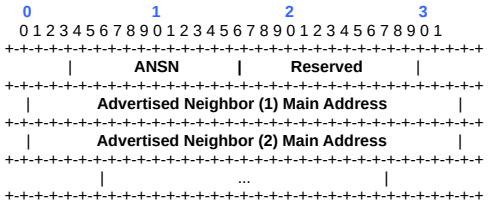
3.3.5 OLSR routing protocol

In this part we will investigate energy consumption for OLSR protocol which has shown superior efficiency in terms of delay and routing overhead for mobile scenario comparing with reactive protocols like AODV and DSR (previous part). First we illustrate OLSR protocol parameters in Table 3.18. The term willingness indicates if a node accept to forward traffic for other nodes in the network (MPR functionality), and it depends on its capability (battery level, power, capacity). For duplicated received messages, there are special table that holds these messages to prevent unnecessary processing.

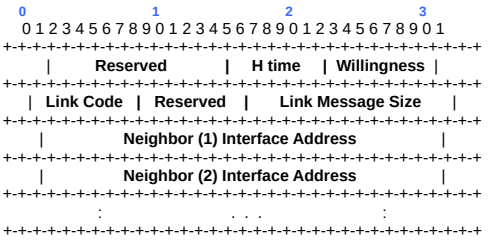
Table 3.18: OLSR routing protocol parameters

OLSR parameters	value
HELLO interval	2 sec.
TC interval	5 sec.
Neighbor hold time	6 sec.
Topology hold time	15 sec.
Duplicate message hold time	30 sec.
Willingness	Default (medium)

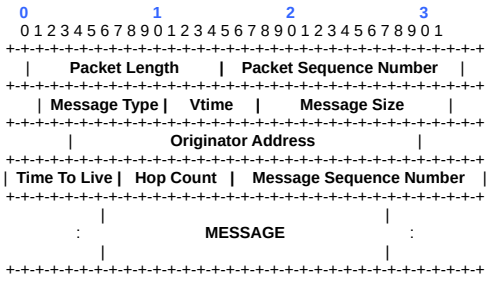
Messages that are hold for (30 sec) in this table will be discarded. Packets in OLSR have the same format [87] and are transmitted using UDP protocol via port number 698 by default. Figure 3.25 illustrates general packet format used for OLSR protocol, and obviously different message types have different data portion sizes. There are two kinds of messages have not been generated in this scenario which are Multiple Interface Declaration (MID) and Host and Network Association (HNA) messages. First message is sent by the host to declare to other nodes in the network that it posses multiple interfaces, while the second message is used by gateway nodes to broadcast information about associated hosts and networks (here we consider only intra-routing). The rest of used messages are HELLO (link sensing, MPR set calculation and neighborhood detection) and TC (MPR selector set addresses; which is broadcasted by MPR nodes to help other nodes building their routing table).



(a) TC message packet format



(b) HELLO message packet format



(c) data packet format

Figure 3.16: OLSR different packets format[87]

After exporting routing tables for all nodes and collecting results, it has been observed that:

- Number of hops for all runs = 2
- Average number of neighbors for node = 6
- Average number of MPRs = 6

Table 3.19 summarizes different MAC packet sizes and energy consumed per packet for OLSR scenario which is further have been used to calculate energy consumption (see Table 3.20) . Notice that routing overhead for all runs was nearly constant and the increase of energy consumption was due to increase in data rate.

Table 3.19: Energy consumed per packet in OLSR case

MAC packet type	Energy consumed for transmission(n joule)	Energy consumed for reception(n joule)
Data segment = 1168 bits	105120	58400
Hello message = 528 bits	47520	26400
TC message = 496 bits	44640	24800
Control packet(ACK/CTS) = 112 bits	10080	5600
Control packet(RTS) = 160 bits	14400	8000

Table 3.20: OLSR performance

OLSR runs	Run 1	Run 2	Run 3	Run 4	Run 5
PDR %	98	88.65	80.65	94.89	64.26
Delay (sec)	0.00200	0.00400	1.50000	0.55000	1.20000
Energy consumption (Joule)	1.77330	2.66020	10.67000	23.21500	39.67

3.4 Summary

We concluded that, to enable dependable intra-routing in PN, OLSR routing mechanism is recommended. And because low power consumption is an important factor to achieve dependability (regarding network connectivity as mentioned previously). We have investigated the performance of OLSR compared with both AODV and DSR with respect to power consumption criterion. As we can see from Figure 3.19 that reactive protocols (DSR, AODV) perform better than proactive (OLSR)protocols with respect to energy consumption. Although we have introduced only fixed scenario in this chapter for simplicity reasons, but DSR has also shown superior performance in mobile scenario [89,90] comparing to other protocols. In [90] it has been shown that the amount of knowledge about network topology has a great effect on power consumption. Also operating in promiscuous mode allow DSR to gain more information on network topology and helped for less energy consumption. As a general conclusion we say that:

- Design components which are forming the structure of OLSR protocol have better interoperability performance than other protocols regarding PN scenario conditions as it could achieve higher throughput (around 2,7 M bps in average) and low delay

- Investigation on some techniques is further needed to enhance the performance of OLSR protocol with respect to PDR and energy consumption

- As there are many efforts to enhance the performance of batteries for mobile devices [91], the cost of power consumption could be further negligible.

- Because OLSR protocol is able to act with multiple interfaces, unidirectional links and different address resolutions it is recommended for PN.

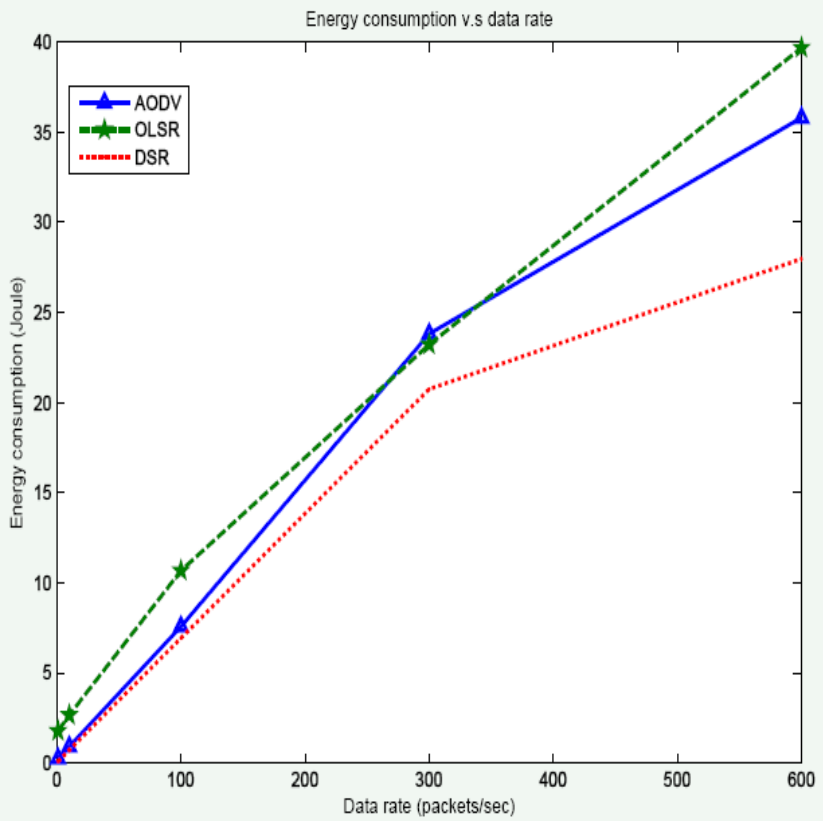


Figure 3.17: Effect of increasing data rate on power consumption for different routing protocols

Table 3.21: Suggested component design for dependable OLSR intra-routing protocol for PN cluster

Routing component	**Suggested value**
Route discovery	Multiple disjoint paths with maximum number of three
Route selection	Default(link state)
Route metric	Hop count should be replaced by an efficient link assessment metric. This metric must satisfy QoS requirements for each individual application [71].
Route maintenance	Default (MPR mechanism)
Route representation	Default (routing table)
Data forwarding	Forwarding packets through available paths (weighted round robin) [58,70]

Here also we can present our suggested design for dependable OLSR routing protocol for PN. We use the output of chapter two to introduce a component based design which

is illustrated in Table 3.21. After introducing our suggested design for dependable routing protocol in PN, we will introduce in next chapter the crucial impact of MAC layer on network performance. As we will see later, MAC layer has a great influence on network performance as it determines the throughput and delay of the traffic. Huge effort could be invested to enhance a single routing protocol mechanism could deliver slice improvement to overall network performance. While a remarkable improvement could simply achieved by deploying routing protocol above a proper MAC layer.

The effect of MAC layer on the performance

4

In previous chapter we noticed the effect of MAC layer on packet delay. Because MAC layer is positioned at the bottom of OSI layer stack, we expected that it should have a remarkable effect on the overall performance of the network. Our measurement metrics for performance were throughput and delay at MAC level. Throughput presents the number of bits (bit/sec) which is successfully received and forwarded by MAC layer to higher layer. And delay presents total delay (contention, management delay, queuing ,etc.) encounters a packet before accessing the medium. In this chapter we didn't consider routing overhead as a performance metric, because MAC layer doesn't differentiate between overhead and data packets in packet processing. We have studied the performance of OLSR routing protocol with different MAC/Physical layers comparing with DSR protocol. Bellow we point out what has been introduced in this chapter:

- How longer delay caused by route discovery process in reactive protocols will affects the performance.
- Whether high throughput guarantees high PDR performance in the network.
- Interaction between MAC and network layer determines the overall performance.

4.1 Throughput of IEEE 802.11

It has been shown that [92], actual throughput that a user can get using 802.11 wireless technologies is significantly smaller than the advertised radio throughput for 802.11 products. By experimenting 802.11b with 11 M bps basic data rate and MAC layer service data unit (MSDU) packet size of 1500 bytes, a throughputs of 4.52 M bps for RTS/CTS scheme and 6.06 M bps for CSMA/CA scheme is achieved. The later scheme has less control packets, thus it achieves higher data throughput. This sounds reasonable because for OSI layering stack, the higher the layer the lower the throughput for that layer because of accumulated overhead that each layer will have to add. The maximum throughput of an application is greatly influenced by both transmission (TCP/UDP) layer and link (MAC) layer protocols. Considering some assumption, it has been shown that the theoretical maximum throughput (TMT) for both application (for application is also called good put) and MAC layer are correlated to each others by the following formula [92].

$$TMT_{APP} = \left[\frac{\beta}{\alpha + \beta}\right] \times TMT_{MAC}\ (bps)\ , \tag{4.1}$$

Where β is the application payload size,and α is total overhead of MAC layer. The above relation assumes that (BER=0, ad-hoc operation mode, no fragmentation, no

collisions, management frames are not considered). Notice that, from Formula (4.1) the throughput of application is always less than the throughput of MAC layer. One of the ways used to calculate the throughput of MAC itself is by dividing MAC service data unit (MSDU) size (bits) by the time taken to transmit (sec) as follow:

$$TMT_{MAC} = \frac{MSDU}{T_{total}} \ (bps) \ , \tag{4.2}$$

Where,

$$T_{total} = T_{DIFS} + T_{SIFS} + T_{RTS} + T_{CTS} + T_{ACK} + T_{DATA} + T_{BO} \ (\mu sec)$$

These delay components are defined as follow:

- T_{DIFS} is DCF inter-frame space time
- T_{SIFS} is short inter-frame space time
- T_{RTS} is the time taken to successfully send a RTS packet
- T_{CTS} is the time taken to successfuly receive CTS packet
- T_{ACK} is the time taken to receive an acknowladgement over the previously sent data packet
- T_{DATA} is the time duration to send a single data packet
- T_{BO} is the back off window (contention) time

This delay is caused by timing diagram of MAC layer. Different MAC layer mechanisms (RTS/CTS and CSMA/CA) co-working with different physical layer technologies (DSSS, FHSS, OFDM) will cause different values for this delay (see [92].table.1) . Thus the term T_{total} is dependent on wireless technology used and not on data rate or routing protocol in use. In other words, MAC layer determines the average end throughput of PN. For example, routing protocol could assign routes for traffic efficiently or it might have high performance route maintenance mechanism, but an argent packet (e.g. health care application) will have to wait longer in the queue due to large back offs or several collisions. Also from [93] we see that for RTS/CTS case the larger the MSDU is the higher the TMT, but it has proven that [94] for CSMA/CA case, the probability of collision doesn't depend on packet length, but on contention window size and number of source nodes. Increasing number of source nodes will increase the probability that the medium is busy and thus increase medium access delay for all source nodes. This in turn will increase end-to-end application delay. Also it has been shown that, to get maximum throughput, contention window size should be proportional to square root of packet size [94]. In this chapter we were able to investigate the effect of variable contention window size by using different type of MAC layer (802.11 type) in our simulation . Also we will show the effect of varying data packet size (which in turn will vary MSDU size) and number of source nodes on the performance.

4.2 Simulation scenario

We use OPNET 14.5 simulation environment with an area of 150x150 square meters and 50 mobile nodes. Random way point mobility is used with speed uniformly distributed between (0,4) meter/sec and 100 sec pause time. The simulation will run for 5 minutes (300 seconds). CBR traffic pattern is used with 4 packets/sec as packet rate. Source node starts to send traffic after 100 seconds from beginning of simulation to a randomly chosen destination node. Notice that we measure global behavior of the network, that means the following:

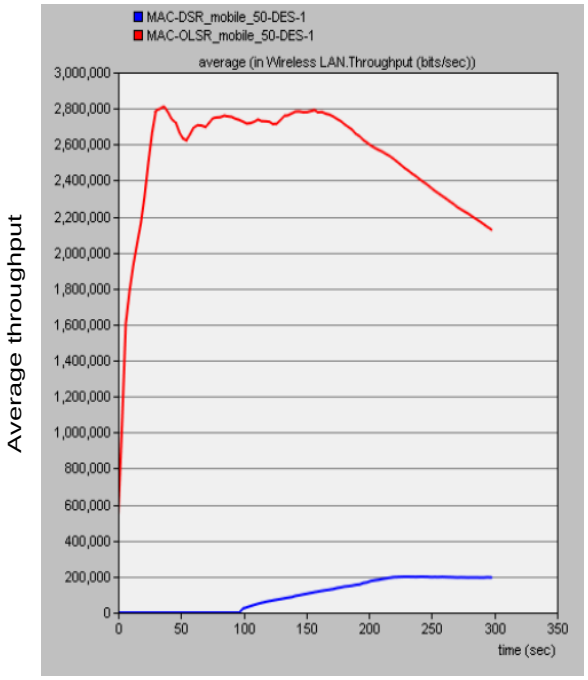
- Measured throughput represents the average bit rate (bits/sec) where bits are forwarded from wireless LAN layers to higher layers in all WLAN nodes of the network during simulation time.
- Measured delay represents the global statistic for the total of queuing and contention delays of the data, management, delayed Block-ACK and Block-ACK Request frames transmitted by all WLAN MACs in the network. For each frame, this delay is calculated as the duration from the time when it is inserted into the transmission queue until the time when the frame is sent to the physical layer for the first time. Hence, it also includes the period for the successful RTS/CTS exchange, if this exchange is used prior to the transmission of that frame. Similarly, it may also include multiple number of back off periods, if the MAC is 802.11e-capable and the initial transmission of the frame is delayed due to one or more internal collisions.

Generally in all presented result curves, DSR is presented in blue color and OLSR in red color. Also, vertical line always represents time in seconds and vertical line represents the collected statistic which is described on the header of each graph. Also we have repeated simulations several times to get sufficient confidence on the results.

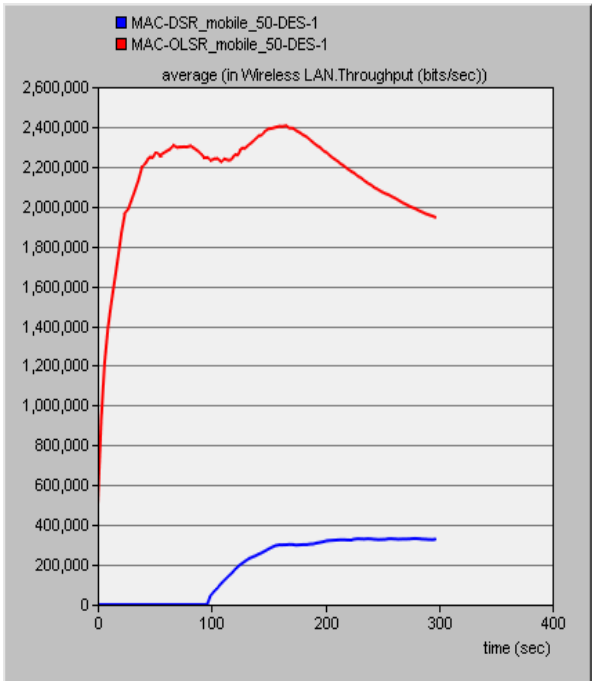
4.2.1 Varying number of source nodes

In this scenario CSMA/CA type of MAC layer (with PHY = IEEE 802.11b) is used and we will investigate the performance under varied number of source nodes (overall traffic load on the network) . Also CBR traffic pattern is used for all sources with 512 bytes per packet and rate of 4 packets/sec. Figures(4.1,4.2) illustrate average throughput and total delay for the network with different source nodes.

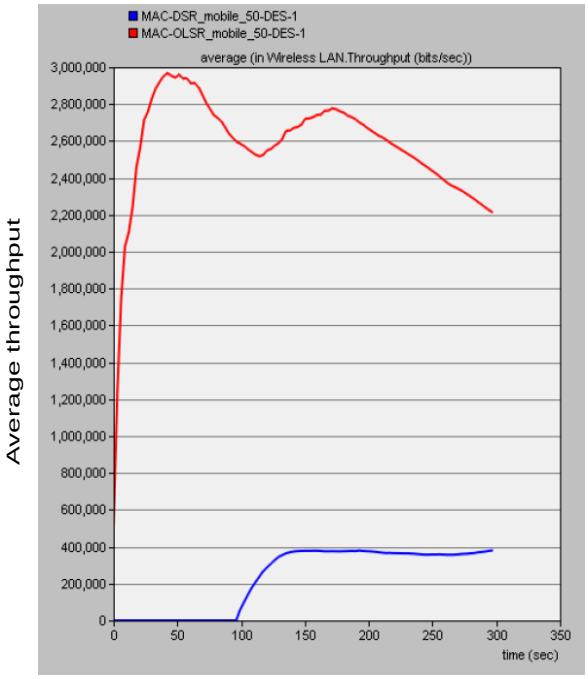
We see large difference in performance between OLSR and DSR protocols. OLSR outperformed DSR in all different cases regarding both throughput and delay. By increasing number of source nodes, the probability of collision will be higher and there will be more back off times. We can see this from Figure 4.2, as delay is increasing by adding more number of source nodes for both protocols. DSR has recorded remarkable delay comparing with OLSR. We refer this delay to reactive nature of DSR protocol as time to discover new routes has an effect on delay performance especially in mobile scenario where route discovery process will be recalled many times. This rises a need to study the effect of route discovery time in DSR protocol, which will be presented next section. OLSR routing protocol with proactive nature allows for data packets to



(a) 10 source nodes



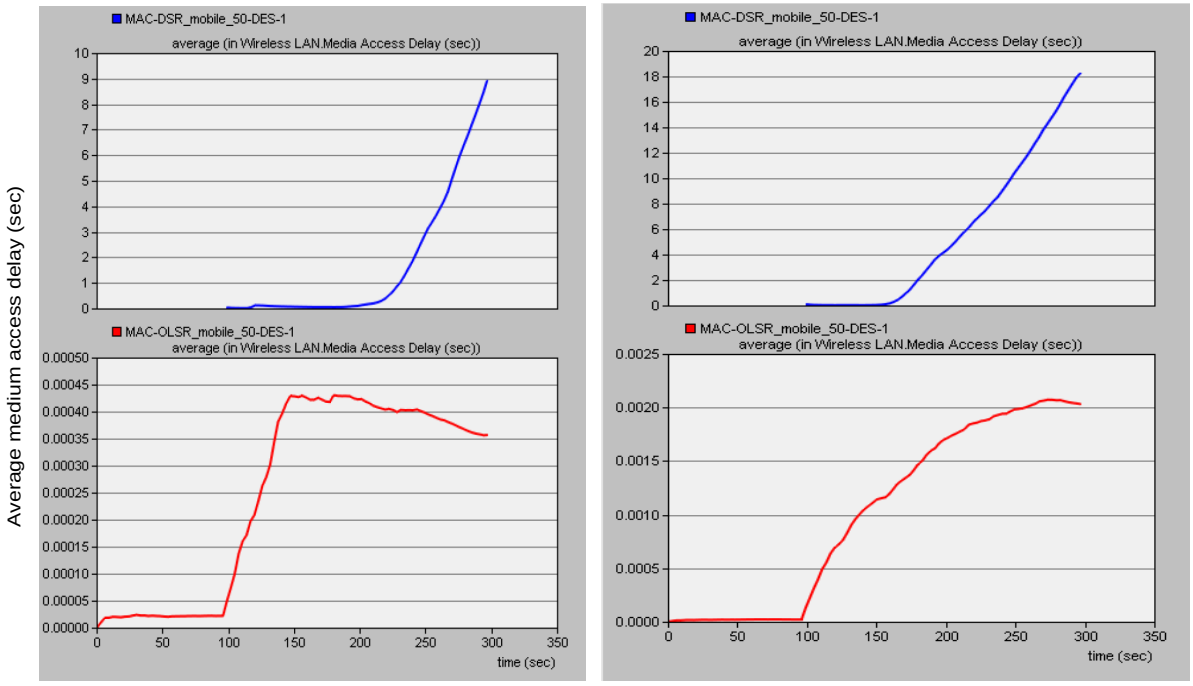
(b) 20 source nodes



(c) 30 source nodes

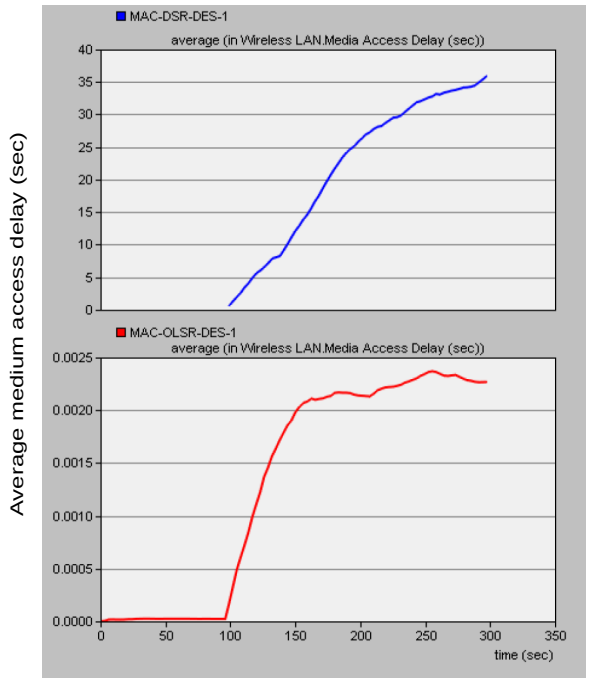
Figure 4.1: Average throughput performance under different number of source nodes

be sent directly when medium is not busy, which will decrease the total medium access delay substantially comparing to DSR. Also notice that , OLSR generates constant



(a) 10 source nodes

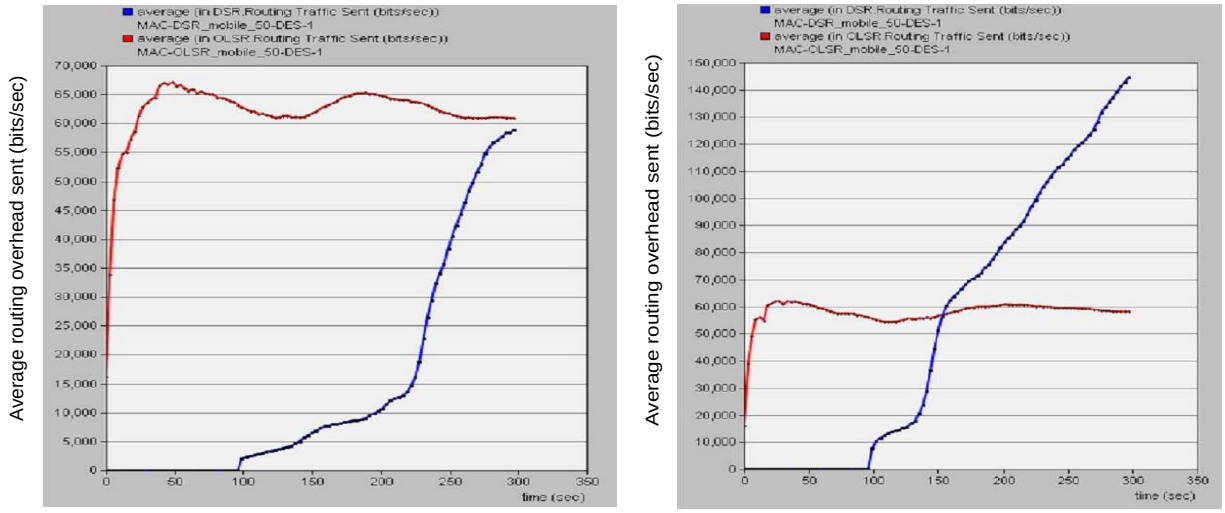
(b) 20 source nodes



(c) 30 source nodes

Figure 4.2: Average delay performance under different number of source nodes

routing overhead in counter to DSR which generates more overhead with increasing number of source nodes in mobile scenario. In Figure 4.3 we illustrate routing overhead



(a) 10 source nodes

(b) 30 source nodes

Figure 4.3: Average routing overhead for both protocols *bits/sec* under different number of source nodes

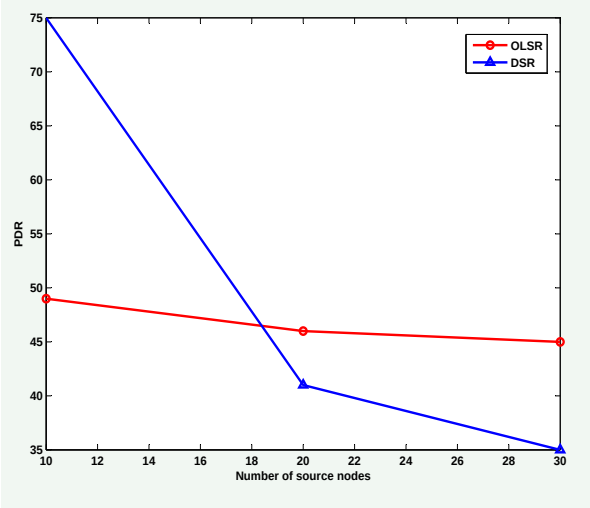


Figure 4.4: PDR performance for both routing protocols with increasing number of source nodes

(bits/sec) generated from both DSR and OLSR protocols for both 10 and 30 number of source nodes. Notice that from Figures(4.1,4.3) that, DSR generates more overhead (bits/sec) than OLSR for 30 source node case with less throughput. Larger amount of overhead and longer delay will have negative effect on PDR performance. Figure 4.4 shows that by increasing number of source nodes, OLSR has more stable PDR performance than DSR. In general, PDR for both protocols is decreasing by increasing number of source nodes but DSR curve as we see has larger slope. That means for PN scenario with DSR protocol, when some PN clusters are merged or split, there could be a sudden sharp change on quality of running applications. And even worse, there

Table 4.1: PDR performance for different route cache implementations

PDR%	All routes	3 routes
50 seconds route expiry time	48.21	29.4
300 seconds route expiry time	46.52	27.25

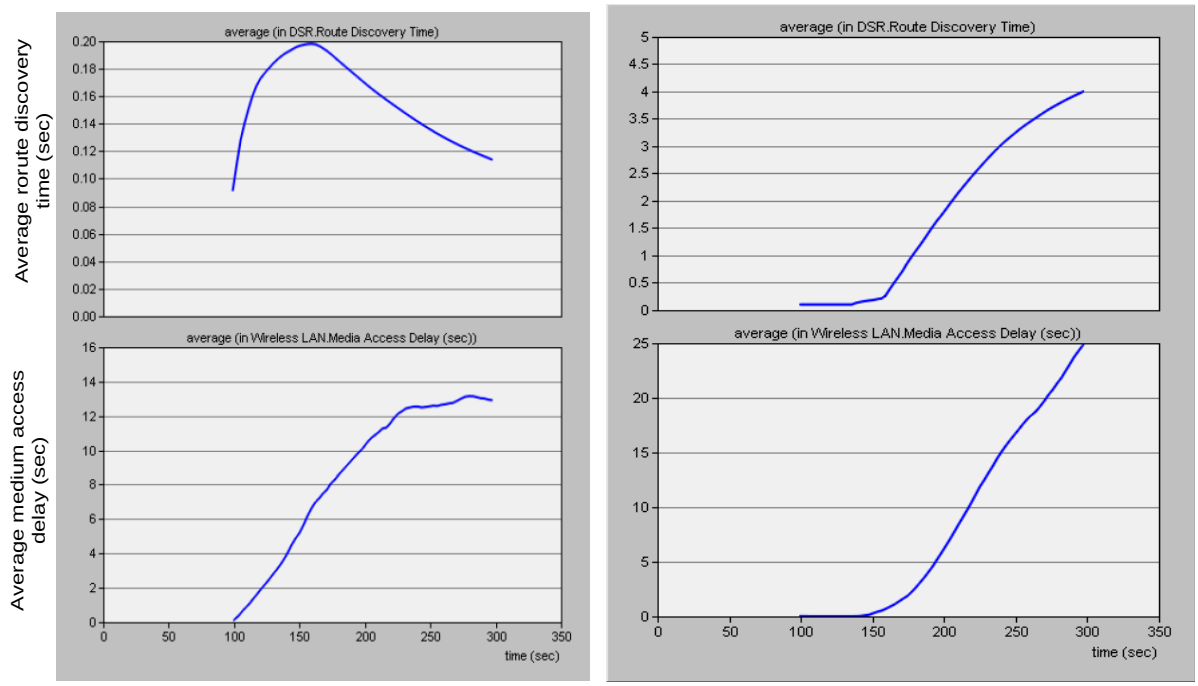
might be discontinuity for some service application running on PN node.

4.2.2 Effect of route discovery time on delay performance

In this scenario there will be 10 source nodes with 512 byte data packet size. Further we will discuss the effect of route discovery time for reactive protocols (DSR) on delay performance. We will change allowable number of cached routes and route expiry time (the expiry time of routes after it has installed in the cache). Figure 4.5 illustrates route discovery time and medium access delay for different number of cached routes and route expiry times respectively. First we see that by increasing route discovery time, medium access delay is also increasing. And this explains the reason behind larger delay which has been recorded for DSR in previous section. Second observation we notice from Figure 4.5 that number of allowable routes to be stored in the cache has larger effect on delay performance of reactive protocols(DSR). Remember from chapter two, it was mentioned that, for disjoint multipath implementation of DSR number of multiple paths should be small. Because this could cause performance degradation, and this what we have also experienced in this part as we saw larger delay for larger number of multiple routes. Table 4.1 shows PDR performance for each case. From Table 4.1 and Figures(4.5,4.6) we see that, the more number of multiple routes the higher the PDR performance of the protocol but with the cost of higher delay. Also we see that higher throughput doesn't guarantee higher PDR. From this we conclude that PDR performance depends mainly on routing protocol and not on MAC layer. Higher throughput could allow for smooth behavior for routing protocol under variation of conditions (as we saw in previous section). DSR has the advantage of multiple cached routes over OLSR protocol. Which gives him superior PDR performance. This result agrees with our conclusion in Chapter 3.

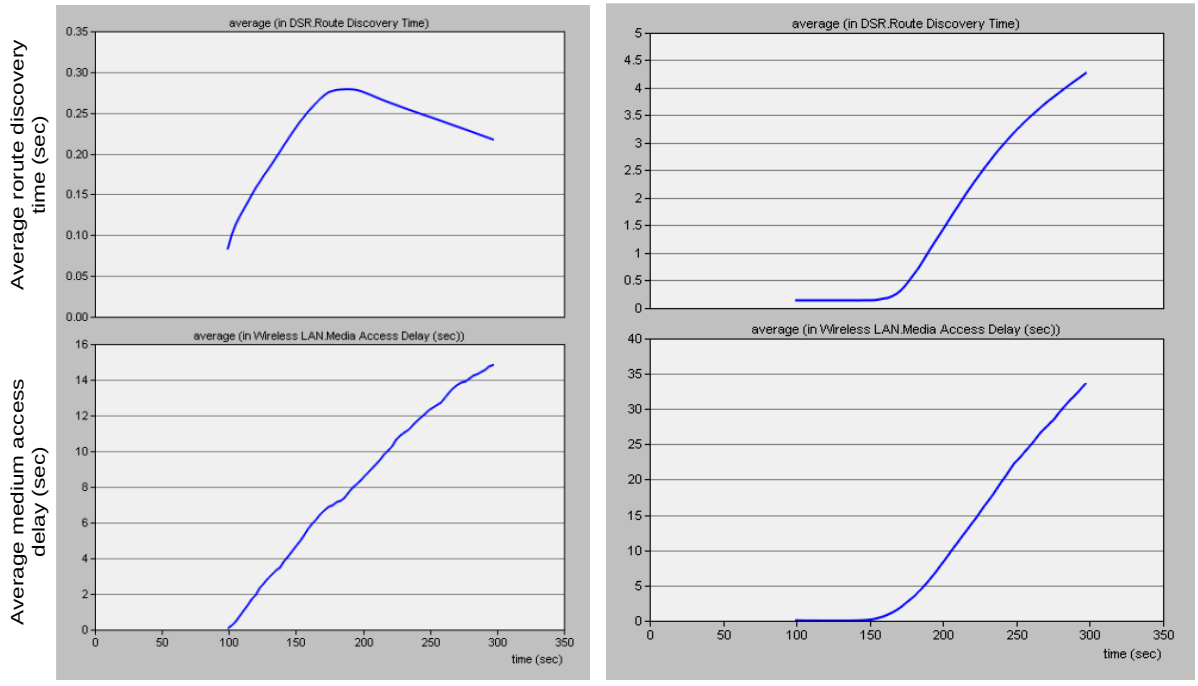
4.2.3 Variable data rate

In this part we investigate the efficiency of both OLSR and DSR routing protocols working above MAC layer with different data rates (1 M bps and 5 M bps). We notice that, average throughput for both protocols was approximately constant with increasing data rate. In average MAC layer has allowed throughput of 2,4 M bps and 0.3 M bps for OLSR and DSR protocol respectively. Which is in most case less than 21.8% of the advertised throughput (11 M bps). OLSR protocol mechanism could gain more throughput from MAC than DSR as the same happened in previous part (different number of source nodes). For OLSR we notice remarkable increase in delay by increasing data rate, while in DSR delay performance didn't change so much. Long delay caused by reactive nature of DSR protocol makes the difference in overall delay performance negligible. From this part we saw that, although less delay could allow



(a) 3 routes with 50 sec expiry time

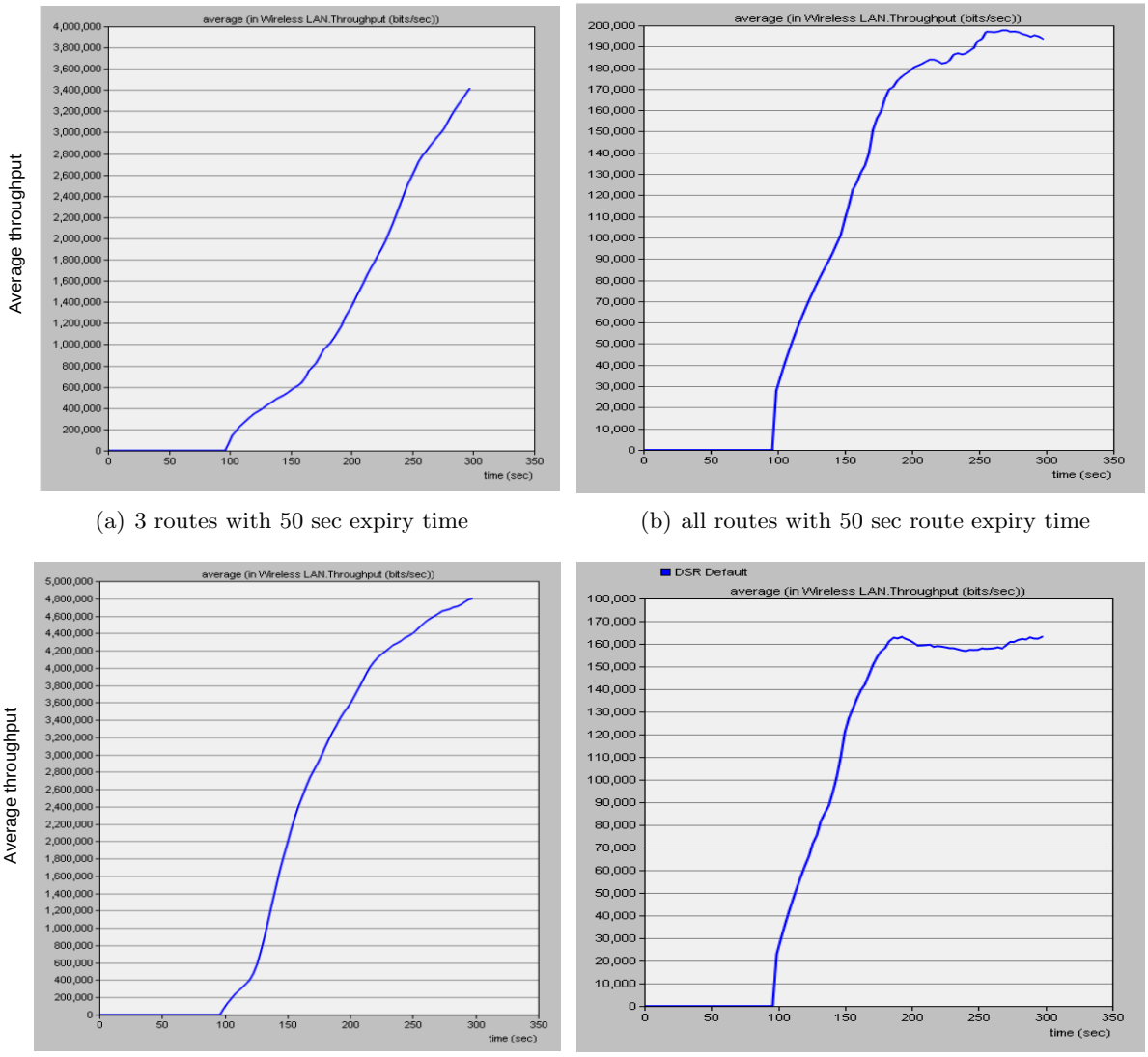
(b) all routes with 50 sec route expiry time



(c) 3 routes with 300 sec expiry time

(d) all routes with 300 sec expiry time

Figure 4.5: DSR delay performance with different cache route implementations



(a) 3 routes with 50 sec expiry time

(b) all routes with 50 sec route expiry time

(c) 3 routes with 300 sec expiry time

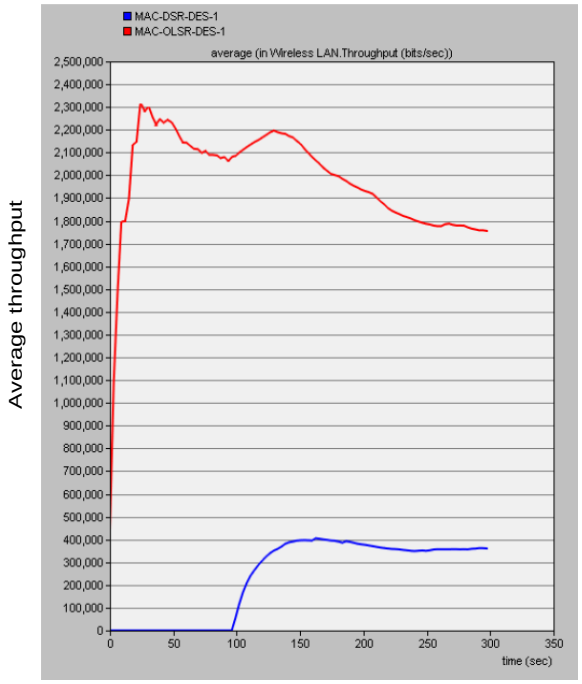
(d) all routes with 300 sec expiry time

Figure 4.6: DSR throughput performance with different cache route implementations

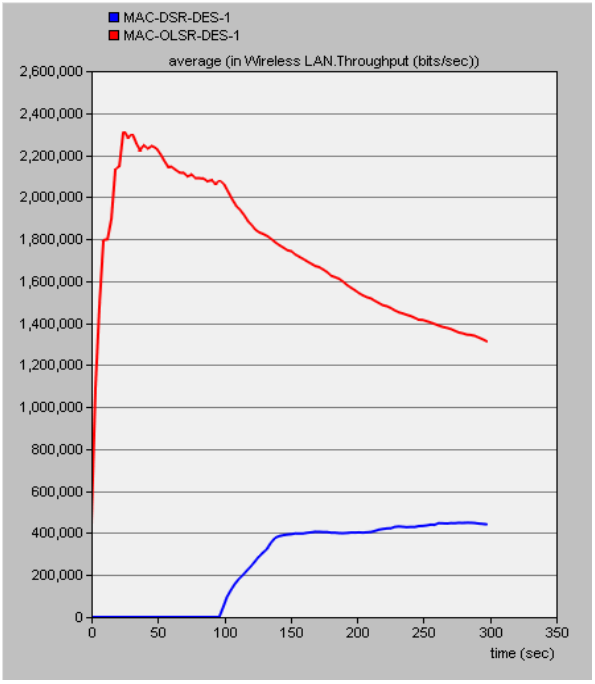
for more throughput but still a more efficient MAC layer is needed to exploit the data rate allowed by underlied physical layer.

4.2.4 Varying MAC layer

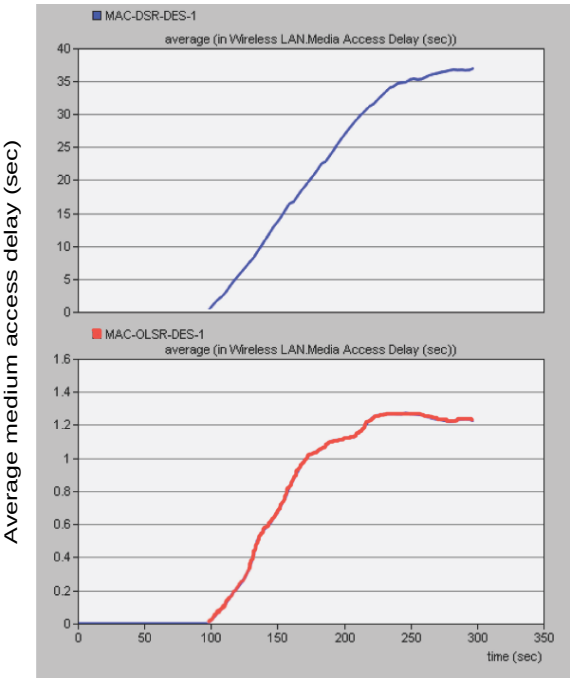
In this part we will investigate the performance of routing protocols under different MAC layer types. In our scenario there will be 10 source nodes from total number of 50 nodes generating CBR data traffic with 4 packets/sec and 512 bytes per packet size. Two modes for MAC layer will be investigated (CSMA/CA, RTS/CTS) under OLSR and DSR routing protocols. Figure 4.8 illustrates the throughput and delay performance for both protocols respectively. RTS/CTS mode has slightly larger throughput due to more packets transmitted (MAC overhead) and this comes with cost of longer



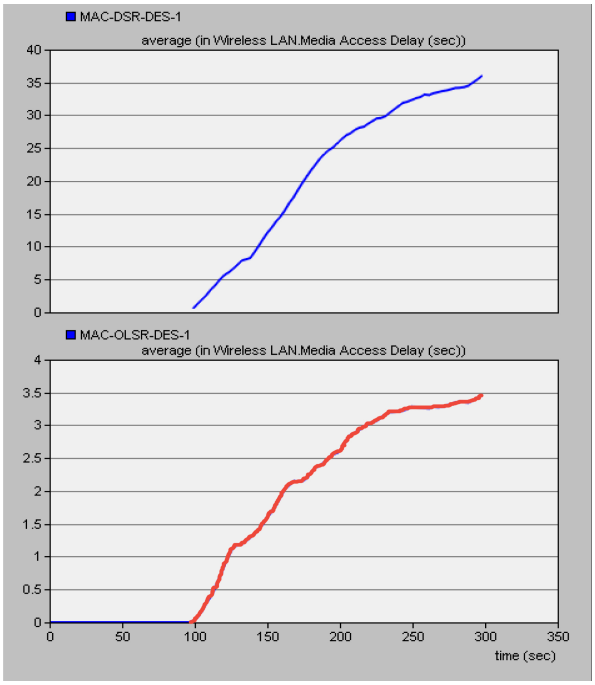
(a) Throughput performance under 1 M bps data rate



(b) Throughput performance under 5 M bps data rate

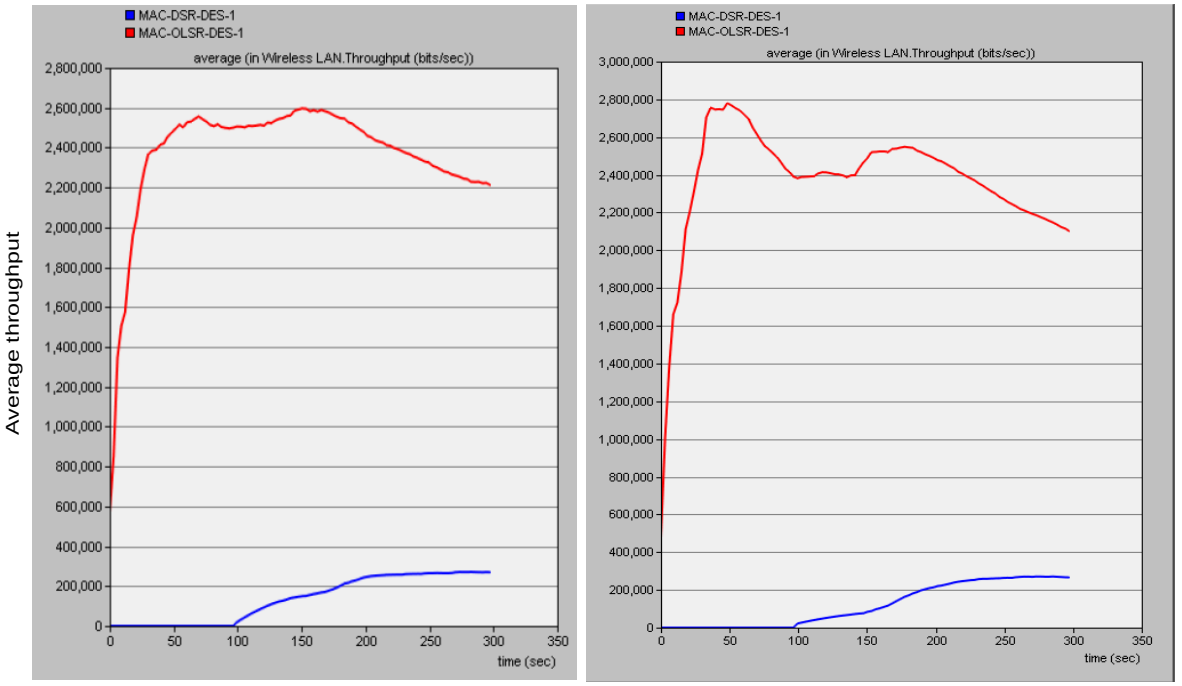


(c) Delay performance under 1 M bps data rate



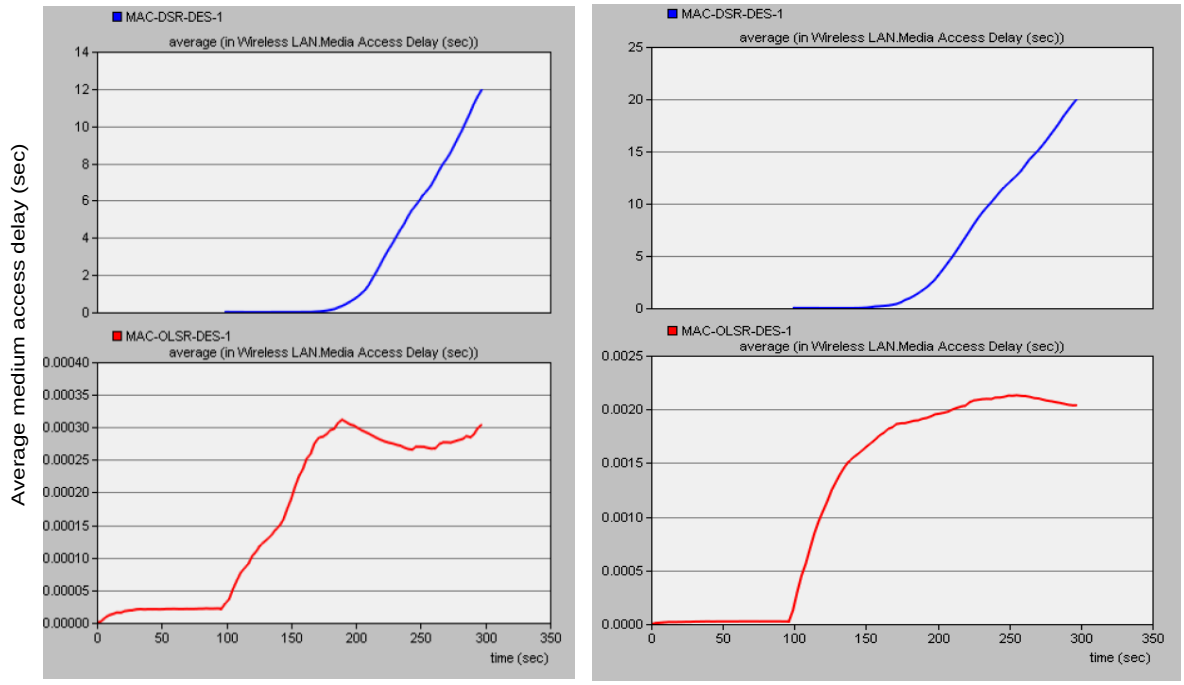
(d) Delay performance under 5 M bps data rate

Figure 4.7: Performance For both protocols with different data rates



(a) CSMA/CA throughput performance

(b) RTS/CTS throughput performance



(c) CSMA/CA delay performance

(d) RTS/CTS delay performance

Figure 4.8: Performance For both protocols with different MAC layer types

Table 4.2: PDR performance for both protocols with different MAC layers

	CSMA/RTS/CTS	CSMA/CA
OLSR	39.36	60.31
DSR	41.4	61.21

Table 4.3: Parameters for different IEEE 802.11 physical layer types

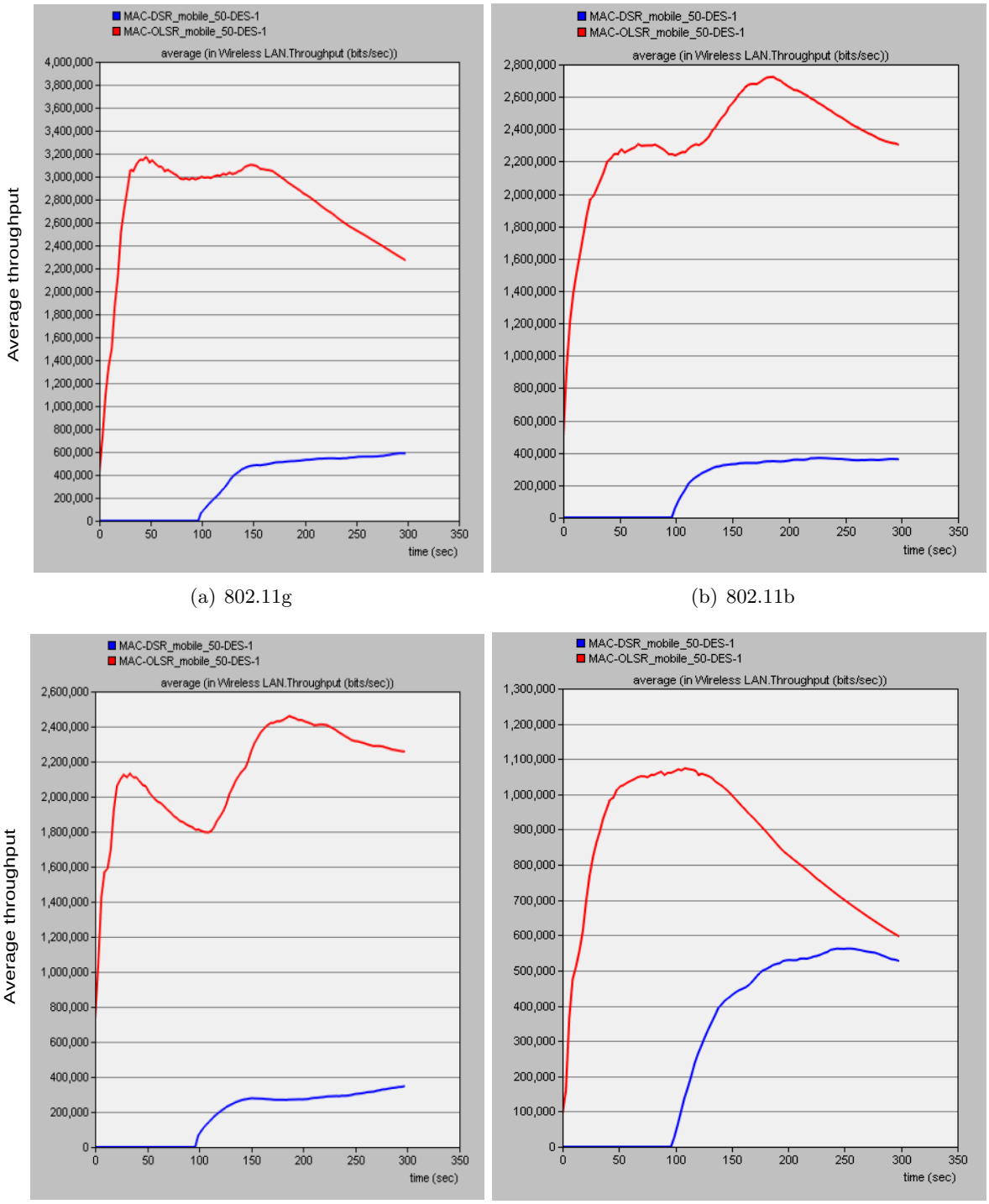
	802.11	802.11a	802.11b	802.11g
Operation frequency	2.4-2.4835 GHz	5.15-5.35 GHz	2.4-2.4835 GHz	2.4-2.4835 GHz
Modulation scheme	FHSS/FSK	OFDM/PSK	DSSS/PSK	OFDM/PSK
data rate per channel	2 M bps	12 M bps	11 M bps	11 M bps
Band width	83.5 MHz	300 MHz	83.5 MHz	83.5 MHz
Range (meters)	20	50	68	68

delay. DSR as we also saw from previous parts has worst delay performance due to reactive nature (added route discovery time to delay). Notice that,(Formula 4.2) measures maximum throughput per MSDU unit, but we measure here a real time average throughput for MAC layer. That means average throughput measured during simulation time at MAC layer for both data and control packets. RTS/CTS case records higher throughput value than CSMA/CA because control traffic has smaller size and lower delay. According to Formula 4.1, measured throughput will be the same as data throughput for CSMA/CA case. But for RTS/CTS, actually delivered data throughput will be less than measured throughput.

4.2.5 Varying physical layer

In this part we will investigate the effect of physical layer on the performance. Figures(4.9,4.10) illustrate the throughput and delay performance with four different physical layers under CSMA/CA MAC layer. Same data rate and scenario of previous part is also used in this part. Table 4.3 illustrates the differences between different kinds of physical layer parameters [96].

We can see that how the performance of the network (throughput and delay) varies with different kinds of IEEE 802.11 standard physical layers . Theoretically speaking, 802.11a should have higher throughput than 802.11g [95], but wee see different behavior from Figure 4.9. Because 802.11a standard operates at higher frequency than 802.11g (see Table 4.2), it has lower range. This has negative effect on throughput performance in mobile scenario . Also 802.11a with slot window size of $9\mu.seconds$ could achieve less delay than 802.11g which has $20\mu.seconds$ slot window size [95]. We see from this part that physical layer also has an impact on the performance of the network. Also considering our suggested routing protocol (OLSR), it has best performance with 802.11 FHSS type of physical layer and worst performance with 802.11a one. Figure 4.11 shows how PDR performance of OLSR is also changing with different PHY layer.



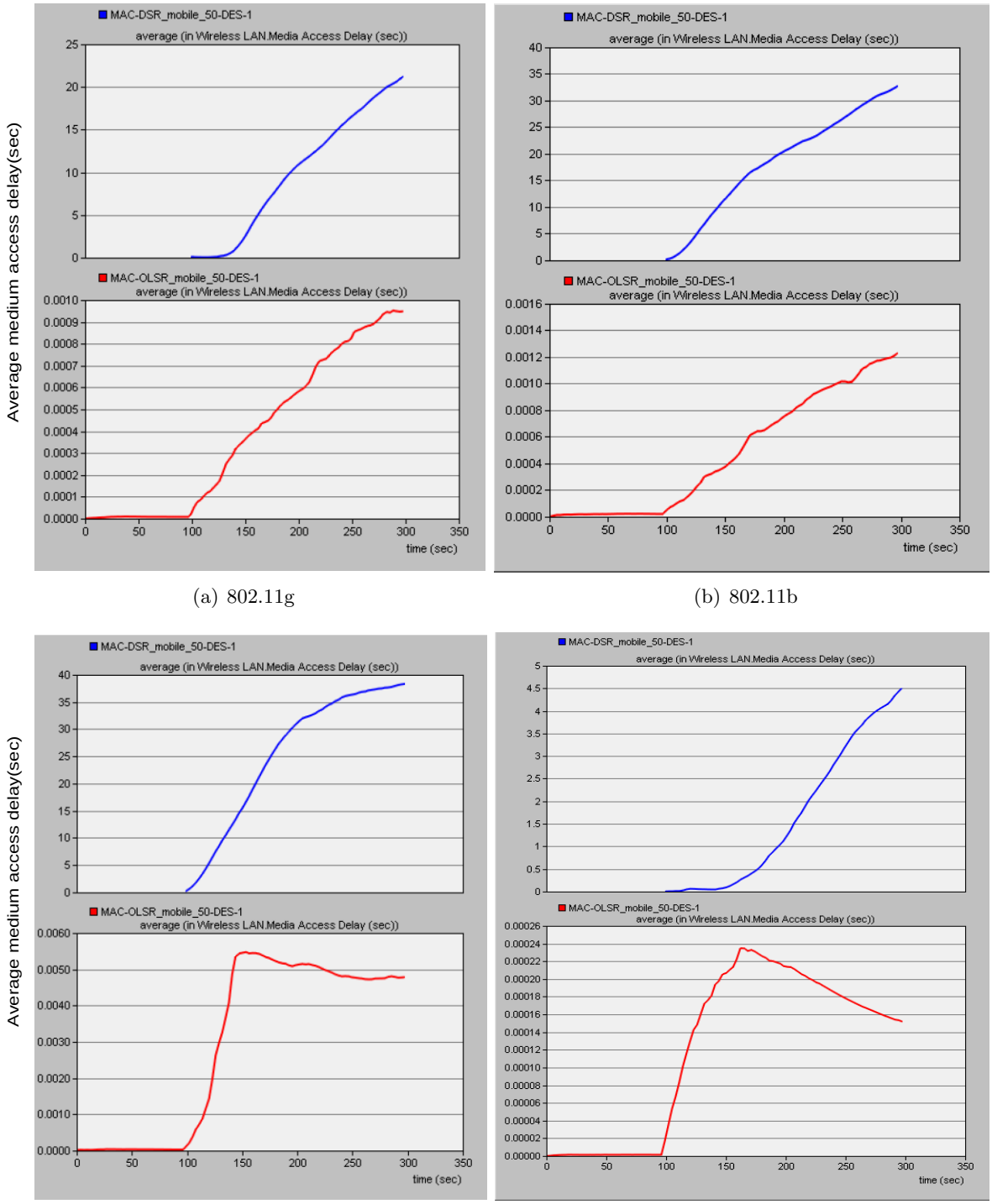
(a) 802.11g

(b) 802.11b

(c) 802.11/FHSS

(d) 802.11a

Figure 4.9: Throughput performance with different types of PHY layers



(a) 802.11g

(b) 802.11b

(c) 802.11/FHSS

(d) 802.11a

Figure 4.10: Delay performance with different types of PHY layers

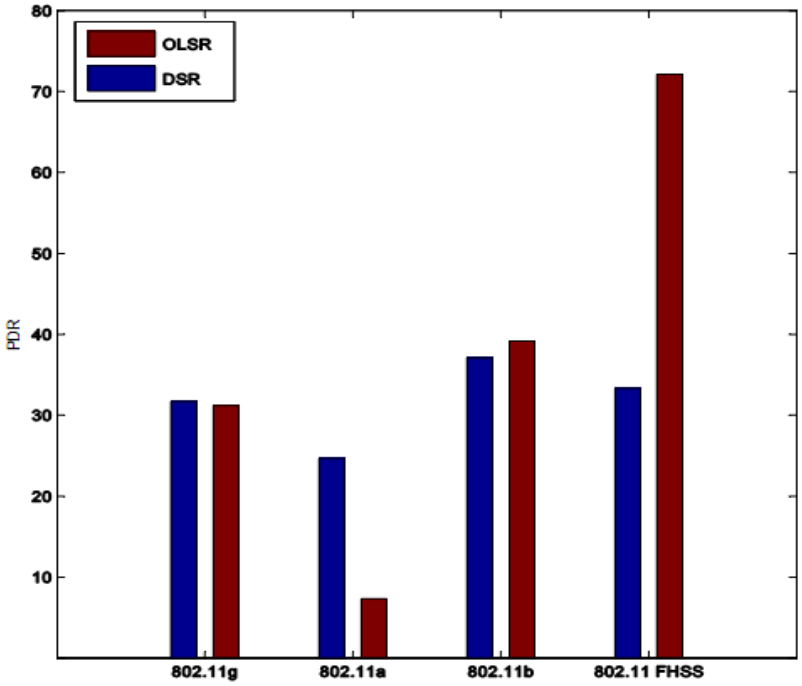


Figure 4.11: PDR performance for both protocols with different PHY layers

4.3 Summary

As we discussed in previous chapters, routing protocols (with large implementation variants) work on the top of MAC layer to assign routes for traffic and maintain these routes either temporary during transmission time (reactive) or permanently (proactive). Further we saw how these protocols could support dependable services in PN by adding dependability means to protocol implementation. But looking at definition of dependability attributes (Chapter 1), there exist a term called correct service. This term could be violated by MAC layer. As from dependability point of view, MAC layer has a great effect on the maximum throughput of an application (see Formula 1) and thus the ability to deliver the "correct service" or satisfy QoS requirements of a specific application. To enable dependability for a routing protocol, we must consider the interaction of this protocol with MAC layer. Because at the end, the network has to offer the required throughput needed for specific application (QoS requirement). We have presented the performance of two different kinds of routing protocols under different MAC and physical layers. The effect of route discovery time in reactive protocol (DSR) case has a great effect on the delay and throughput performance. While data packets with DSR protocol will have to wait more time because of route discovery, in OLSR case there is always route available for the data. This helped OLSR to show better throughput and delay performance under all different conditions. One important notice in this chapter is that, PDR performance depends solely on routing protocol mechanism. Although DSR has shown less throughput than OLSR but it could achieve higher PDR due to aggressive cashing mechanism. Dependable routing protocol should

deliver all data packets to destination with minimum delay to achieve required quality (data rate). Also we noticed that higher throughput for OLSR routing protocol allows to have smooth behavior when network condition changes(number of source nodes). Another interesting notice is that, for reactive protocols (DSR) number of multiple paths has stronger effect on throughput and delay performance than route expiry time. With higher number of routes there will be higher route discovery time which will cause longer delay and thus lower throughput. But from routing domain and as we saw in Chapters(2,3), increasing number of multiple paths allows for higher delivery ratio at destination. Adding multi-path technique to OLSR would improve the reliability. In later part, by changing MAC layer types we noticed larger change in delay performance with OLSR than DSR due to proactive nature. For both protocols we noticed lower PDR performance with RTS/CTS type of MAC layer due to adding more control overhead. We have also introduced the performance under different physical layers. Also we have shown the performance with different physical layers. IEEE 802.11a with high frequency band and shorter range has shown poor throughput and PDR performance with OLSR protocol than other physical layers. While 802.11 with FHSS enabled for better throughput and PDR performance. OFDM delivers less throughput than FHSS for same application data rate due to serial to parallel converter before IFFT stage. In general we would like to say, proactive protocols perform better than reactive protocols over all variants of 802.11 MAC layer with respect to throughput and delay. And because delay has major effect on throughput, it must be considered as route metric in routing protocols. Because this could ensure good performance with different MAC layers (heterogeneous network scenario). Beside delay metric there should be also some accurate link metrics like (ETX, ETT, see Chapter 2) to ensure good performance with different physical layers. The controller on the top will be a QoS routing protocol which would assign for each individual traffic a suitable route according to QoS requirements.

5 Summary

In this thesis we have answered the question *"How to enable dependable intra-routing for application data packets inside a personal network cluster"*. Personal network (PN) is dependable when reliance can justifiably be placed on the service it delivers. By definition, dependable system reduces the chance that a fault occurs to a minimal acceptable value. When a fault happens, the system should still be able to offer the expected correct service. We defined the fault in our case by inability for a network to carry out routing service due to either link breakage or link quality degradation. We have shown that, there are some requirements from a routing protocol to enable dependability in PN such as:

- *PN is always able to route application data packets at anytime (service is available all the time).*
- *PN is always able to route application data packets with required QoS agreements (ability to offer correct service all the time).*
- *PN is self recoverable from errors when they occur.*

To achieve these requirements, the following tools must be implemented in routing protocol:

- **Fault prevention**: proactive multipath protocol would always make some dependable route available and ready to route application data packets.
- **Fault tolerance**: multipath could also enable fault tolerance by reducing the effect of single path failure and exploit channel diversity (mitigate multipath fading effects)
- **Fault forecasting**: accurate link quality assessment or monitoring QoS levels at destination in real time could help to switch the traffic through alternative route/link prior to fault occurance.
- **Fault removal**: when an error happens, the protocol should be able to route the traffic through alternative route without affecting the overall performance. Local repair mechanism (like in AODV protocol) with/or cooperative caching technique would be able to fulfill this task.

After investigating the performance of some MANET routing protocols which could be suitable for PN (we excluded position based protocols), both reactive and proactive protocols have shown acceptable performance for supposed PN scenario (mobility speed of human being inside a room, connected network with maximum number of 100 nodes in area of 100 x 100 square meters.) We also introduced different multi-path

techniques which may enhance the performance of routing protocols. To refine our investigation we simulated a PN scenario to compare the performance of both reactive (DSR,AODV) and proactive (OLSR) protocols using OPNET 14.5 simulation environment in some mobile scenarios. For constant bit rate (CBR) traffic, OLSR has shown less delay and overhead performance than reactive protocols (DSR,AODV). But it has shown drawback with lower PDR performance. Aggressive caching technique in DSR allowed for higher PDR, where more than one path is available in cache memory of the source node. In case of client/server and client/client type of applications where communication happens in request/response fashion, OLSR routing protocol performed better than reactive protocols (AODV, DSR) due to proactive nature (routes in both directions are available all the time). TCP protocol may guarantee 100 % PDR for all protocols. While UDP has shown very bad performance with such type of traffic. Since power consumption is very crucial issue for ad-hoc networks, we investigate the performance of three protocols with respect to power consumption. We found that DSR has the least power consumption while OLSR has the largest amount of power consumption. Lower PDR and higher power consumption both are drawbacks for OLSR routing protocol which have to be recovered with some appropriate mechanisms. We conclude that, among available ad-hoc routing protocols OLSR is able to support dependability for PN with less modifications in default implementation than other protocols. Dependable routing protocol must satisfy QoS requirements for applications. Network resources must be shared among different applications. Multipath mechanisms with better route metrics than hop count (ETT, ETX, etc.) are needed. Multi-path technique is able to recover the weakness of OLSR protocol (low PDR performance) as was the case with DSR protocol (aggressive caching). Also the proactive nature of OLSR will allow the network to be always ready to offer routing service for application data in shorter time. In Chapter four we have shown that, the overall network performance is a result of an interaction between MAC and network layers. OLSR has always lower delay and higher throughput than DSR under all different conditions (source number, data rate, MAC /PHY type). Altough DSR has shown better PDR performance, but the delay caused by route discovery process (reactive nature of DSR) has cause longer MAC/application delay and lower throughput. PDR is solely dependent on routing protocol. To enhance the network performance with respect to this metric one should focus on network domain solution. Higher throughput for OLSR allows for smooth PDR behavior when network conditions changed, while DSR had sudden drop in PDR value. The number of multiple routes has larger effect on delay than route expiry time. Higher number of multiple paths will enable higher PDR performance but with cost of longer delay (see Chapters 2,4). At MAC layer level, delay has severe effect on throughput. Low throughput and long delay will cause bad performance for real time applications like gaming and multimedia applications. Since delay has major effect on throughput, it must be considered as route metric in routing protocols. This can ensure good performance with different MAC layers (heterogeneous network scenario). Beside delay metric there should be also some accurate link metrics like (ETX, ETT, see Chapter 2) to ensure good performance at physical layer level. QoS mechanism will work on resource sharing among different application demands. Acording to QoS requirements of individual applications, resources will be assigned or released. This

will optimize the use of scarce network resources of a personal network.

Abbreviations

ACK	acknowledgment
AODV	ad-hoc on demand distance vector routing protocol
B.W	band width
CBR	constant bit rate
CBR	component based routing protocol
CSMA/CA	carrier sense multiple access with collision avoidance
DSR	distance vector routing protocol
DSSS	direct sequence spread spectrum
FHSS	frequency hopping spread spectrum
FSK	frequency shift keying
FTP	file transfer protocol
HTTP	hypertext transfer protocol
IP	Internet protocol
IETF	Internet engineering task force
IEEE	institute for electrical and electronic engineers
LHS	left hand side
MAC	medium access control
MANET	mobile ad-hoc network
MPR	multi point relay
NIC	network interface card
OFDM	orthogonal frequency division multiplexing
OSI	Open Systems Interconnection Reference Model
OLSR	optimized link state routing protocol
PN	personal network
PDR	packet delivery ratio
PHY	physical layer
PSK	phase shift keying
RHS	right hand side
RTS/CTS	request to send / clear to send
RREP	route replay
RREQ	route request
RERR	route error
Rx	receive
Tx	transmit
TCP	transmission control protocol
UDP	universal datagram protocol
VOIP	voice over Internet protocol
VS	versus
w.r.t	with respect to

Bibliography

[1] Martin Jacobson, "Personal networks, architecture for self organized personal wireless communications" *Ph.D. dissertation, Delft University of Technology 2008, Netherlands*

[2] Algirdas,Jean-claude Laprie,Brian Randell, "Fundamental concepts of computer system dependability" *IARO/IEEE-RAS Workshop on Robot dependability: Technological Challange of Dependable Robots in Human Environments, Seoul,Korea, May2001*

[3] I.G.Niemegeers and S.M.Heemstra,"Research issues in Ad-Hoc Distributed Personal Networking" *wireless personal communications;an international journal, volume 26,issue 2-3 (2003), ISSN 0929-6212*

[4] Divya Prasad, John Mc Dermid and Ian Wand,"Dependability Terminology: Similarities and Differences" *IEEE Aerospace and Electronic Systems Magazine 1996*

[5] Ramin Hekmat,"Fundamental Properties of Wireless Mobile ad-hoc Networks" *Ph.D. dissertation, Delft University of Technology 2005, Netherlands*

[6] S.Corson, J.Macker, "Routing Protocol Performance Issues and Evaluation Considerations" *RFC 2501, January 1999*

[7] Cmpista, Esposito, Moraes, Costa, Duarte, Passos,de Albuquerque, Saade, Rubinstein " Routing Metrics and Protocols for Wireless Mesh Networks" *Network,IEEE, volume: 22,issue:1,Pages:6-12,Feb 2008*

[8] Matthias Hollick, Ivan Martinovic, Tronje Krop, Ivica Rimac, "A survey on dependable routing in sensor networks,ad-hoc networks and cellular networks " *In proceeding of the 30th EUROMICRO conference(August 31-September03,2004),Pages:495-502,IEEE Computer Society,Washington,DC 2004*

[9] Y.Ganjali, A.Keshavarzian, "Load Balancing in Ad Hoc Networks: Single-path Routing vs. Multi-path Routing " *INFOCOM 2004,23rd Annual Conference of the IEEE Computer and Communications Societies,Vol.2(2004),pp.1120-1125, vol.2*

[10] Victor, K. Li, Zhenxin Lu, "Ad Hoc Network Routing" *international conference on networking sensing and control, IEEE 2004*

[11] Claudio Basile,Marc-Olivier Killijian ,David Powell, "A survey of Dependability Issues in Mobile Wireless Networks" *Laas CNRS Toulouse, France 2003*

[12] JosephP.Macker M.ScottCorson, "Mobile Ad-Hoc Networking and the IETF" *ACM mobile computing and communications review,Pages:11-13,vol 3,issue 1,January 1999*

[13] Omer Ozan Sonmez, "A Survey on Reliable Multicast Approaches for Mobile Ad Hoc Networks" *Software technologies,IEEE 2005*

[14] Myung Jong Lee, Jianling Zheng, Xuhui Hu, Hsin-hui Juan, Chunhui Zhu, Yong Liu, June Seung Yoon, and Tarek N. Saadawi, "A New Taxonomy of Routing Algorithms for Wireless Mobile Ad Hoc Networks:The Component Approach" *Communications magazine IEEE 2006,volume:44, issue:11*

[15] Yaling Yang, Jun Wang, Robin Kravets, "Designing Routing Metrics for Mesh Networks" *IEEE workshop on wireless mesh networks, WiMesh 2005*

[16] L.Zho, Z.Haas, "Securing ad-hoc networks" *special issues on network security,vol 13,issue 6 IEEE network magazine 1999*

[17] MaheshK. Marina Samir R. Das, "On-demand Multipath Distance Vector Routing in Ad Hoc Networks" *ACM SIGMOBILE mobile computing and communications review, volume 6,issue 3, july 2002*

[18] Miguel Elias M.Campista,Diego G.Passos, "Routing Metrics and Protocols for WMN" *networks IEEE 2008, volume 22, issue 1.*

[19] Per Johnsson, Tony Laresson, Nicklas Hedman, Bartosz Mielczarek, "Scenario-based Performance Analysis of Routing Protocols for Mobile ad-hoc networks" *International conference on mobile computing and networking, proceeding of the 5th annual ACM/IEEE international conference 1999,Seattle, Washington, United states,Pages:195-206*

[20] Samir R.Das, Charles E.Perkins, and Elizabith M.Royer, "Performance comparison of two On-demand Routing protocols for Ad-hoc Networks" *INFOCOM IEEE 2000,pp. 16-28, vol. 8, IEEE personal communications 2001*

[21] Muhammad Mahmudul Islam,Ronald pose,Carol Copp, "Routing Protocols for Ad-hoc Networks" *chapter IX , mobile multimedia communications:concept applications and challenges, IGI global 2008*

[22] Farhat Anwar, Md. Saiful Azad, Md. Arafatur Rahman, and Mohammad Moshee Uddin, "Performance Analysis of Ad hoc Routing Protocols in Mobile WiMAX Environment " *IAENG international journal of computer science July 2008,vol: 35, issue: 3,pages:353-360*

[23] M.Abolhasan, T.A.Wysocki ,"Performance investigation on three-class of MANET routing protocols " *conference on communications, Asia-pacific, IEEE 2005*

[24] Mehran Abolhasan, Tadeusz Wysocki, Eryk Dutkiewicz ,"A review of routing protocols for mobile ad hoc networks" *ad-hoc networks, science direct, volume 2, issue 1, January 2004*

[25] Huda Al Amri, Mehran Abolhasan , Tadeusz Wysocki,"Scalability of MANET Routing Protocols for Heterogeneous and Homogeneous Networks " *computer and electrical engineering,Copyright ElSEVIER 2009,ISSN 0045-7906, DOI: 10.1016/j.compeleceng.2008.11.008, http://www.sciencedirect.com/science/article/B6V25-4V936J9-1/2/4660fd37a8e25512aa4f0272adfcc512*

[26] T G Basavaraju and Subir Kumar Sarkar,"ECA1RP: An Efficient Congestion Adaptive Routing Protocol for Mobile Ad hoc Networks" *6th international conference on ITS telecommunications proceedings , IEEE 2006*

[27] Narendra Singh Yadav, R.P.Yadav, "The Effects of Speed on the Performance of Routing Protocols in Mobile Ad-hoc Networks" *International Journal of Electronics, Circuits and Systems Volume 1 Number 2,2008*

[28] Xiaoguang Niu, Zhihua Tao, Gongyi Wu, Changcheng Huang, Li Cui, "Hybrid Cluster Routing: An Efficient Routing Protocol for Mobile Ad Hoc Networks" *IEEE international conference on communications 2006.*

[29] Arun Kumar, Lokanatha.C.Reddy, Prakash Hiremath, "Performance Comparison of Wireless Mobile Ad-Hoc Network Routing Protocols" *IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.6, June 2008*

[30] Elizabeth M. Royer, Chai-Keong Toh, "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks" *IEEE personal communications 1999, vol 6,pages:46-55*

[31] S.R.Chaudhry,A.N.Al-Khwildi,Y.K.Casey,H.Aldelou and H.S.Al-Raweshidy, "Proactive and Reactive Routing Protocol Simulation Comparison" *WiMob information and communication technologies,ICTTA, IEEE 2006*

[32] Victor.Li,Zhenxin Lu, "Ad Hoc Routing" *international conference on networking,sensing and control, IEEE 2004*

[33] Nandiraju, Nagesh S. and Nandiraju, Deepti S. and Agrawal, Dharma P., "Multipath Routing in Wireless Mesh Networks" *Mobile Adhoc and Sensor Systems (MASS), 2006 IEEE International Conference on,DOI:10.1109/MOBHOC.2006.278644,http : //dx.doi.org/*10.1109/*MOBHOC*.2006.278644

[34] Adibi, S.Erfani,"A multipath routing survey for mobile ad-hoc networks",*Consumer Communications and Networking Conference, 2006. CCNC 2006. 3rd IEEE,Volume 2, Issue , 8-10 Jan. 2006 Page(s): 984 - 988*

[35] Adibi, S. and Erfani, S., "A multipath routing survey for mobile ad-hoc networks" *Consumer Communications and Networking Conference, 2006. CCNC 2006. 3rd IEEE,vol 2,pages:984-988*

[36] The ATM Forum, "Private network-network interface" *(PNNI 1.0)*

[37] Stephen Mueller, Rosep. Tsang, Dipak Ghosal , "Multipath routing in mobile ad hoc networks: Issues and challenges " *In Performance Tools and Applications to Networked Systems, volume 2965 of LNCS 2004*

[38] Shivanajay Marwaha Chen , Chen Khong , Tham Dipti Srinivasan , "Mobile Agents based Routing Protocol for Mobile Ad Hoc Networks" *In in Proceedings of IEEE GLOBOCOM 2002*

[39] Tamilarasi,M.Shyam Sunder,V.R.Haputhanthri,U.M.Somathilaka, C.Babu,N.R.Chandramathi,S.Palanivelu, T.G., "Scalability Improved DSR Protocol for MANETs" *International Conference on Computational Intelligence and Multimedia Applications, 2007.Publication Date: 13-15 Dec. 2007,Volume: 4,On page(s): 283-287,Location: Sivakasi, Tamil Nadu,ISBN: 0-7695-3050-8*

[40] Mehran Abulhassan, Tadeusz Wysocki ,Justin Lipman, "A New Strategy to Improve Proactive Route Updates in mobile ad hoc networks" *EURASIP Journal on Wireless Communications and Networking,Volume 2005 , Issue 5*(*October*2005)

[41] Krishnan,R.Silvester, J.A., "Choice of Allocation Granularity in Multipath Source Routing Schemes" *INFOCOM '93. Proceedings.Twelfth Annual Joint Conference of the IEEE Computer and Communications Societies. Networking: Foundation for the Future. IEEE Publication Date: 1993,On page(s): 322-329 vol.1,Meeting Date: 03/28/1993 - 04/01/1993,Location: San Francisco, CA, USA,ISBN: 0-8186-3580-0*

[42] Shengming Jiang, Dajiang He, and Jianqiang Rao, "A Prediction-based link availability estimation for mobile ad hoc networks" *IEEE INFOCOM 2001*

[43] Roy Leung,Roy Leung Jilei, Edmond Poon,Ah-lot Charles Chan,Baochun Li, "MP-DSR: A QoS-aware Multi-path Dynamic Source Routing Protocol for wireless ad hoc networks" *IEEE LCN01*

[44] Ye,Z.Krishnamurthy,S.V.Tripathi,S.K., "A Framework for Reliable Routing in Mobile Ad hoc networks" *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies. IEEE Publication Date: 30 March-3 April 2003,Volume: 1, On page(s): 270- 280 vol.1,ISSN: 0743-166X ,ISBN: 0-7803-7752-4*

[45] Kui Wu, Janelle Harms,"Performance Study of a Multipath Routing Method for Wireless Mobile Ad Hoc networks" *lecture notes on computer science,Springer Berlin,*$ISSN : 0302-9743(Print)1611-3349(Online)$

[46] Pearlman,M.R.Haas,Z.J.Sholander,P.Tabrizi, S.S.,"On the Impact of Alternate Path Routing for Load Balancing in mobile ad hoc networks" *Mobile and Ad Hoc Networking and Computing, 2000. MobiHOC. 2000 First Annual Workshop on Publication Date: 2000,On page(s): 3-10,Location: Boston, MA, USA,ISBN: 0-7803-6534-8*

[47] Hui Wang, Ke Ma and Nenghai Yu., "Performance Analysis of Multi-path Routing in wireless ad hoc networks" *IEEE 2005,vol 2, pages:723-726*

[48] Ram Krishnan and John A.Selvester, "Choice of Allocation Granularity in Multipath source routing scheme" *IEEE INFOCOM 1993, SanFrancisco, CA, March 1993, pp:322-329*

[49] Asis Nasipuri and Samir R. Das, "On-Demand Multipath Routing for Mobile Ad Hoc Networks" *In Procceedings of the 8th annual IEEE International Conference on Computer Communications and Networks(ICCCN),Boston, MA, October 1999, pp.64-70*

[50] Alvin Valera, Winston K.G. Seah, and SV Rao, "Cooperative Packet Caching and Shortest Multipath Routing in Mobile Ad hoc Networks" *In procceedings of IEEE INFOCOM March-April 2003, pp 260-269*

[51] Ayanoglu,E.Chih-Lin,I.Gitlin,R.D.,Mazo,J.E.,"Diversity Coding for Transparent Self-Healing and Fault-Tolerant Communication Networks" *Communications, IEEE Transactions on,Volume 41, Issue 11, Nov 1993 Page(s):1677 - 1686,Digital Object Identifier 10.1109/26.241748*

[52] N.F.Maxemchuk, "Dispersity routing: Past and present" *Military communications conference MILCOM IEEE 2007*

[53] Siuli Roy,Dola Saha,Somprakash Bandyopadhyay,Somprakash B,Shinsuke Tanaka,Tetsuro Uera, "Improving End-to-End Delay through Load Balancing with Multipath Routing in Ad Hoc Wireless Networks using Directional Antenna" *in Proc. IWDC 2003: 5th International Workshop, LNCS v2918*

[54] Rui Ma and Jacek Ilow, "Reliable Multipath Routing with Fixed Delays in MANET Using Regenerating Nodes" *Procceedings of the 28th annual IEEE International Conference on Local Computer Networks 2003, pp 719*

[55] Chunsoo Ahn, Jitae Shin, and Eui-Nam Huh, "Enhanced Multipath Routing Protocol Using Congestion Metric in Wireless Ad Hoc Networks" *lecture notes on computer science,Springer Berlin,*$0302-9743(Print)1611-3349(Online)$*,2006*

[56] Wen-Hwa Liao, Yu-Chee Tseng, Shu-Ling Wang, and Jang-Ping Sheu, "A Multi-path QoS Routing Protocol in a Wireless Mobile ad Hoc Network" *Proceedings of the First International Conference on Networking-Part 2,Lecture Notes In Computer Science; Vol. 2094,2001*

[57] Yuh-shyan Chen,Yu-Chee Tseng,Jang-ping Sheu,Po-hsuen Kuo, "On-Demand, Link-State, Multi-Path QoS Routing in a Wireless Mobile ad-hoc network" *Computer communications 2002,volume 27*

[58] Chunhung Richard LIN, "On-Demand QoS Routing in Multihop Mobile Networks" *IEEE 2001*

[59] Shirshu Varma,Tiwary,U.S.Anshul,Jain Sharma, "Statistical Energy Efficient Multipath Routing protocol" *international conference on Information Networking, 2008. ICOIN 2008,Volume , Issue , 23-25 Jan. 2008 Page(s):1 - 5*

[60] ZHAN Song-tao, XU Guo-xin, "Rate allocation strategies for energy-efficient multipath routing in Ad-hoc networks towards B3G " *Copyright 2007 The Journal of China Universities of Posts and Telecommunications Published by Elsevier B.V. doi* : $10.1016/j.physletb.2003.10.071$

[61] Peter P.pham and Sylvie Perreau, "Performance analysis of reactive shortest path and multipath routing mechanism with load balance" *IEEE 2003*

[62] Aristotelis and Zygmunt, "Analysis of multipath routing,part1: the effect of packet delivery ratio" *IEEE January 2004*

[63] Yashar Ganjali, Abtin Keshavarzian, "Load balancing in ad hoc networks: singlepath routing vs. multipath routing" *Proc. IEEE INFOCOM '04, Mar. 2004*

[64] Castaneda and Das, "Query localization techniques for on-demand routing protocols in ad hoc networks" *Selected Papers from Mobicom'99,Pages: 137 - 151,2002,ISSN:1022-0038*

[65] J.H.Weber, "Error correcting codes" *Lecture notes TU delft university 2007*

[66] Mao Kun, Yu Jingdong, Ren Zhi, "The research and simulation of multi-path OLSR for mobile ad-hoc network" *International symposium on communications and information technology, ISCIT IEEE 2005*

[67] Jaizi Yi, Eddy Cizron, Salima Hamma, Benoit Parrein, "Simulation and performance analysis of MP-OLSR for mobile ad-hoc networks" *IEEE wireless communications and networking conference, March 31-April 3,Las Vegas IEEE WCNC 2008*

[68] Xun Zhou, Yu Lu , Ge Ma, "Routing improvement using multiple disjoint paths for ad hoc networks" *IEEE WOCN 2006, International conference on wirless and optical communications networks, pages 5*

[69] Xun Zhou, Yu Lu, Bin Xi, "A novel routing protocol for ad-hoc sensor networks using multiple disjoint paths" *2nd international conference on Broadband Networks, Broadnets 2005,pages:944-948,vol 2, IEEE October 2005*

[70] Mao Kun, Yu Jingdong, Ren Zhi, "The research and simulation of multipath OLSR for mobile ad-hoc network" *International symposium on Communications and Information Technology 2005, ISCIT IEEE October 2005, vol 1, pages:540-543*

[71] Dang Quan Nguyen, Pascale Minet, "QoS support and OLSR routing in a mobile ad-hoc network" *in 1st proceeding of the international conference on networking ,systems ,mobile communications and learning technologies ICNICONSMCL 2006, IEEE 2006*

[72] Christian Bettstetter, "The Connectivity of Ad-hoc Networks" *The computer Journal vol 47, no. 4, the British computer society 2003*

[73] Matheus.k,Zurbes.S, Taori.R, "Fundamental properties of ad-hoc networks like Bluetooth: a radio network perspective" *58th Vehicular Technology Conference, VCT IEEE fall-2003, vol 5, pages:3050-3054*

[74] K.Murugan , S.Shanmugavel,"Implementation and Performance Study of Route Caching Mechanisms in DSR and HER Routing Algorithms for MANET" *Parallel and Distributed Processing and Applications ISPA 2005, LNCS 3758, pp. 1135 1145, Springer Berlin / Heidelberg 2005*

[75] E. Hyytia, H. Koskinen, P. Lassila, A. Penttinen and J. Virtamo,"Random Way Point Model in Wireless Networks" *Networks and Algorithms: complexity in Physics and Computer Science Helsinki, June 16-19, 2005*

[76] M. Stemm, R.H. Katz,"Measuring and reducing energy consumption of network interfaces in hand-held devices" *IEICE Transactions on Communications E80-B (8) (1997) 11251131*

[77] Wendi Rabiner Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan,"Energy-Efficient Communication Protocol for Wireless Microsensor Networks" *Proceedings of the 33rd Hawaii International Conference on System Sciences 2000*

[78] Q. Gao , K.J. Blow , D.J. Holding , I.W. Marshall , X.H. Peng,"Radio range adjustment for energy efficient wireless sensor networks"*doi* : 10.1016/*j.adhoc.*2004.04.007, 2004 *Elsevier B.V. All rights reserved*

[79] Gomez, J.Campbell, A.T.Naghshineh, M.Bisdikian,"Conserving Transmission Power in Wireless Ad Hoc Networks", *9th international conference on network protocols,IEEE 2001*

[80] Alvin Valera, Winston K. G. Seah, Sv Rao,"Cooperative packet caching and shortest multipath routing in mobile ad hoc networks" *in Proceedings of IEEE INFOCOM, March-April 2003*

[81] Marco Fotino, Antonio Gozzi, Juan-Carlos Cano, Carlos Calafate, Floriano De Rango, Pietro Manzoni, and Salvatore Marano,"Evaluating Energy Consumption of Proactive and Reactive Routing Protocols in a MANET" *IFIP International Federation for Information Processing, wireless sensor and actor networks, ISBN 978-0-387-74898-6, Springer link 2007*

[82] Samir.Das,"Ad hoc On-Demand Distance Vector (AODV) Routing" *RFC3561,Network Working Group 2003*

[83] *www.OPNET.com*

[84] Mingzhe LI, Mark Claypool, Rbert Kinicki, and James Nichols,"Characteristics of Streaming Media Stored on the Web" *ACM Transactions on Internet Technology, Vol. 5, No. 4, November 2005, Pages 601626*

[85] S.D. Hoglund, D.H., Baker,"Designing an Enterprise Mobility Solution in the Health care Environment" *Engineering in Medicine and Biology Magazine, IEEE, March-April 2008 Volume: 27, Issue: 2 page(s): 86-95*

[86] David B. Johnson, David A. Maltz, Yih-Chun Hu,"The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)" *RFC 4728, February 2007*

[87] T. Clausen, P. Jacquet , Project Hipercom,"Optimized Link State Routing Protocol (OLSR)" *RFC 3626,Network working group, October 2003*

[88] Laura Marie Feeney, Martin Nilsson," Investigating the Energy Consumption of a Wireless Network Interface in an Ad Hoc Networking Environment" *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE 2001*

[89] Juan-Carlos Can and Pietro Manzoni,"A Performance Comparison of Energy Consumption for Mobile Ad Hoc Network Routing Protocols" *ISBN 0-7695-0728-WOO, 2000 IEEE.*

[90] Amer Filipovic and Amitava Datta,"Building Blocks of Energy and Cost Efficient Wireless Sensor Networks" *EWSN 2004, LNCS 2920, pp. 218233, 2004 , Springer-Verlag Berlin Heidelberg 2004*

[91] "http://www.technologyreview.com"

[92] Jangeun Jun, Pushkin Peddabachagari, Mihail Sichitiu,"Theoretical Maximum Throughput of IEEE 802.11 and its Applications",*In Proceedings of the IEEE International Symposium on Network Computing and Applications 2003*

[93] Omesh Tickoo and Biplab Sikdar,"Queuing Analysis and Delay Mitigation in IEEE 802.11 Random Access MAC based Wireless Networks" *IEEE INFOCOM 2004*

[94] Y. Tay and K. Chua,"A capacity analysis for the IEEE 802.11 MAC protocol" *Wireless Networks, vol. 7, no. 2, pp. 159-171, March, 2001*

[95] Puttipong Mahasukhon, Micheal Hempel, Song Ci, Hamid Sharif,"Comparison of Throughput Performance for the IEEE 802.11a and 802.11g Networks" *21st international conference on advanced networking and applications (AINA'07), ISBN: 0-7695-2846-5/07, IEEE 2007*

[96] Dimitris Vassiss, G.Kormentzas, A.Rouskas, I.Maglogiannis,"The IEEE 802.11g standard for high data rate WLANs" *ISBN: 0890-8044/05, IEEE network 2005*

[97] Fouad Tobagi, Amit Vyas, Sangwook Ha, Olufunmilola Awoniyi,"Interactions between the physical layer and upper layers in wireless networks" *doi:10.1016/j.adhoc.2007.02.015, Elsevier B.V 2007*

[98] $http://www.ietf.org/rfc/rfc2501.txt$