A Cross-Layer Modification to the DSR Routing Protocol in Wireless Mesh Networks

Mustafa Ramadhan
Technological University Dublin

Follow this and additional works at: https://arrow.tudublin.ie/engdoc

Part of the Digital Communications and Networking Commons

Recommended Citation
A Cross-Layer Modification to the DSR Routing Protocol in Wireless Mesh Networks

by

Mustafa Khalid Ramadhan
MSc (Eng.)

A thesis submitted to the Dublin Institute of Technology for the degree of

Doctor of Philosophy

Communications Network Research Institute (CNRI), School of Electronic and Communications Engineering, Dublin Institute of Technology (DIT), Dublin, Ireland.

September 2010
Abstract

A cross-layer modification to the DSR routing protocol that finds high throughput paths in WMNs has been introduced in this work. The Access Efficiency Factor (AEF) has been introduced in this modification as a local congestion avoidance metric for the DSR routing mechanism as an alternative to the hop count ($H_c$) metric. In this modification, the selected path is identified by finding a path with the highest minimum $AEF (\text{max\_min\_AEF})$ value. The basis of this study is to compare the performance of the $H_c$ and $\text{max\_min\_AEF}$ as routing metrics for the DSR protocol in WMNs using the OPNET modeler. Performance comparisons between $\text{max\_min\_AEF}$, Metric Path ($MP$), and the well known $ETT$ metrics are also carried out in this work. The results of this modification suggest that employing the $\text{max\_min\_AEF}$ as a routing metric outperforms the $H_c$, $ETT$, and $MP$ within the DSR protocol in WMNs in terms of throughput. This is because the $\text{max\_min\_AEF}$ is based upon avoiding directing traffic through congested nodes where significant packet loss is likely to occur. This throughput improvement is associated with an increment in the delay time due to the long paths taken to avoid congested regions. To overcome this drawback, a further modification to the routing discovery mechanism has been made by imposing a hop count limit ($H_{CL}$) on the discovered paths. Tuning the $H_{CL}$ allows the network manager to trade-off throughput against delay. The choice of congestion avoidance metric exhibits another shortcoming owing to its dependency on the packet size. It penalises the smaller packets over large ones in terms of path lengths. This has been corrected for by introducing a $\text{ModAEF}$ metric that explicitly considers the size of the packet. The $\text{ModAEF}$ metric includes a tuning factor that allows the operator determine the level of the weighting that should be applied to the packet size to correct for this dependence.
Declaration

I certify that this thesis which I now submit for examination for the award of ________________, is entirely my own work and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

This thesis was prepared according to the regulations for postgraduate study by research of the Dublin Institute of Technology and has not been submitted in whole or in part for an award in any other Institute or University.

The work reported on in this thesis conforms to the principles and requirements of the Institute's guidelines for ethics in research.

The Institute has permission to keep, to lend or to copy this thesis in whole or in part, on condition that any such use of the material of the thesis be fully acknowledged.

Signature_________________________                Date___________________
Acknowledgements

I would like to express my gratitude to my supervisor Dr. Mark Davis for giving me the opportunity to study at CNRI. All the staff at CNRI, past and present, have influenced this work in some positive way.

I would like to give special thanks to Professor Gerald Farrell, Dr. Fred Mtenzi, and Barry Duignan for their support and encouragement. Special thanks also go to Professor Hugh Byrne at the Focas Institute. I also like to thank Mr. Bertrand Faust and Desmond Kernan for allowing the use of the OPNET laboratory and for their cooperation.

My friends, in particular Bernadette O’Donoghue, Sister Thomasina Cosgrave, and Carole Craig, have given me overwhelming encouragement and support throughout my education.

Most of all I would like to thank my parents. Everything I know and do is because of them. I also like to give special thanks for my siblings for their ongoing support.

Finally, I would like to thank my wife, Alaa Jaber, for her patience, love and encouragement she has my deepest gratitude.
Abstract.............................................................................................................................. i

Declaration........................................................................................................................ ii

Acknowledgement ........................................................................................................... iii

Chapter 1            Introduction .....................................................................................1

1.1 Thesis Organisation ...................................................................................................8

Chapter 2            Technical Background ........................................................................10

2.1 Introduction to IEEE 802.11 ..................................................................................10

2.1.1 IEEE 802.11b Standard.......................................................................................11

2.1.2 IEEE 802.11a Standard.......................................................................................12

2.1.3 IEEE 802.11g Standard......................................................................................12

2.1.4 IEEE 802.11e Standard.......................................................................................13

2.1.5 IEEE 802.11h Standard.......................................................................................13

2.1.6 IEEE 802.11n Standard.......................................................................................13

2.2 IEEE 802.11 MAC Mechanism ..............................................................................14

2.2.1 Distributed Coordination Function (DCF)..........................................................14

2.3 Access Efficiency Factor (AEF) and MAC Bandwidth Components......................18

2.3.1 AEF and a Station Capacity under Ideal Network Conditions .........................21

2.3.2 AEF and a Station Capacity in the Presence of other Stations............................23

2.4 Summary ..................................................................................................................27
CHAPTER 3  ROUTING OVERVIEW OF WMNS.................................28

3.1 Wireless Mesh Network..............................................................28

3.2 Routing in Wireless Mesh Network........................................35

3.2.1 Routing Metrics.................................................................39

3.2.1.1 Hop Count ........................................................................39

3.2.1.2 Per-hop round trip time (RTT) ......................................40

3.2.1.3 Per-hop Packet Pair Delay (PktPair) ............................41

3.2.1.4 Expected Transmission Count (ETX) ..............................42

3.2.1.5 Expected Transmission Time (ETT) ...............................44

3.2.1.6 Weighted Cumulative ETT (WCETT) ..............................45

3.2.1.7 Metric of Interference and Channel-switching (MIC) ....46

3.2.1.8 Expected Throughput (ETP) ...........................................48

3.2.1.9 Bottleneck Link Capacity (BLC) .................................49

3.2.1.10 Metric Path (MP) .......................................................50

3.2.2 Comparison of Metrics.....................................................52

3.2.3 A New Congestion Aware Routing Strategy ......................55

3.2.4 Overview of routing protocols ..........................................55

3.2.4.1. Destination Sequenced Distance Vector (DSDV) Routing 56
3.2.4.2 Open Shortest Path First (OSPF) ................................................................. 58
3.2.4.3 The Wireless Routing Protocol (WRP) .......................................................... 60
3.2.4.4 Zone Routing Protocol (ZRP) ......................................................................... 62
3.2.4.5 Temporally Ordered Routing Algorithm (TORA) ........................................... 64
3.2.4.6 Optimized Link State Routing Protocol (OLSR)............................................. 66
3.2.4.7 Ad Hoc On-Demand Distance Vector Routing (AODV) .............................. 69
3.2.4.8 Dynamic Source Routing (DSR) .................................................................... 71
3.2.4.9 Modification to the DSR .............................................................................. 80
3.3 Summary ............................................................................................................. 81

CHAPTER 4 SIMULATION MODEL ................................................................. 82

4.1 Justification for Adopting Simulation Approach .................................................... 82
4.2 Network Modeling ............................................................................................ 83
4.2.1 Network Modeling Using OPNET ................................................................. 85
4.2.2 DSR Model for WLAN Node .......................................................................... 87
4.2.3 Implementing the modification ....................................................................... 91
4.2.4 Simulation settings ....................................................................................... 98
4.3 Free Space Propagation .................................................................................... 98
4.4 Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) ....... 101
4.5 Density factor (DF) ........................................................................................ 102
4.6 Modified DSR Routing Discovery Mechanism .................................................103

4.6.1 First Modification to the DSR Routing Discovery Mechanism .................104

4.6.2 Second Modification to the DSR Routing Discovery Mechanism ..............107

4.6.3 Third Modification to the DSR Routing Discovery Mechanism .................107

4.7 Methodology .....................................................................................................108

4.7.1 Scenarios of Group A .................................................................111

4.7.2 Scenarios of Group B ................................................................111

4.7.3 Scenarios of Group C ..................................................................111

4.7.3.1 Scenarios of Group C-1 ......................................................112

4.7.3.2 Scenarios of Group C-2 ......................................................112

4.7.3.3 Scenarios of Group C-3 ......................................................112

4.7.4 Scenarios of Group D ..................................................................112

4.7.5 Scenarios of Group E ..................................................................113

4.7.6 Scenarios of Group F ..................................................................113

4.7.7 Scenarios of Group G ..................................................................114

4.7.8 Scenarios of Group H .................................................................115

4.7.9 Scenarios of Group I ..................................................................115

4.7.10 Scenarios of Group J ...............................................................116

4.7.11 Scenarios of Group K ...............................................................116
4.7.11.1 Scenarios of Group $K-1$ .................................................................117
4.7.11.2 Scenarios of Group $K-2$ .................................................................117
4.7.12 Scenarios of Group $L$ .........................................................................118
4.8 Modeling Assumption ................................................................................118
4.9 Summary ......................................................................................................122

CHAPTER 5  RESULTS AND DISCUSSIONS ........................................123
5.1 Operation of the Modified $DSR$ Path Selection Rule ..............................123
5.2 Performance Examination of the Modified $DSR$ Protocol .......................135
5.2.1 Simulation Results of Implementing the Modified $DSR$ protocol ..........137
5.2.2 Effect of Load Variations ......................................................................144
5.2.3 Effect of Traffic Type ..........................................................................151
5.2.4 Effect of the Number of Available Gateways ......................................153
5.2.5 Uplink and Downlink Traffic Stream ....................................................156
5.2.6 Transport Protocol ..............................................................................161
5.2.7 Incorporation of the Hop Count Parameter .........................................163
5.3 Modified Version of $AEF$ ......................................................................167
5.3.1 Simulation Results Obtained for $ModAEF$ .........................................170
5.3.2 Examination of the Effect Packet Size Variation .................................175
5.4 Dynamic behaviour of the new metric .....................................................178
5.5 Performance Evaluation of the AEF metric against the ETT and MP ...............180

5.6 Performance Comparison of the Routing Metrics Examined in the Thesis........184

5.7 Summary .............................................................................................................186

Chapter 6  Summary and Conclusion .................................................................188

6.1 Conclusions ........................................................................................................194

6.2 Suggestions for Future Work ...........................................................................195

Publications ..........................................................................................................199

References .............................................................................................................200

Appendix ..............................................................................................................239
List of Figures

Fig. 2.1: Example of DCF operation .................................................................17

Fig. 2.2: The various time intervals involved in accessing
the medium under CSMA/CA .................................................................19

Fig. 2.3: The MAC bandwidth operating plane description .........................23

Fig. 3.1: Infrastructure/Backbone WMNs .................................................30

Fig. 3.2: Client WMNs .............................................................................31

Fig. 3.3: Hybrid WMNs ..........................................................................32

Fig. 3.4: Hierarchical architecture of nodes with cluster heads ....................37

Fig. 3.5: Estimated transmission count (ETX)
to node E from each node .......................................................................43

Fig. 3.6: An example of the DSR route discovery mechanism .....................73

Fig. 3.7: An example of the DSR gratuitous mechanism ............................76

Fig. 3.8: An example of the DSR route maintenance-caching
negative information ..............................................................................76

Fig. 4.1: Hierarchical architecture of the OPNET
simulation modeler ..................................................................................86

Fig. 4.2: Node model of a WLAN station ....................................................89

Fig. 4.3: Data traffic packet flow ...............................................................90

Fig. 4.4: Flow chart of the modified DSR route discovery mechanism ..........94

Fig. 4.5: Example of Route Discovery mechanism .....................................105

Fig. 4.6: Network topology examples of different
number of gateways nodes (in Red) ......................................................114
Fig. 5.1: Basic packet forwarding operation at a node ..................................................124

Fig. 5.2: Operation of the modified DSR path selection rule
based upon AEF metric ........................................................................................................125

Fig. 5.3: Congestion avoidance operation of the
modified DSR path selection rule ..........................................................................................127

Fig. 5.4: Interaction model of various network load
parameters based upon AEF metric .......................................................................................129

Fig. 5.5: Load distribution for the modified DSR against
the standard DSR for networks of DF =2 ...........................................................................130

Fig. 5.6: CDF of the packet lost ratio for the modified DSR against
the standard DSR for networks where DF = 2 .................................................................131

Fig. 5.7: Load distribution for a particular topology using
the standard DSR protocol for networks for DF =2 ..........................................................133

Fig. 5.8: Load distribution for a particular topology using
the modified DSR protocol for networks for DF =2 ..........................................................134

Fig. 5.9: CCDFs of the percentage throughput improvement for
DF = 1, 1.5, 2, 2.5, and 3 scenarios [Pz = 512 B, Pr = 5 pps] ..............................................138

Fig. 5.10: CCDFs of the percentage throughput improvement for
DF = 4, 5, 6, 8, and 10 scenarios [Pz = 512 B, Pr = 5 pps] .................................................138

Fig. 5.11: Probability of percentage throughput improvement as a function
of node density factor [Pz = 512 B, Pr = 5 pps] .................................................................140

Fig. 5.12: CCDFs of the percentage delay increment for
DF = 1, 1.5, 2, 2.5, and 3 scenarios [Pz = 512 B, Pr = 5 pps] ..............................................141

Fig. 5.13: CCDFs of the percentage delay increment
for $DF = 4, 5, 6, 8,$ and $10$ scenarios $[Pz = 512$ B, $Pr = 5$ pps] ……………...141

Fig. 5.14: Probability of percentage delay increment as a function of node density factor $[Pz = 512$ B, $Pr = 5$ pps] …………………………….143

Fig. 5.15: Load distribution for the modified $DSR$ using different number of gateways for network of $DF = 2$ …………………...156

Fig. 5.16: CCDF of the percentage throughput improvement for downlink traffic stream and the uplink traffic stream scenarios ……………………………158

Fig. 5.17: CCDF of the percentage delay increment for the downlink traffic stream and the uplink traffic stream scenarios ……………………………158

Fig. 5.18: CCDF of the percentage throughput improvement for downlink and uplink traffic stream and the uplink traffic stream scenarios ……………………………159

Fig. 5.19: CCDF of the percentage delay increment for downlink and uplink traffic stream and the uplink traffic stream scenarios ……………………………160

Fig. 5.20: CCDF of the percentage throughput improvement for TCP traffic against UDP ………………………………………………………………………163

Fig. 5.21: CCDF of the percentage delay increment for TCP traffic against UDP ………………………………………………………………………163

Fig. 5.22: Hop Count against Density Factor ($DF$) using the modified path selection rule based upon the $AEF$ metric ……………………………168

Fig. 5.23: PDF of the Hop Count using the modified path selection rule based upon the $AEF$ metric …………………………………………………169

Fig. 5.24: Hop Count ($Hc$) against Density Factor ($DF$) using $ModAEF$ metric ………………………………………………………………174

Fig. 5.25: PDF of the Hop Count ($Hc$) using $ModAEF$ metric ……………………………………174
Fig. 5.26: Hop Count \((Hc)\) against \(\alpha\) factor for scenarios of \(DF = 2\) ..........................177

Fig. 5.27: Hop Count \((Hc)\) against \(\alpha\) factor for scenarios of \(DF = 4\) ..........................177

Fig. 5.28: Normalised throughput at the gateway node against the time interval

for network of one gateway and \(DF = 1\), one gateway and \(DF = 2\), two

gateways and \(DF = 2\), three gateways and \(DF = 2\) .................................179

Fig. 5.29: Normalised throughput at the gateway node against the time

interval for network of one gateway and \(DF = 2\) when one node is removed

from the network ................................................................. 180
List of Tables

Table 3.1: Comparison among different routing metrics ..........................................52
Table 4.1: Modified DSR Files ..................................................................................97
Table 4.2: DSR parameters .......................................................................................98
Table 4.3: Classification of the simulation scenarios .................................................110
Table 5.1: Average load per node (pps) .................................................................132
Table 5.2: Probability percentage throughput improvement for all
examined scenarios of different $DF$ values .........................................................139
Table 5.3: Probability percentage delay increment for all
examined scenarios of different $DF$ values .........................................................142
Table 5.4: Probability percentage throughput improvement for all
examined scenarios of different $DF$ values .........................................................146
Table 5.5: Probability Percentage delay increment for
all examined scenarios of different $DF$ values ..................................................146
Table 5.6: Probability percentage throughput improvement for all
examined scenarios of different $DF$ values .........................................................148
Table 5.7: Probability percentage delay increment for all
examined scenarios of different $DF$ values .........................................................148
Table 5.8: Probability percentage throughput improvement for all
examined scenarios of different $DF$ values .........................................................152
Table 5.9: Probability percentage delay increment for all
examined scenarios of different $DF$ values .........................................................153
Table 5.10: Probability percentage throughput improvement for all examined scenarios ..........................................................154

Table 5.11: Probability percentage delay increment for all examined scenarios ..........................................................154

Table 5.12: Probability percentage throughput improvement for all examined scenarios of different $Hc$ limits .................................................165

Table 5.13: Probability percentage delay increment for all examined scenarios of different $Hc$ limits .................................................166

Table 5.14: Probability percentage throughput improvement for all examined scenarios of different $DF$ Value .................................................171

Table 5.15: Probability percentage delay increment for all examined scenarios of different $DF$ Value .................................................172

Table 5.16: Probability percentage throughput improvement for all examined scenarios of different $DF$ Value .................................................182

Table 5.17: Probability percentage delay increment for all examined scenarios of different $DF$ Value .................................................183
## Abbreviations and Acronyms

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACK</td>
<td>Acknowledgment</td>
</tr>
<tr>
<td>AODV</td>
<td>Ad-hoc On-demand Distance Vector</td>
</tr>
<tr>
<td>AEF</td>
<td>Access Efficiency Factor</td>
</tr>
<tr>
<td>AP</td>
<td>Access Point</td>
</tr>
<tr>
<td>BLC</td>
<td>Bottleneck Link Capacity</td>
</tr>
<tr>
<td>BSS</td>
<td>Basic Service Set</td>
</tr>
<tr>
<td>CCK</td>
<td>Complimentary Code Keying</td>
</tr>
<tr>
<td>CGSR</td>
<td>Clusterhead Gateway Switch Routing protocol</td>
</tr>
<tr>
<td>CTS</td>
<td>Clear To Send</td>
</tr>
<tr>
<td>CSMA/CA</td>
<td>Carrier Sense Multiple Access with Collision Avoidance</td>
</tr>
<tr>
<td>CW</td>
<td>Contention Window</td>
</tr>
<tr>
<td>CWB</td>
<td>Contention Window Based</td>
</tr>
<tr>
<td>DCF</td>
<td>Distributed Coordination Function</td>
</tr>
<tr>
<td>DF</td>
<td>Density Factor</td>
</tr>
<tr>
<td>DIFS</td>
<td>Distributed Inter-Frame Space</td>
</tr>
<tr>
<td>DREAM</td>
<td>Distance Routing Effect Algorithm for Mobility</td>
</tr>
<tr>
<td>DSDV</td>
<td>Destination-Sequenced Distance-Vector Routing</td>
</tr>
<tr>
<td>DSR</td>
<td>Dynamic Source Routing</td>
</tr>
<tr>
<td>DSSS</td>
<td>Direct Sequence Spread Spectrum</td>
</tr>
<tr>
<td>ETP</td>
<td>Expected Throughput</td>
</tr>
<tr>
<td>ETT</td>
<td>Expected Transmission Time</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>ETX</td>
<td>Expected Transmission Count</td>
</tr>
<tr>
<td>EWMA</td>
<td>Exponentially Weighted Moving Average</td>
</tr>
<tr>
<td>FHSS</td>
<td>Frequency Hopping Spread Spectrum</td>
</tr>
<tr>
<td>FSL</td>
<td>Free Space Loss</td>
</tr>
<tr>
<td>FSR</td>
<td>Fisheye State Routing in Mobile Ad Hoc Networks</td>
</tr>
<tr>
<td>GSR</td>
<td>Global State Routing</td>
</tr>
<tr>
<td>IBSS</td>
<td>Independent Basic Service Set</td>
</tr>
<tr>
<td>IEEE</td>
<td>Institute of Electrical and Electronic Engineers</td>
</tr>
<tr>
<td>IR</td>
<td>Infrared</td>
</tr>
<tr>
<td>ISM</td>
<td>Industrial, Scientific and Medical</td>
</tr>
<tr>
<td>LAR</td>
<td>Location-Aided Routing</td>
</tr>
<tr>
<td>MAC</td>
<td>Medium Access Control</td>
</tr>
<tr>
<td>MAN</td>
<td>Metropolitan Area Network</td>
</tr>
<tr>
<td>MIC</td>
<td>Metric of Interference and Channel- switching</td>
</tr>
<tr>
<td>MIMO</td>
<td>Multiple Input Single Output</td>
</tr>
<tr>
<td>ModAEF</td>
<td>Modified Access Efficiency Factor</td>
</tr>
<tr>
<td>MP</td>
<td>Metric Path</td>
</tr>
<tr>
<td>NAV</td>
<td>Network Allocation Vector</td>
</tr>
<tr>
<td>OFDM</td>
<td>Orthogonal Frequency Division Multiplexing</td>
</tr>
<tr>
<td>OLSR</td>
<td>Optimized Link State Routing Protocol</td>
</tr>
<tr>
<td>OSPF</td>
<td>Open Shortest Path First</td>
</tr>
<tr>
<td>PC</td>
<td>Point Coordinator</td>
</tr>
<tr>
<td>PCF</td>
<td>Point Coordination Function</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Definition</td>
</tr>
<tr>
<td>--------------</td>
<td>------------</td>
</tr>
<tr>
<td>PHY</td>
<td>Physical Layer</td>
</tr>
<tr>
<td>PktPair</td>
<td>Packet Pair Delay</td>
</tr>
<tr>
<td>QoS</td>
<td>Quality of Service</td>
</tr>
<tr>
<td>RTS</td>
<td>Request To Send</td>
</tr>
<tr>
<td>RTT</td>
<td>Round Trip Time</td>
</tr>
<tr>
<td>SIFS</td>
<td>Short time called Short Inter Frame Space</td>
</tr>
<tr>
<td>SPF</td>
<td>Shortest Path First</td>
</tr>
<tr>
<td>TORA</td>
<td>Temporally Ordered Routing Algorithm</td>
</tr>
<tr>
<td>WCETT</td>
<td>Weighted Cumulative ETT</td>
</tr>
<tr>
<td>WLAN</td>
<td>Wireless Local Area Network</td>
</tr>
<tr>
<td>WMN</td>
<td>Wireless Mesh Network</td>
</tr>
<tr>
<td>WPAN</td>
<td>Wireless Personal Area Network</td>
</tr>
<tr>
<td>WRP</td>
<td>Wireless Routing Protocol</td>
</tr>
<tr>
<td>ZRP</td>
<td>Zone Routing Protocol</td>
</tr>
</tbody>
</table>
CHAPTER 1

INTRODUCTION

Wireless networks have become widely used because they provide mobility, flexibility, cost effectiveness, and ease of deployment. Wireless Local Area Network (WLAN) technologies are a type of wireless networks based on the IEEE 802.11 family of specifications that were initially designed by the working group (WG) 11 of the IEEE LAN/MAN Standards Committee [1]. The IEEE 802.11b standard was approved in 1999 and that helped to increase the popularity of wireless LANs [2]. It offers a maximum raw data rate of up to 11 Mbps. The increased throughput offered by IEEE 802.11b compared to the older IEEE 802.11 legacy standard, combined with price reductions, has ensured that IEEE 802.11b has become the most popular Wireless LAN technology.

Wireless Mesh Networks (WMNs) are a consequence of the evolution of the wireless networks in providing functionalities and ease of access to meet growing communication needs. WMNs have a wide range of applications and provide support for applications that are not possible with other existing wireless networks such as cellular networks, wireless sensor networks, Ad Hoc networks etc [3].

At present most of the deployed IEEE 802.11 Wireless Local Area Networks (WLANs) operate in infrastructure mode where a central Access Point (AP) is present. Although channel access in such configurations is decentralised, all traffic in the network flows via
the AP. WMNs overcomes the main drawback of WLAN technology [4]. In WMNs, APs are placed in range of each other to allow them to forward each other’s packets to and from a common gateway. Bandwidth reduction is a main drawback of implementing these technologies [5]. This can be a major problem for users when they share the same wireless medium.

WMNs are generally considered a type of ad-hoc network, as they share common features due to the lack of wired infrastructure. Similar to Ad Hoc networks, each node in WMNs operates as a host and a wireless router. In WMNs, unlike Ad Hoc networks, end hosts and routing nodes are distinct. Routers are usually stationary. WMNs exhibit unique traffic patterns, which partially resemble Ad Hoc networks. Data traffic is tends to flow between users and the network gateway(s). This constitutes the main differentiator between WMNs and Ad Hoc networks [6]. Likewise, in Ad Hoc networks traffic can also flow between any pair of nodes.

WMNs have attracted the attention of networking industries due to their many desirable characteristics such as multi-hop routing, self-configuration, self-healing, self-managing, reliability, and scalability. These characteristics bring many advantages to WMNs such as low up-front costs, easy network maintenance, robustness, delivering reliable services for large variety of applications, and can deliver scalable performance as the mesh can be expanded easily and incrementally as needed [7].

WMNs consist of two types of nodes: mesh routers and mesh clients. Mesh routers have minimal mobility and contain in addition to the routing capability for gateway/bridge
functions additional routing functions to maintain the mesh network [8]. They provide integration with other networks such as the Internet, cellular, etc. and also provide network access for both mesh and conventional clients. Mesh routers are usually equipped with multiple wireless interfaces with the same or different wireless access technologies in order to improve flexibility. Mesh clients can be either mobile or stationary. They can form a client mesh network among themselves and with mesh routers [9]. Mesh clients can also work as a router for mesh networking and are usually equipped with a single wireless interface.

WMN architectures can be classified into three main types based on the functionality of the nodes which are: Infrastructure/backbone WMNs. This type of network is the most commonly used [10], where the end-devices do not participate in the relaying of the packet and the multi-radio relay nodes are part of the network infrastructure. The other type of architecture is client WMNs which is similar to Ad Hoc [11] where client nodes form peer-to-peer mesh network among themselves. In a client mesh network, a mesh router is not required and the end user participates in packet forwarding [12]. Hybrid WMNs is the third type of architecture, this form of network is a combination of infrastructure and client meshing as the end user make up mesh client and mesh router nodes are part of the network infrastructure. In hybrid mesh WMNs both client mesh and backbone nodes have the ability to forward the packets to a destination [13]. This type of architecture is expected to be the best choice in the next generation WMNs [14].

Routing over wireless mesh networks is a complex problem due to the dynamic nature of the link qualities, even when nodes are static. A key challenge in WMNs is the need for an
efficient routing mechanism that determines a path according to certain performance metrics related to the link quality. The routing problem in WMNs is generally concerned with finding a good path between the source and the destination nodes. It generally focuses on multiple objectives to be optimized, such as: path capacity (which refers to the number of bits per second (bps) that can be sent along the path between the source and the destination nodes) and end-to-end delay. The growth of WMNs has resulted in a demand for the development of a high throughput routing metric. Many link quality routing algorithms for WMNs have been proposed, more details about these are given in Chapter 3.

The most widely used routing metric for WMNs for finding the routing path is the hop-count metric. It has been shown that the hop count metric is not an efficient metric for many situations as it does not consider the variability of the wireless link [15]. It ignores the link quality between different wireless links and also does not take into account the interference in the network. For example, in a highly congested network, the hop-count metric will not be the appropriate performance metric. A widely used hop-count protocol is the Dynamic Source Routing (DSR) protocol. The DSR protocol operates on-demand and employs an efficient route discovery mechanism. Route discovery packets are used to determine the route from source to destination. Routed packets contain the address of each node it will traverse in order to get to its destination.

A routing algorithm that takes into account the variability of the wireless link quality is required, since the hop-count metric is not aware of the nature of the wireless link. To achieve this, a cross-layer technique should be employed for routing in order to help in finding reliable and efficient paths to enhance the performance of the network. The cross-
layer approach can be referred to as a protocol design based on actively utilizing the
dependence between protocol layers to enhance the network performance. This is unlike
layering, where the protocols at the different layers are designed independently [16]. The
objective of this technique is to provide the routing layer with view of other layers’
information in order to obtain improvement in the network performance. This work
proposes a cross-layer approach that employs the locally generated MAC layer information
in the network layer in order to find a good route between the source and the destination
nodes in the network. A congestion avoidance technique has been developed by introducing
a new routing metric and path selection rule based on avoiding congested nodes where
packet loss is likely to occur and which will result in a reduced throughput. For this
purpose, in this work, a new access efficiency mechanism \( (AEF) \) metric has been derived
based on MAC bandwidth components framework, previously introduced by Davis et al
[17]. It has been adopted as a local access contention metric at a network node. In this
modification, the selected path is identified by finding a path with the highest minimum
\( AEF (\text{max}\_\text{min}\_AEF) \) value. This choice of path will contain the bottleneck node that is
least likely to become congested. The original intention was to use the \( AEF \) as a measure of
the local bandwidth availability at a node. However, as the research progressed and more
results became available it became apparent that the critical issue determining the WMN
performance is packet loss at a node arising from congestion, i.e. more packets arriving into
the node than were being capable of being transmitted by the node.

The novelty of this work is the development of a new congestion avoidance metric and path
selection rule. In this work, the performance of networks with variable node densities,
transmission ranges, packet size, traffic type, and number of gateways have been examined.
The performance of the modified DSR routing algorithm has been evaluated against the standard routing metric of the DSR protocol. It has also been evaluated against the Metric Path (MP) and the Expected Transmission Time (ETT) metric which was specifically designed for WMNs. The modified AEF-based routing algorithm has shown a significant improvement in the global throughput (defined as the total number of data bits per second received by the gateway node) of the network due to the congestion avoidance mechanism that results in reduced dropped packets at the nodes. Unfortunately, this improvement in the throughput is associated with an increase in delay, which might be considered a drawback of this technique. Avoiding routing through congested areas leads to routing the network load through long transmission paths; hence the end-to-end delay is increased.

To overcome this drawback, another modification to the DSR protocol has been introduced in this work. In addition to the AEF, a hop count limit (HCL) is included in the routing mechanism to control the end-to-end delay time. Tuning the HCL allows the network operator to trade-off throughput against delay time by setting the HCL to an upper limit. In this modification, the selected routes are based on the two criteria. The first is to find a path with the highest minimum AEF in order to maximize the end-to-end throughput by avoiding congestion and hence reduce the packet loss. The second criteria is to limit the hop count to some maximum value that overcomes the shortcoming of increased delay, i.e. it excludes routes whose hop count exceeds the specified HCL. The simulation experiments in detailed in Chapter 5 demonstrate the effectiveness of this proposal.
The choice of using the $AEF$ as congestion metric has an unfortunate drawback owing to its dependency on the packet size. This has the consequence that small packets tend to take longer paths towards the gateway node compared with large packet sizes. This dependency on the packet size has been corrected by developing a modified version of the $AEF$ metric, $ModAEF$, which explicitly considers the size of the packet. A tuning factor $\alpha$ was also introduced to allow the network operator determine the level of the weighting that should be applied to the packet size to correct for this dependence.

The contribution of this work is the development of a simple and effective routing metric ($AEF$) that explicitly considers the local access contention experienced at the node which provides a measure of the local availability of transmission opportunities. When used within the $DSR$ routing protocol in WMNs, the $AEF$ outperforms the standard hop count, the widely used $ETT$ metric, and Metric Path ($MP$) [18] in computer simulations using the OPNET modeler. This is because the new $AEF$ metric path selection rule seeks to avoid directing traffic through congested nodes and operates to route traffic around the congested node. This work introduces a viable alternative routing metric to more traditional link quality based metrics. It also identified the critical role played by access contention is determining routing protocol performance. This new cross-layer $AEF$ metric highlighted the dependence of network capacity on packet size and show how this can be managed within the new $AEF$ metric. The dependence on packet size is not necessarily a shortcoming of the new $AEF$ metric. Since from a network perspective, the capacity of the network will depend on the size of the packets being transmitted where the greater the packet size, the greater the capacity, i.e. the maximum global throughput of the network. This dependence on packet size is also shared by the $AEF$ metric, so in a sense the $AEF$
also captures this dependence which can lead to improved routing decisions. Furthermore, by implementing the $\alpha$ tuning factor in the modified $AEF$ metric, this dependency can be controlled and this can lead to optimized network performance.

1.1 Thesis Organisation

This thesis is organised as follows:

Chapter 2 describes the main technologies used throughout the course of the research by introducing general technical background regarding wireless networks. A brief introduction about some of the IEEE 802.11 technologies for WLANs is given in this chapter. An overview of the MAC specification is introduced as well as a brief discussion regarding the Distributed Coordination Function (DCF) as an access method to the wireless medium. The Access Efficiency Factor ($AEF$) is detailed in this chapter because it is utilised as a metric for the routing discovery mechanism. The $AEF$ is derived from the MAC bandwidth framework also described in this chapter.

Chapter 3 provides an overview of WMNs with some description of their characteristics and architecture. A brief description of several routing metrics and routing protocols is presented as these play an important role in WMNs. Most attention is paid to the Dynamic Source Routing Protocol ($DSR$) as it is the main subject matter of the thesis.

Chapter 4 introduces some description about the OPNET modeler that is used to evaluate the performance of the modified $DSR$ routing algorithm. The modifications to the $DSR$ protocol and simulator setting are also introduced in this chapter as well as the assumptions
used in developing the simulation model. The various different network scenarios examined are described here.

Chapter 5 presents the simulation results with an analysis of the performance of the newly introduced DSR routing algorithm based upon the $AEF$ metric against the standard routing metric of the $DSR$ protocol. An analysis of the performance of the modified version of the $AEF$ ($ModAEF$) metric is also presented in this chapter. A performance evaluation of imposing a hop count limit on the length of the discovered transmission paths is also introduced in this chapter. For further evaluation to the effectiveness of the newly introduced metric, a comparison of the performance of the modified $DSR$ routing algorithm based upon the $AEF$ metric, the $DSR$ routing algorithm based upon the $ETT$, and the modified $DSR$ based upon the Metric Path ($MP$) are given in this chapter. Also the stability of the new metric is considered here.

Chapter 6 presents a summary and conclusion from the work carried out. It also suggests possible areas of further research.
CHAPTER 2

TECHNICAL BACKGROUND

Overview

This chapter presents an overview of the IEEE 802.11 standard which defines Media Access Control (MAC) and Physical (PHY) layer specifications for wireless local area networks (WLANs). In this regard, some explanation about the MAC specification will be given. The IEEE 802.11 MAC specification defines the Distributed Coordination Function (DCF) as an access method for wireless medium and is the method used in this work. The Access Efficiency Factor ($AEF$) is introduced in this chapter as a metric for the routing discovery mechanism which is derived from a MAC bandwidth framework described in this chapter.

2.1 Introduction to IEEE 802.11

The Institute of Electrical and Electronic Engineers (IEEE) ratified the IEEE 802.11 Wireless Local Area Network (WLAN) standard in 1997 [19]. It relates to the group of popular IEEE 802.x standards, e.g., IEEE 802.3 Ethernet [20] and IEEE 802.5 Token Ring [21]. The IEEE 802.11 standard defines Media Access Control (MAC) and Physical (PHY) layer specifications for WLANs. It addresses local area networking where the connected devices communicate over an air interface with other devices that are within reception range of each other. Three different physical layer specifications were defined in the
standard, namely Frequency Hopping Spread Spectrum (FHSS), Direct Sequence Spread Spectrum (DSSS) and Infrared (IR), with a maximum data transmission rate of up to 2 Mbps [22]. The DSSS and FHSS Physical layers operated in the license free 2.4 GHz ISM (Industrial, Scientific and Medical) band while the IR operates in the light frequency spectrum.

In addition to the physical layer specifications defined by the IEEE 802.11 standard, the standard defines two methods for medium access: Distributed Coordination Function (DCF) and the optional Point Coordination Function (PCF) [23]. More details about these methods will be given in the section 2.2.

Two different architectures are defined in the IEEE 802.11 standard which are the Basic Service Set (BSS) and the Independent Basic Service Set (IBSS) [24]. In the former, all the wireless stations are associated with an Access Point (AP) and all communication occurs through the AP. In the latter, all stations within the transmission range of each other can communicate directly without the need for an AP. This kind of architecture is intended to support a wireless ad-hoc network in absence of any network infrastructure. Driven by the demand for higher data transmission rates, the technology has continued to develop with the introduction of new physical layer specifications. A brief introduction will be given regarding some of the IEEE 802.11 technologies for WLANs in the coming sections.

2.1.1 IEEE 802.11b Standard

One of the most popular technologies in the wireless LAN market is the IEEE 802.11b standard. In 1999, IEEE ratified the enhanced Physical layer specification 802.11b which
supports data transmission rates up to 11 Mbps. This popular technology provides low cost wireless Internet capability for end users. The IEEE 802.11b standard specifies the use of DSSS modulation with up to fourteen defined channels. Most commonly, three channels one, six, and eleven, are used because they offer the least amount of frequency overlap. The IEEE 802.11b operates in the 2.4 GHz ISM band with a data transmission rate of up to 11 Mbps with a single carrier per channel. There are four possible transmission rates defined, i.e. 1, 2, 5.5 and 11 Mbps. The IEEE 802.11b standard defines the channel access protocol used at the MAC layer, namely Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) [25]. It has become the most commonly utilized IEEE 802.11 technologies for WLANs to support a wide variety of applications such as video streaming, voice streaming, and file transfer etc. It is designed to cover large areas of up to 100 meters in diameter.

2.1.2 IEEE 802.11a Standard

Following the release of 802.11b revision, the IEEE ratified the amendment on IEEE 802.11a in late 1999. This standard specifies the use of Orthogonal Frequency Division Multiplexing (OFDM) and operates in the 5 GHz ISM band with data transmission rates of up to 54 Mbps [26]. There are 8 rates defined, i.e., 6, 9, 12, 18, 24, 36, 48, and 54 Mbps. But only 6, 12, and 24 Mbps are mandatory with the rest being optional.

2.1.3 IEEE 802.11g Standard

In 2003, the IEEE introduced the IEEE 802.11g standard to address the data transmission rate limitations in IEEE 802.11b [27]. It operates in the 2.4 GHz ISM band using the same modulation technique as IEEE 802.11a (OFDM) with a data transmission rate of up to 54 Mbps [27]. In this specification, additional mechanisms such as Complimentary Code
Keying (CCK) were included to ensure backward compatibility with existing IEEE 802.11b systems.

2.1.4 IEEE 802.11e Standard

In 2005, the IEEE defined another enhancement to the standard called 802.11e by enhancing the MAC sub-layer to improve quality of service (QoS) for better support of video and voice services over WLANs [28]. This standard is common to all IEEE 802.11 PHYs and is backward compatible with the already existing IEEE 802.11 WLANs.

2.1.5 IEEE 802.11h Standard

The IEEE 802.11h is introduced as an enhancement to the IEEE 802.11 in order to satisfy the European regulatory requirements in the 5 GHz band and improve the configuration and the efficient function of WLANs [29]. In this standard, the transmission power control and dynamic frequency selection were included to reduce interference and to meet European Radiocommunications Committee regulatory requirements.

2.1.6 IEEE 802.11n Standard

The IEEE has ratified the IEEE 802.11n standard in September 2009 which supports much higher data rates (> 100 Mbps) than the previous IEEE 802.11 standards [30]. It is achieved by modifying both the PHY and MAC sub-layers using several new features such as Multiple Input Multiple Output (MIMO) technology and channel bonding in 2.4GHz and 5GHz bands.
2.2 IEEE 802.11 MAC Mechanism

IEEE 802.11 MAC specifies two different access methods: The mandatory DCF which uses a distributed, backoff based mechanism for channel access based on CSMA/CA, and the PCF which provides centrally controlled channel access through polling [31].

DCF is the basic mechanism to access the medium which can be used in both infrastructure mode and Ad Hoc mode [32]. Each station in the network contends for access to the medium in distributed manner based on the CSMA/CA protocol. In PCF, access to the channel is determined centrally by the base station, usually referred to as the Point Coordinator (PC) [33]. The PC controls the medium access based on the polling scheme. The PC polls individual stations to concede access to the medium based on their requirements. Stations in this method do not content for the access to the medium and instead the medium access is controlled centrally, the access mechanism is sometimes referred to as contention-free channel access [34]. Only the DCF mechanism is explained in the next section as it is utilized in this work by simulating a IEEE 802.11b radio interface using the OPNET modeler.

2.2.1 Distributed Coordination Function (DCF)

The Distribution Coordination Function (DCF) is defined in the IEEE 802.11 standard as the basic MAC mechanism. The OPNET network modeler employed in this work includes a simulation model of the complete IEEE 802.11 MAC to accurately model the contention of stations for access to the shared wireless medium. The DCF mechanism uses the CSMA/CA algorithm to manage access to the medium. It designed to use both physical carrier sense (performed at the physical layer) and virtual carrier sense (provided by the
Network Allocation Vector NAV at the MAC layer) to reduce the probability of two or more stations attempting to simultaneously transmit a packet on the medium which results in a packet collision occurring [35]. In this algorithm, if the wireless medium is sensed busy by either carrier sense mechanism, the station defers before transmitting. In DCF, data frames are transmitted by two mechanisms, i.e. the basic access mechanism and the Request-To-Send (RTS) and Clear-To-Send (CTS) mechanism [36].

The basic mechanism is mandatory for all IEEE 802.11 implementations. DCF using basic access mechanism can be described as a listen-before-talk mechanism where all stations must contend with each other to access the medium in order to transmit their data [37]. Any station wishing to transmit first listens to the medium during a DCF Inter Frame Space (DIFS). If the medium is busy, the station defers its transmission until the medium becomes idle. When the station senses the medium as idle, it additionally waits for a random backoff interval as a part of the collision avoidance mechanism. The random backoff interval is randomly chosen according to the following formula [38]:

\[
\text{Backoff Interval} = BC \times \text{Slot _Time}
\]  

(2.1)

where \(BC\) is a pseudorandom integer drawn from a uniform distribution over the interval \([0, CW - 1]\) and where \(CW\) is the size of Contention Window. Frame transmission is initiated when a backoff interval reaches a zero value. If the medium becomes busy while a station is decreasing the backoff timer, the backoff procedure is paused and is resumed after the medium is sensed to be idle for an interval of \(DIFS\). When the data packet reaches the destination, the destination station waits for a short time called Short Inter Frame Space (\(SIFS\)). The destination station then sends back an Acknowledgement (\(ACK\)) frame to the
source station to announce a successful transmission. When the medium is busy, all other stations must wait for the channel to become idle. During the busy period, the waiting stations maintain a random backoff interval counter. These stations start decrementing when the medium is sensed idle. The decrementing of the backoff counter is frozen when the medium is sensed busy and is resumed when the medium is free for a time interval of a DIFS. When there is more than one station attempting to transmit, the station with the lowest backoff number wins the medium.

After a successful transmission, a new backoff value is selected and the contention window is set to its minimum value (with a default value of 31 in IEEE 802.11), otherwise the CW value is doubled up to the maximum value (with a default value of 1023 in IEEE 802.11) [39]. Contention window (CW) sizes are always 1 less than an integer power of 2 (e.g., 31, 63, 127, 255, 511, and 1023) [40]. A collision may occur since more than one station may be concurrently attempting to gain access to the medium. When a transmission fails to be positively acknowledged, the size of the contention window CW is doubled, i.e. a new BC value is chosen [41].

Figure 2.1 illustrates this operation. Two stations A and B share the same wireless channel. At the end of the packet transmission by station B, stations B and A wait for a DIFS and then choose a randomly generated backoff time. As can be seen in the figure, station B chooses a backoff counter value equal to 9, before transmitting the next packet, while station A chooses a backoff counter value equal to 4. As the value of the backoff counter of station A is smaller than that of the station B. The backoff counter of station A reaches the value of zero before station B and hence wins the medium for its transmission. Once the
station A starts transmitting, the station B freezes its backoff timer at value 5. When station A finishes transmitting its packet, it sets its backoff counter for a new value after a DIFS. Station B restarts its backoff counter decrement from where it halted prior to station A’s transmission and start transmitting its packet after sensing the channel for a DIFS.

![Figure 2.1: Example of DCF operation.](image)

The main problem with the DCF mechanism when it operates in any WLAN environment is that the medium is shared among the contending nodes where all stations in the network must contend with each other to win access to the medium. The MAC bandwidth components framework [17, 42] can be used to describe how the distributed MAC mechanism allocates the bandwidth of the medium among the contending stations. Under the MAC bandwidth components framework, three parameters are defined which describe how a station utilizes the bandwidth of the medium. These parameters are the load bandwidth $BW_{load}$ which corresponds to the time on the medium used by a station in transporting its load, the access bandwidth $BW_{access}$ which is associated with the contention
mechanism (whereby a station wins access to the wireless medium), and the free bandwidth $BW_{free}$ which represents the medium time currently unused by a station and it is associated with the network capacity experienced by the station. Two of these parameters are used to define the access efficiency factor ($\eta$) which is used as a basis of the modification to the $DSR$ protocol. More details about those components will be given in the next section.

2.3 Access Efficiency Factor ($AEF$) and MAC Bandwidth Components

Based upon the explanation of the basic access mechanism given above, a number of different time intervals on the wireless medium can be defined [17], see Figure 2.2. The definitions can be made based on the busy time and the idle time which is the complementary time interval. The busy time is associated with the transport of the traffic and corresponds to the transmission of frames and their positive acknowledgments. The $T_{busy}^{(i)}$ is defined as the duration of the $i^{th}$ busy intervals within the measurement interval of interest, then the busy time $T_{busy}$ can be written as follows [17]:

$$ T_{busy} = \sum_{i} T_{busy}^{(i)} $$  \hspace{1cm} (2.2)

This interval can be stated in the form of normalized bandwidth as follows [17]:

$$ BW_{busy} = \frac{T_{busy}}{T_{busy} + T_{idle}} $$  \hspace{1cm} (2.3)

Where $BW_{busy}$ represents the portion of the medium bandwidth utilized by all stations in transmitting their loads, i.e.:

$$ BW_{busy} = \sum_{k} BW_{load}(k) $$  \hspace{1cm} (2.4)
$BW_{load}(k)$ is the fraction of time interval on the medium utilized by a station $k$ to transmit its frame. The complementary time interval is the $T_{idle}$ which represents the time that can be used by a station in contending for access to the medium when it has a data or management frame waiting transmission. In the case when the station has no frame to transmit then the idle time is not being used and is considered as free time which can be viewed as spare capacity. This free interval can be used by the station when it is required. The idle time interval $T_{idle}$ is stated as follows:

$$T_{idle} = 1 - T_{busy}$$ (2.5)
Normalizing and converting the idle time interval to a normalized bandwidth as follows [17]:

\[
BW_{idle} = \frac{T_{idle}}{T_{busy} + T_{idle}} \tag{2.6}
\]

\(BW_{idle}\) represents the portion of the bandwidth that is idle and may be exploited by a station to win the access opportunities for its load. In other word, it corresponds to the fraction of the interval time on the medium when no transmission is taking place. During these idle intervals the station may use it to decrement its backoff counter to win transmission opportunities. However different stations use the idle time differently. Consequently, different stations perceive different capacity in the network depending on the load of the specific station and the load of all competing stations. The idle bandwidth consists of two components, an access bandwidth \(BW_{access}(k)\) which represents the time required by a station \(k\) for accessing the wireless medium and a free bandwidth \(BW_{free}(k)\) corresponding to the remaining unexploited idle bandwidth. The idle bandwidth can be stated as follows:

\[
BW_{access}(k) + BW_{free}(k) = BW_{idle} = 1 - BW_{busy} \tag{2.7}
\]

It is possible to associate the transmitted frame with a particular station \(k\) by examining the address fields contained in the MAC header. This can lead to the concept of the load bandwidth \(BW_{load}(k)\) which represents the fraction of the interval time on the medium consumed by a frame transmission from the station \(k\) and can be defined in terms of a bandwidth as follows [17]:

\[
BW_{load}(k) = \frac{T_{load}(k)}{T_{busy} + T_{idle}} \tag{2.8}
\]

Where \(T_{load}(k)\) is the busy duration of the \(i^{th}\) busy intervals on the medium used by a station \(k\) in transmitting its load (and includes collisions) which can be written as follows [17]:
\[ T_{\text{load}}(k) = \sum_i T_{\text{load}}^{(i)}(k) \quad (2.9) \]

Next, we introduce an access efficiency term \( \eta_a \) which is a measure of how efficiently a station utilises the time on the medium to transmit its load. Assuming no hidden nodes, the \( \eta_a(k) \) of station \( k \) can be defined as follows [17]:

\[ \eta_a(k) = \frac{BW_{\text{load}}(k)}{BW_{\text{access}}(k)} \quad (2.10) \]

The station’s capacity is defined as the maximum load bandwidth that can be supported on the medium. In other words, all of the medium idle time is used to win transmission opportunities for the load, i.e.:

\[ BW_{\text{access}}(k) = BW_{\text{idle}} \quad (2.11) \]

That means:

\[ BW_{\text{free}}(k) = 0 \quad (2.12) \]

### 2.3.1 AEF and Station Capacity under Ideal Network Conditions

In this work, a new AEF metric, which is described below in equation (2.16), is derived from the MAC bandwidth components framework that was introduced by Davis et al [17].

In calculating the capacity of an isolated single station at the saturation condition (maximum load that can be supported by the station) when all the free time is used to support the station’s access is given by [17]:

\[ BW_{\text{load}}^{(\text{sat})}(k) + BW_{\text{access}}(k) = 1 \quad (2.13) \]

Substituting (2.10) into (2.13):

\[ BW_{\text{load}}^{(\text{sat})}(k) + \frac{BW_{\text{load}}^{(\text{sat})}(k)}{\eta_a(k)} = 1 \quad (2.14) \]
Equation (2.14) can be rewritten as follows:

\[
BW_{load}^{(sat)}(k) \left( \eta_{a}(k) + \frac{1}{\eta_{a}(k)} \right) = 1
\]  

(2.15)

By defining the access efficiency factor (AEF), \( \eta_f \) as follows:

\[
\eta_f(k) = \frac{\eta_{a}(k)}{1 + \eta_{a}(k)}
\]

(2.16)

The AEF is a measure of how efficiently a station \( k \) contends for access to the wireless medium. The AEF also takes into account the impact of the link errors occurrence. The affect of the retransmissions is to increase average access time due to the doubling of contention window. Equation (2.15) can be expressed as follows:

\[
\eta_f(k) = BW_{load}^{(sat)}(k)
\]

(2.17)

In the Equation (2.17), \( \eta_f \) corresponds to the maximum load achieved by a station under ideal network conditions, i.e. when no other stations are presented. The capacity of a station in the network in the presence of other stations can be calculated as shown in the next section.

The AEF provides an indication of the local contention experienced at a node, which has been implemented in the DSR routine protocol in order to find a route in the WMN capable of avoiding congestion/sustaining high throughput paths. In this work, the AEF and its modified version ModAEF, see section 4.3.3, have been employed as metrics for routing discovery mechanism.
2.3.2 AEF and a Station Capacity in the Presence of other Stations

The capacity of a station $i$ at the saturation condition in the presence of other stations can be computed as explained below. Based on the MAC bandwidth operating plane [42], see Figure 2.3, the capacity of node $i$ can be derived as follows:

$$C(i) = BW_{(\text{sat})}^i = BW_{\text{load}}^i + \Delta BW_{\text{load}}^i$$  \hspace{1cm} (2.18)

Where $\Delta BW_{\text{load}}^i$ is the additional load bandwidth that can be won by the station $i$ from the available free bandwidth of the medium and can be defined as below. In the Figure 2.3, the $\Delta BW_{\text{load}}$ can be derived as follows:

$$\tan \theta = \frac{\Delta BW_{\text{load}}}{A2}$$  \hspace{1cm} (2.19)

Figure 2.3: The MAC bandwidth operating plane description.
Figure 2.3 demonstrates the MAC bandwidth operating plane that is formed in terms of the load and access bandwidth [42]. In this figure, the operating plane of a station is characterized by its position in this plane specified by its \((BW_{load}, BW_{access})\) components. The operating point of the WLAN is also represented in this plane in terms of the \((BW_{busy}, BW_{idle})\) values. The WLAN operating point is constrained to lie along a line. This restriction does not apply to the stations whose operating points \((BW_{load}(k), BW_{access}(k))\) may lie anywhere within the region bounded by \(BW_{busy}\) and \(BW_{idle}\). In this figure, the \(BW_{free}(k)\) component can also be visualised in terms of the distance of the station’s operating point from the \(BW_{idle}\) boundary and can be expressed as follows:

\[
BW_{free} = A2 + \Delta BW_{load} \tag{2.20}
\]

Substituting Equation (2.19) in (2.20) results in the following:

\[
BW_{free} = \frac{\Delta BW_{load}}{\tan \theta} + \Delta BW_{load} \tag{2.21}
\]

Equation (2.21) can be rewritten as follows:

\[
BW_{free} = \Delta BW_{load} \left( 1 + \frac{1}{\tan \theta} \right) \tag{2.22}
\]

In the Figure 2.3, \(\tan \theta\) can be expressed as follows:

\[
\tan \theta = \frac{BW_{load}}{BW_{access}} \tag{2.23}
\]

Employing Equation (2.10), the Equation (2.23) can be rewritten as follows:

\[
\tan \theta = \frac{BW_{load}}{BW_{access}} = \eta_a \tag{2.24}
\]

Using the above equation, Equation (2.22) can be rewritten as follows:

\[
BW_{free} = \Delta BW_{load} \left( 1 + \frac{1}{\eta_a} \right) \tag{2.25}
\]
Based on the Equation (2.25), the $\Delta B_{\text{load}}$ for a station $i$ can be defined as follows:

$$
\Delta B_{\text{load}}(i) = \frac{\eta_a(i)}{1 + \eta_a(i)} \ B_{\text{free}}(i)
$$  (2.26)

Substitute Equation (2.7) in (2.26):

$$
\Delta B_{\text{load}}(i) = \frac{\eta_a(i)}{1 + \eta_a(i)} \ [B_{\text{idle}}(i) - B_{\text{access}}(i)]
$$  (2.27)

The $B_{\text{idle}}$ component can be formulated as follows:

$$
B_{\text{idle}}(i) = 1 - \sum_{k=1}^{N} B_{\text{load}}(k)
$$  (2.28)

Where $N$ is the number of nodes in the network and $k$ is any station in the network.

Equation (2.27) can be written as follows:

$$
\Delta B_{\text{load}}(i) = \frac{\eta_a(i)}{1 + \eta_a(i)} \ [1 - \sum_{k=1}^{N} B_{\text{load}}(k) - B_{\text{access}}(i)]
$$  (2.29)

Substitute Equation (2.29) in (2.18):

$$
C(i) = B_{\text{load}}(i) + \frac{\eta_a(i)}{1 + \eta_a(i)} \ [1 - \sum_{k=1}^{N} B_{\text{load}}(k) - B_{\text{access}}(i)]
$$  (2.30)

The above equation can be expressed as follows:

$$
C(i) = \frac{\eta_a(i)}{1 + \eta_a(i)} \ [1 - \sum_{k=1}^{N} B_{\text{load}}(k) - B_{\text{access}}(i)] + \frac{1 + \eta_a(i)}{\eta_a(i)} \ B_{\text{load}}(i)
$$  (2.31)

Equation (2.31) can be stated as follows:

$$
C(i) = \frac{\eta_a(i)}{1 + \eta_a(i)} \ [1 - \sum_{k=1}^{N} B_{\text{load}}(k) - B_{\text{access}}(i) + B_{\text{load}}(i) + \frac{B_{\text{load}}(i)}{\eta_a(i)}]
$$  (2.32)

i.e.

$$
C(i) = \frac{\eta_a(i)}{1 + \eta_a(i)} \ [1 - \sum_{k=1}^{N} B_{\text{load}}(k) - B_{\text{access}}(i) + B_{\text{load}}(i) + B_{\text{access}}(i)]
$$  (2.33)

Then:
\[ C(i) = \frac{\eta_a(i)}{1 + \eta_a(i)} \left[ 1 - \sum_{k=1}^{N} BW_{load}(k) + BW_{load}(i) \right] \quad (2.34) \]

Equation (2.34) can be presented as follows:

\[ C(i) = \frac{\eta_a(i)}{1 + \eta_a(i)} \left[ 1 - \sum_{\substack{k=1 \atop k \neq i}}^{N} BW_{load}(k) \right] \quad (2.35) \]

Substituting Equation (2.16) in (2.35):

\[ C(i) = \eta_f(i) \left[ 1 - \sum_{\substack{k=1 \atop k \neq i}}^{N} BW_{load}(k) \right] \quad (2.36) \]

In the case when no other stations are present:

\[ C(i) = \eta_f(i) \left[ 1 - 0 \right] = \eta_f(i) \quad (2.37) \]

Equation 2.37 shows that the capacity of the station \( i \) depends only on its access efficiency factor when there is no station competing with it. Winning a sufficient number of transmission opportunities by a station is determined by the presence of other stations in its transmission range as is illustrated in Equation 2.36. Equation 2.36 indicates that the AEF can be considered as a measure of the node capacity. Hence the new AEF metric can be used as an indicator to the congestion which is based on the node’s load and the contention level experienced locally at the node. In this regard, introducing a routing algorithm operating on the basis of choosing a path with the highest minimum AEF will result in avoiding routing through congested regions of the network. The aim of the modification to the DSR routing algorithm in this work is to find high throughput paths by avoiding routing through highly congested nodes by avoiding bottleneck nodes. Initially it was intended to select paths with large capacities to ensure high throughputs. However, subsequent analysis (introduced in Chapter 5) revealed the actual operation of this mechanism involved the
avoidance of congestion. The novelty of this work involves incorporating the new $AEF$ metric with a new path selection rule that leads to select a path containing the bottleneck node that is least likely to become congested.

2.4 Summary

An overview of IEEE 80.11 standards has been presented in this chapter. A brief explanation about Media Access Control (MAC) and Physical (PHY) layer specifications which are defined by the IEEE 802.11 standard for WLANs has been introduced in this chapter. An overview of the Distributed Coordination Function (DCF) (defined by the IEEE 802.11 MAC specification) as an access method for the wireless medium has also been given in this chapter as it is the MAC method employed in this work. Definition of different time intervals on the wireless medium based on the busy and idle times has been introduced through the MAC bandwidth framework described by Davis et al [17]. The MAC bandwidth framework was utilized to introduce the access efficiency factor ($AEF$) as the cost metric for the routing mechanism of the $DSR$ protocol. The derivation of the $AEF$ metric has been demonstrated. The relationship between the node’s capacity (in the presence of other stations), which is a measure of local availability of bandwidth at a node, and the $AEF$ has been also presented in this chapter. The rationale of using the $AEF$ metric in this work is to measure the level of congestion locally at the node. The original intention was to modify the $DSR$ path selection rule by selecting high capacity paths. However, as the research progressed and more results become available it became apparent that the avoidance of congestion is the critical issue in determining the WMN performance.
CHAPTER 3
ROUTING OVERVIEW OF WMNs

Overview
An introduction to WMNs will be given in this chapter with a description of their characteristics and architectures. The chapter will then consider routing which plays a crucial role in WMNs performance. Several routing metrics will be discussed. Finally there will be an overview of the most popular routing protocols designed for wireless networks with particular attention paid to the Dynamic Source Routing Protocol (DSR) as it is the subject matter of this thesis.

3.1 Wireless Mesh Network
The WMN is a relatively new wireless multihop technology which is composed of wireless access points (AP) that facilitate the connectivity and intercommunication of wireless clients through multi-hop wireless paths. The mesh network may be connected to the Internet through gateway routers. The APs are considered as the nodes of the mesh and may be based on different wireless technologies (e.g. Wi-Fi and WiMAX) and connected in a hierarchical fashion. WMNs share a number of common features with Ad Hoc networks [43]. Similar to Ad Hoc networks, each node in the network operates as a host and a wireless router [44]. Unlike Ad Hoc networks, end hosts and routing nodes are distinct.
Routers are usually stationary. A WMN is more reliable and offers greater redundancy compared to an Ad Hoc network. When a node fails, the rest of the nodes can still communicate with each other, directly or through one or more intermediate nodes. Clients can connect to the WMN routers using common networking interfaces (e.g., Ethernet, IEEE 802.11, Bluetooth). WMNs can be implemented with various wireless technologies including IEEE 802.11, IEEE 802.16, cellular technologies or combinations of more than one type. In most proposed applications, the WMN provides connectivity to an infrastructure network, typically connected to the Internet through a gateway. There are different types of mesh network and they can be classified based on their architecture into three types as follows:

**Infrastructure/Backbone WMNs:** The architecture is shown in Figure 3.1, where dashed and solid lines indicate wireless and wired links, respectively. In this architecture, the mesh routers form an infrastructure for clients. Mesh clients are not actively involved in routing and forwarding packets. They gain access to each other through mesh routers which provide a backbone for mesh clients and enables integration to WMNs with existing wireless networks [45]. This is done through the gateway/bridge functionalities provided for in mesh routers if the mesh clients are equipped with the same radio technologies as mesh routers. If different radio technologies are used, clients must communicate with the base stations that have Ethernet connections to mesh routers.
Client WMNs: In this form of architecture, mesh clients provide for a peer-to-peer network among themselves and they are actively involved in routing operations, see Figure 3.2 [47]. In this mesh architecture a mesh router is not required and the mesh nodes perform routing and configuration as well as providing wireless access to end user applications. In Client WMNs, a packet destined to a node in the network hops through multiple nodes to reach the destination. Client WMNs are usually equipped with a single type of radio on devices. Thus, a Client WMN is actually the same as a conventional Ad Hoc network. In comparison to infrastructure meshing, the requirements on end-user devices are increased, since the end-users must perform additional functions such as routing and self-configuration.
Hybrid WMNs: This type of network is the combination of infrastructure and client meshing as shown in Figure 3.3. Mesh clients can communicate directly with each other and can also access the network through mesh routers [48]. In this form of architecture both client mesh and backbone can forward the data to the destination. The infrastructure part of this architecture provides connectivity to other networks such as the Internet, Wi-Fi, WiMAX, cellular, and sensor networks; the routing capabilities of clients provide improved connectivity and coverage inside the WMN.
The main advantage of WMNs compared to the traditional broadband internet access technologies (cable-modem and xDSL) is the dramatically reduced initial investment and deployment time. The main advantage compared to the fixed wireless metropolitan area networks (WMANs) (e.g., IEEE 802.16) is the coverage area (especially in built up urban areas with significant obstructions such as trees, buildings, etc) and reliability (multiple available routes can avoid failed nodes and poor links) [49]. In addition to this, some implementations allow for mobile user access. WMNs overcome one of the important drawbacks of WLAN technology in multi-access point exploitations as it is required to separately provide wired network connectivity to each AP. In WMNs, APs are placed in range of each other to allow them to forward each other’s packets to and from a common gateway. The main drawback of these implementations compared to the infrastructure networks is the reduced network capacity, the nodes need to forward traffic of other nodes in addition to its own traffic. The characteristics of WMNs that have a strong impact on
routing need to be identified. Several advantages of WMNs over competing technologies are listed below:

**Scalability and reliability:** Scalability is a critical issue of WMNs. Theoretically, the more nodes involved the greater the overall performance and reliability of the mesh. Without support of this feature, the network performance degrades significantly as the network size increases. Reliability is an important component in the design and deployment of any communications network. Terminal-pair reliability is an important measure of wireless network reliability. Terminal-pair reliability can be defined as the probability of successful communication between any two terminals in a network [50]. The implemented routing protocol in the network should be able to reroute fast around broken links and failed nodes.

**Network Connectivity:** Several advantages of WMNs originate from mesh connectivity. The procedure of managing network connectivity for maximum reliability and redundancy in the wireless industry is referred to Network Connectivity [50, 51]. To ensure reliable mesh connectivity, network self-organization and topology control algorithms are needed.

**Quality of Service (QoS):** Quality of Service (QoS) is a complex issue in wireless environments due to the significant potential for interference among nodes in relative close proximity to one another. Most applications of WMNs are broadband services with heterogeneous QoS requirements. More performance metrics are required in addition to end-to-end transmission delay and fairness, such as delay jitter, aggregate and per-node throughput, and packet loss ratios, must be considered by routing protocols [51].
**Self-configuration:** One of the characteristics of a WMN is the ability to build and configure itself. Any node joining the network becomes a full member of the mesh topology automatically soon after booting up [51]. The network automatically includes the new node into the existing system with no requirement for a manual configuration. It also makes it self-reconfiguring.

**Self-healing:** This indicates the capability of a mesh network to reorganize itself and remain functioning even if one or more end nodes are removed from the network or moved from one location to another. In a WMN, messages can be sent through an alternative path if a node fails in the network using other nodes. So that human intervention is not necessary for rerouting of messages [52]. Loss of one or more nodes doesn't necessarily affect the network's operation. However, even though WMNs are considered as a special type of Ad Hoc network, there are still significant differences between WMNs and Ad Hoc networks [53]:

**Gateways:** Most WMNs are designed to provide connectivity to mesh clients (usually connected to the Internet). Therefore, they have specialized nodes (the gateways) to form the backbone of WMNs which provide connectivity to the mesh clients.

**Traffic pattern:** The common assumption in Ad Hoc networks is that any node is equally likely to be the source or the destination of a traffic flow. While in WMNs the traffic flow is between mesh clients and the Internet via the gateways.
**Mobility:** Nodes in WMNs are either stationary (e.g., on lamp posts, rooftops, etc.) or mobile which are capable of roaming in the coverage area provided by the stationary nodes.

### 3.2 Routing in Wireless Mesh Networks

One of the important issues for wireless networks is the choice of the routing protocol as it plays an important role in managing the formation, configuration, and maintenance of the topology of the network. In order for nodes to successfully communicate with each other they must gather information regarding the network topology. This is generally achieved either reactively or proactively. A reactive routing protocol establishes a route to a destination on demand. Among the most commonly used reactive protocols are Ad-hoc On-demand Distance Vector routing (*AODV*) [54] and Dynamic Source Routing protocol (*DSR*) [55] both of which employ a minimum hop count.

It has been shown that reactive methods are more successful in terms of throughput and delay time for WMNs if such networks are highly dynamic and nodes are allowed to roam [56]. Proactive routing protocols require periodic propagation of routing information in order that all nodes are able to calculate routes to other nodes, so that when a route is needed it is immediately available [57]. Highly Dynamic Destination-Seqenced Distance-Vector Routing (*DSDV*) [58], Wireless Routing Protocol (*WRP*) [59], Clusterhead Gateway Switch Routing protocol (*CGSR*) [60], Global State Routing (*GSR*) [61], Fisheye State Routing in Mobile Ad Hoc Networks (*FSR*) [62], Optimized Link State Routing Protocol (*OLSR*) [63], are examples of a proactive routing protocol which use periodic broadcasts to discover neighbour nodes.
Designing new routing protocols for WMNs is still an active research topic as new performance metrics need to be discovered and utilized to improve the performance of routing protocols. Finding an optimal routing protocol for WMNs must account for the available bandwidth at a node, link load, packet loss ratio, etc. The routing protocols which have been developed for Ad Hoc networks such as DSR and AODV can be applied to WMNs as they share common features [64]. In addition to these Ad Hoc routing protocols, there are other research efforts that have been conducted into designing new routing protocols to better utilize the special characteristics of WMNs. Some routing protocols are concerned with multi-radio multi-channel routing (routing protocols based on channel selection mechanisms) [65] and others are concerned with hierarchical routing [66, 67]. For example, Kodialam et al have presented channel assignment and routing algorithms that characterize the capacity regions between a given set of source and destination pairs based on the assumption that a radio interface is capable of switching channels rapidly [68]. Raniwala et al have proposed a centralized joint-channel assignment and multi-path routing algorithm based on the traffic loads as they assumed the channel for a radio interface is not switchable and it requires the nodes to maintain channel assignment information of the neighbouring nodes [69, 70]. Alicherry et al formulated the joint channel assignment and routing problem taking into account the interference constraints. In this work, a solution is proposed to optimize the network throughput by allocating the wireless capacity fairly among clients [71].

Some researchers have explored multi-path routing for routing between a source-destination pair. It utilizes the resource redundancy and diversity in the underlying network to provide benefits such as fault tolerance, bandwidth aggregation, load balancing, and
improvement in QoS metrics such as delay. Good examples of this type of protocol are the DSR and AODV protocols. Other routing protocols use hierarchical routing in which nodes are self organized into clusters [72]. Each cluster has a cluster head. The cluster head combining the above information is used to set up a table which contains its cluster members and their connected neighbouring clusters. A cluster member which is connected to another neighbouring cluster is called a cluster gateway; see Figure 3.4 as an example. This type of protocol tends to perform better when the node density is high because of less overhead and shorter average routing length. However, the complexity of maintaining the hierarchy can not be neglected. Furthermore, if the head node of the cluster does not have high processing capabilities, they may become the performance bottleneck. Examples of this type of routing protocol can be found in [73, 74, 75, 76].

![Hierarchical architecture of nodes with cluster heads.](image)

Some other routing protocols classified as geographical protocols which take advantage of node location information. These types of protocols take into account the influence of
physical distances and distribution of nodes to areas as significant to network performance. Geographical routing protocols reduce routing overhead for routing setup and maintenance due to the frequent topology changes. They typically depend on flooding for route discovery or link state updates, which limit their scalability and efficiency [77].

On the other hand, these protocols are efficient in wireless networks as the nodes need to learn only the location information of their direct neighbours in order to forward data. Also, geographical routing has a fast response and can find new routes quickly by using only local information for mobile networks with frequently topology changes. In addition, this type of protocol conserves energy and bandwidth since discovery floods and state propagation are not required beyond a single hop. Examples of this type of protocol are Location-Aided Routing (LAR) [78], Distance Routing Effect Algorithm for Mobility (DREAM) [79], and Zone Routing Protocol (ZRP) [80].

Most routing protocols include at least some periodic behaviour which means protocol operations are performed regularly at some interval despite environment variations [81]. This typically limits the ability of the protocols to adapt to changing environments. When the interval is too short, the protocol will be inefficient as it performs its activities more often than required to react to changes in network topology. When the interval is too long, the protocol will not react sufficiently quickly to changes and packets will be lost [81]. In this work, the DSR protocol has been modified in order to be applied to WMN. The DSR has several advantages over other routing protocols such as its simplicity and efficiency. It operates entirely on demand and is designed mainly to be used in multihop wireless Ad
Hoc networks. WMNs can be considered as a special type of multihop wireless Ad Hoc as they share common features.

3.2.1 Routing Metrics

Routing metrics are used to assign weights to routes by routing protocols to provide measurable values that can be used to determine how useful a route will be. In general, there are several routes between each pair of nodes in a network. Each of which has a different set of links with different throughputs. The route with a high throughput should be selected by the protocol. Routing protocols use route metrics to make decisions about the best route to be selected between a pair of nodes. To perform an efficient route selection, good routing metrics are required for path computation. In order to gain a better understanding of the routing metrics, in this section several routing metrics will be briefly described which can be employed by the routing protocol for wireless mesh networks to find best possible paths. Then a brief overview of the well known reactive and proactive routing protocols used for WMNs will be given with more details about the DSR protocol as it is the subject of this thesis.

3.2.1.1 Hop Count

Hop count represents the number of hops traversed by a packet between its source and destination and it is a widely used as a routing metric for Ad Hoc networks because of node mobility which leads to frequent link breakages [82]. It reflects the effects of the path length on the performance of an end-to-end flow. The path weight equals the total number of links through the path. This metric is used in most of the common routing protocols like AODV [54], DSR [55], and DSDV [58].
However, hop count does not take into account the interference in the network nor the differences of link quality between different wireless links, including the available bandwidth, transmission rates, link load, packet loss ratio, and so on [83, 84, 85]. It may choose paths which have a high loss ratio (the ratio of the data packets originated by the sources fail to deliver to the destination) and poor performance in terms of different metrics such as throughput, number of dropped packets, and end-to-end delay [65].

3.2.1.2 Per-hop Round Trip Time (RTT)

The mechanism of this metric is based on computing the round trip delay observed by unicast probes between neighbouring nodes [86]. The measurement is done by a node sending a probe packet carrying a timestamp to its adjacent nodes every 500 ms [87]. An immediate response will be made by each neighbour sending back a probe acknowledgment. This operation enables the sender to compute the RTT to each of its neighbours. The computed delay time is recorded in the routing table. The selected path by the routing algorithm is the one with the smallest sum of RTTs to routing data packets [88]. The development of this metric was intended to avoid highly loaded links but it can lead to route instability [89]. This metric ignores the interference experienced by the links as well as the link data rates which have an important effect on the performance of the network [90]. Also it doesn’t consider the MAC overheads that are associated with transmitting each single data packet [91]. If either the node or the neighbour is busy, the probe or the acknowledgment packet will experience queuing delay, resulting in high RTT. The RTT metric has some other disadvantages such as the overhead of measuring the round trip time and this technique might not scale to dense networks as every pair of nodes must probe each other [92].
3.2.1.3 Per-hop Packet Pair Delay (*PktPair*)

*PktPair* metric was introduced to overcome the limitations associated with the *RTT* metric due to queuing delays. This metric operates by sending a small packet of size 137 bytes ahead of a large packet of size 1000 bytes. It computes the delay between a pair of back-to-back probes to an adjacent node by sending a small and big packet in sequence. This adjacent node calculates the delay between the receipt of the first and the second packets. Then it feeds back this calculated delay time to the sender node. The sender maintains an Exponentially Weighted Moving Average (*EWMA*) of these delays for each of its immediate neighbours. This average is employed by the routing algorithm as the cost metric for the link. The objective of using a pair of successive probe packets eliminates the effect of queuing delays [93, 94].

The main advantage of the *PktPair* metric over the *RTT* is that it is not affected by queuing delays at the sending node [87]. Since both packets in a pair will be delayed equally. In addition, using a larger packet for the second probe makes the metric more sensitive to the link bandwidth than the *RTT* metric. This metric is load-dependent and hence should vary with offered traffic load [87]. The main advantage of this is the ability of differentiate between high and low bandwidth links which occur frequently owing to the use of heterogeneous radios or variable link quality and rate control algorithms [95]. The mechanism of sending a pair of packets in sequence to each immediate neighbour make the *PktPair* metric subject to overheads even higher than those of the *RTT* [87]. This metric also is not immune to the self-interference phenomenon (this phenomenon is produced by different packets of the same flow competing for medium access at different nodes [96]).
due to the contention between the probe packets and the data packets for the wireless channel.

3.2.1.4 Expected Transmission Count (ETX)

This metric was the first metric proposed for WMNs hence it explicitly accounts for link quality during path selection [97]. The ETX metric estimates the expected number of attempts required to successfully transmit a packet on a link, for further details about ETX we refer to [98]. The ETX finds the route with the highest probability of successful packet delivery, as an alternative to the shortest path. They are some drawbacks associated with this metric: ETX does not differentiate between links with different capacities as IEEE 802.11 broadcast frames are sent at the network basic physical rate and probe packets are usually smaller than data packets [99]. Also the loss probability of small probe packets differs from the loss probability of data packets [99]. The ETX is calculated for each node in the network by periodically broadcasting probe packets containing the number of received probes from each neighbour.

The ETX of a link is determined by using the forward delivery ratio $P_f$ of probes (the probability that a data packet successfully arrives at the recipient) and reverse delivery ratio $P_r$ (the probability that the ACK packet is successfully received) over a link between two nodes in the network. The expected probability that a transmission is successfully received and acknowledged is $(P_f \times P_r)$. The expected number of transmissions is given as [98]:

$$ETX = \frac{1}{P_f \times P_r}$$

(3.1)
The *ETX* computation considers both forward and reverse directions because of data and acknowledgment frame (*ACK*) transmission. The selected path is the one with the minimum sum of *ETX*s along the path to the destination.

Figure 3.5: Estimated transmission count (*ETX*) to node \(D\) from each node.

In the Figure 3.5, each node’s *ETX* value is the sum of the link *ETX* value along the lowest-*ETX* path to the destination node \(D\). As one can see in the figure, the node \(S\) will select the path 2 (\(S, C, D\)) to route its data packet to the destination \(D\).

The implementation of *ETX* has highlighted some shortcomings, namely that broadcasts are usually performed at the network basic rate and that probe packets (approximately 60 bytes [100]) are smaller than typical data packets. Thus, unless the network is operating at low rates and low packet sizes, the use of *ETX* is ineffective because it neither distinguishes links with different bandwidths nor considers data-packet sizes [101]. It only considers the link loss ratio and does not capture the interference experienced by the links which has a
significant impact on the link quality and the data rate at which packets are transmitted over each link [102]. That may not lead to good paths when the links vary. It also does not take into account the load of the link which means it might route the traffic through heavily loaded node, i.e. discover a route through a congested network area [103].

3.2.1.5 Expected Transmission Time (ETT)

To overcome the drawbacks associated with ETX, the Expected Transmission Time (ETT) metric was proposed. These two metrics were designed specifically for WLANs. ETT is the product between ETX and the average time \( t \) a single data packet requires to be transmitted \((ETT = ETX \times t)\). This time \( t \) can be calculated by dividing a fixed data-packet size \( S \) by the actual link data rate \( B \), then [104]:

\[
ETT = ETX \times \frac{S}{B}
\]

(3.2)

ETT metric uses a periodic broadcast procedure by probing the network with packets of two sizes. The small packet sizes are always transmitted at basic rate (1 Mbps) which corresponds to ACKs [105]. The large packet sizes are transmitted at various rates and correspond to data. This means that large packets can be broadcast at different rates based on the used IEEE 802.11 technology. For example, when using IEEE 802.11b large packets will be broadcast at four different rates (1, 2, 5.5, and 11Mbps) [88]. The ETT depends on the loss rate and the bandwidth of each link. The selected path is the one with the lowest sum of ETT values, to learn more about ETT see [104, 106]. It has shown that the ETT metric is more effective than ETX, but it does not capture the interference that might be caused by a single link with high loss rate along a path, which can cause a dramatic reduction in the overall path performance [107]. Some other drawbacks with this metric are
that it does not consider the MAC overhead delays [108]. The main shortcoming of this metric is that it does not take into consideration the contention arises from other nodes competing for access to the wireless medium [109]. It also does not consider the load on the link, therefore it can not avoid routing through heavily loaded nodes, i.e. highly congested nodes, which leads to unbalanced resource usage [110].

3.2.1.6 Weighted Cumulative ETT (WCETT)

The Weighted Cumulative Expected Transmission was proposed to optimize the capacity of the transmission path and the end-to-end delay by finding paths with less intra-flow interference (interference between nodes on the path of the same flow) [111]. This metric is a sum of end-to-end delay and channel diversity. It computes an end-to-end value by taking into consideration all channels used along the route in order to avoid intra-flow interference [112]. The WCETT metric of a path \( p \) is defined as follows:

\[
WCETT_p = (1 - \alpha) \times \sum_{i \in p} ETT_i + \alpha \times \max_{i \in j \in k} X_j
\]

(3.3)

Where \( X_j \) is the sum of the ETT values of links which are on channel \( j \) in a system which has \( k \) orthogonal channels. The first component of the equation estimates the end-to-end delay experienced by a packet travelling along a path by accumulating the individual link ETTs. Therefore, it generally favours shorter high quality paths. The second component of the equation observes the impact of channel diversity. This is achieved by accumulating the ETTs of all links of a given channel and then takes the maximum over all channels. This will ensure low intra-flow interference. Adopting the \( \alpha \) parameter (within the bound \( 0 \leq \alpha \leq 1 \)) is to trade-off the path length against channel diversity, for further information about this metric see [113].
Like ETX and ETT, WCETT also does not consider interflow interference (interference between different flows that have neighbouring links) [114]. This may lead this metric to route the traffic through congested areas which results in performance reduction. The main disadvantage of WCETT is that it does not favour channel reuse [115]. WCETT, like ETX and ETT, neglects link load or link congestion when establishing paths and it also does not guarantee shortest paths [104, 116].

### 3.2.1.7 Metric of Interference and Channel-switching (MIC)

Metric of Interference and Channel-switching (MIC) takes into consideration the shared nature of wireless channels and utilizes the extra resources available from multi-radio/multichannel nodes [117]. MIC is a combination of two metrics: Interference-aware Resource Usage (IRU) and Channel Switching Cost (CSC), see [117]. Each of which reflects different characteristics of mesh networks. MIC for a path \( p \) is defined as follows [118]:

\[
MIC(p) = \frac{1}{N \times \min(ETT)} \sum_{\text{link } i \in p} IRU_i + \sum_{\text{node } i \in p} CSC_i
\]  

(3.4)

Where \( N \) is the total number of nodes in the network. The two components IRU and CSC are defined as follows:

\[
IRU_i = ETT_i \times N_i
\]  

(3.5)

\[
CSC_i = \begin{cases} 
  w_1 & \text{if } CH(\text{prev } (i)) \neq CH(i) \\
  w_2 & \text{if } CH(\text{prev } (i)) = CH(i)
\end{cases}
\]  

(3.6)

\[
0 \leq w_1 < w_2
\]  

(3.7)
Where $N_i$ is the set of neighbours that interfere with the transmissions on link $l$. $CH(i)$ represents the channel assigned for node $i$’s transmission and $\text{prev}(i)$ represents the previous hop of node $i$ along the path $p$. The relationship $w_1 < w_2$ captures the intraflow interference [118]. The first component of Equation (3.4) captures the interflow interference, the transmission rates, and loss ratio of wireless links. While the second component of the equation captures the influence of intraflow interference in two consecutive links [118, 119].

Although MIC provides better throughput and delay performance compared to the existing routing metrics, it suffers from high overhead. This is due to the requirement of updated information on the $ETT$ for each link which can significantly affect the performance of the network. It makes an assumption that all links located in the collision domain of a particular link contribute to same level of interference [120]. It estimates the amount of interference on a link only by the position of interfering nodes no matter whether they are involved in any transmission simultaneously with that link or not [121]. The other drawback with this metric is that it does not capture the link loss ratio, data rate of the link in the absence of interfering neighbors, and makes no consideration to the load balancing [122]. The $IRU$ component of the $MIC$ metric also assumes that a link will always contend with neighboring nodes regardless of their current activity. This will lead to routing traffic around the edge of the topology where nodes have fewer neighbors and hence create longer, slower paths [119]. It has also been realized that the intra-flow interference measuring does not take into account exact phenomenon of carrier sense on wireless links. They provide some ideas to address this, but conclude that the benefit gained is not worth the extra complexity [119].
3.2.1.8 Expected Throughput (ETP)

ETP is a MAC-aware routing metric which takes into account the reduction in the capacity of a link due to its contention with neighbouring links located in its transmission domain. This metric focuses on the intraflow contention [123]. ETP finds better routes than ETX and ETT in mesh networks with long paths as these two metrics do not make spatial measurements [123]. This metric predicts better routes in mesh networks with heterogeneous link rates than ETX and ETT. This is because ETP captures the bandwidth sharing mechanism of IEEE 802.11 DCF more accurately than these metrics and also these metrics do not take into account the throughput reduction of fast links due to contention from slow links [123]. The ETP of a link \( k \) can be defined as follows:

\[
ETP(k) = \frac{p_k^{(f)} \cdot p_k^{(r)}}{b_k}
\]  

(3.8)

\( p_k^{(f)} \) and \( p_k^{(r)} \) are the packet success probabilities of link \( k \) in the forward and reverse directions respectively. Where \( b_k \) is the expected bandwidth received by a link \( k \) in the path \( P \) and can be defined as follows [123]:

\[
b_k = \frac{1}{\left( \sum_{j \in S_k \cap P} \frac{1}{r_j} \right)}
\]  

(3.9)

\( S_k \) is the contention domain of the link \( k \) and represents the set of all the links in the network that preclude a transmission on link \( k \). Then, \( S_k \cap P \) is the set of links on path \( P \) that contend with link \( k \). The \( r_k \) is the nominal bit rate of link \( k \). In the form of ETX, the ETP can be formed as follows:

\[
ETP(k) = \frac{1}{ETX_k \cdot b_k}
\]  

(3.10)
The throughput of the bottleneck link of a path can be computed as follows:

$$f(P) = \min_{K \in P} ETP(K)$$  \hspace{1cm} (3.11)

The routing strategy is to find the path with the highest routing metric $f(.)$. The $ETP$ has a more accurate model for the impact of contention in IEEE 802.11 MAC than $ETX$ and $ETT$. The drawback of this metric is that it does not consider MAC contention between different flows, i.e. it does not take into account the interflow interference [123]. This metric makes a conservative estimate for long paths. It does not consider the impact of node’s loading on the performance of a path [124].

### 3.2.1.9 Bottleneck Link Capacity ($BLC$)

The $BLC$ metric is based on the estimation of the Expected Busy Time ($EBT$) of a successfully transmitting a packet on a link [125]. Using the transmission mechanism in the MAC layer and the Packet Loss Ratio ($PLR$), the $EBT$ can be computed. The residual capacity of a link is defined as the ratio between the idle time and $EBT$. Considering a path $P$, if the residual capacity of a link $i$ is $LC_i$, then $BLC$ is introduced as follows [125]:

$$BLC_P = \frac{\min_{i \in P} LC_i}{\mu^K}$$  \hspace{1cm} (3.12)

Where $K$ is the length of the routing path $P$ and $\mu$ is a fine-tuning parameter larger than 1. The $BLC$ metric is an indicator to the residual capacity of the bottleneck link of a routing path. This metric penalizes the long routing path as it is shown in the equation above through the division of the minimum residual capacity by $\mu$ parameter. The $BLC$ metric considers load-balancing in links by considering the busy time in its calculation. This metric does not consider the self-interference of a routing path as the minimum residual capacity is considered in $BLC$. In other words, if two routing paths have different self-
interferences, then the bottleneck link can have the same residual capacity [126]. The same problem applies to interference from other routing paths.

3.2.1.10 Metric Path (MP)

A cross-layer routing metric has been introduced in this work that takes into account available bandwidth \((AB)\) as well as the number of retransmissions \((NR)\) to improve the WMN performance [18]. The number of retransmissions \((NR)\) can be set to 0 as an approximation when the network is below saturation, i.e. almost all the packets get transmitted successfully in the first attempt. When the link quality is poor, retransmission attempts is required which is carried out by MAC protocol. Suppose there are packets from the source node \(S_i\) to the destination node \(S_j\), there is a path which can be defined as \(q_{i,j}\). This \(q_{i,j}\) can be found easily from the route reply information. Now the \(ANR\) of the \(q_{i,j}\) can be defined as follow [18]:

\[
ANR_{path} = \frac{\sum_{k} NR(S_k)}{\text{hopnumber}_{path}}
\] (3.13)

Where \(k \in i \rightarrow j\), \(\text{hopnumber}_{path}\) is the current number of hops from the source node \(S_i\) to the destination node \(S_j\). Suppose there is a path from the source node \(S_i\) to the destination node \(S_j\), and the nodes on this path are \(S_i, S_{t1}, S_{t2}, S_{t3}, \ldots, S_{tn}, S_j\). The available bandwidth \((AB)\) of each hop of this path can be measured by sending the probe packet every \(T\) seconds. The bottleneck of a path is the least available bandwidth \((LAB)\) of the path which can be defined as follow [18]:

\[
LAB_{path} = \min (AB_{t_1,t_2}, AB_{t_1,t_2}, \ldots, AB_{t_{n-1},t_n}, AB_{tn,j})
\] (3.14)

Where \(t_1,t_2,\ldots,t_n \in i \rightarrow j\), \(AB_{tn-1,tn}\) is the available bandwidth of the hop from the node \(S_{n-1}\) to the node \(S_n\). With the above simplifications, the introduced routing metric computation can be summarized as follows [18]:

50
This work introduces a routing metric based on the cross-layer mechanism in wireless mesh networks which is based on the end-to-end delay and the available bandwidth. The proposed metric takes into account information from the other layer, it will help to find a relatively reasonable path. Simulation results demonstrated that it can improve the system throughput no matter if it is in stationary or mobile scenarios due to selecting paths with high available bandwidth while also avoiding areas of MAC congestion. Using such a technique as a congestion measure highlights some shortcomings. Using the number of retransmission attempts is a poor measure of congestion. The number of transmission attempts could be used as an indication of the link quality where the larger the number of retransmission attempts the lower the quality of the link. However, increasing the number of the retransmission attempts could lead to an increase in the possibility of node congestion, but this will depend on the on the number of packets which arrive at the node and number of packets transmitted by the node within a unit time. That means increases in the number of retransmission attempts does not necessary lead to the node to be congested. This routing metric also uses the available bandwidth as indication of the node congestion. Once again, taking the available bandwidth as a measure to the congestion is a poor node congestion indication. A link with a low available bandwidth will not necessarily lead to node congestion. Based on the above, the $MP$ metric takes no explicit consideration of the interference experienced by the nodes. The implemented path selection mechanism in this work selects the longer path over the shorter one, due to the implementation of the $hopnumber_{path}$ parameter, which is considered as another shortcoming of this metric.
3.2.2 Comparison of Metrics

In this section, a comparison among some of congestion related routing metrics has been introduced following the literature review. The comparison in presented in the following table in order to highlight the features of each routing metric used for the routing decision.

This comparison shows the parameters used by the routing metrics, the protocol that it was implemented in, and the path selection rule used. It also indicates if the metric takes into account parameters such as local congestion at a node, local contention, and hop count.

<table>
<thead>
<tr>
<th>Table 3.1 Comparison among different routing metrics.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Author</strong></td>
</tr>
<tr>
<td>A Congestion and Interference Aware Routing Metric for Wireless Mesh Networks [127] (J. Zhu, 2008)</td>
</tr>
<tr>
<td>A Congestion-Aware Multipath Routing with Cross Layer Design for Wireless Mesh Networks [128] (W. Song, 2009)</td>
</tr>
<tr>
<td>A Study of Congestion-aware Routing Protocols for Wireless Ad hoc Networks [129] (W. Wei, 2008)</td>
</tr>
</tbody>
</table>
- MAC Overhead, Channel Delay, Buffer Occupancy (BO)
- Weighted channel delay (WCD)
- BO is a poor indication to node congestion. Difficult to determine the optimal weighting factors.

- Average number of transmission opportunities (Instantaneous MAC utilization) & Instantaneous queue length
- Queue Length plus MAC Utilization Threshold
- Indirect (via BO)
- BO is a poor indication of node congestion. Number of TXOPs is a poor indication of node congestion. Difficult to determine the optimal threshold of the congested level

### Hop-Count Based Congestion-Aware Multi-path Routing in Wireless Mesh Network [132] (H. Q. Vo, 2006)
- Packet Rate, Data Rate, Estimated time to transmit a packet
- Weighted sum (W) of Packet Rate, Data Rate, Estimated time to transmit a packet
- Indirect (via W)
- No
- Estimated time of transmitted load is a poor indication of congestion. Combination between link bandwidth and estimated time of transmitted load to be used as indication to congestion is poor. Difficult to determine the optimal threshold of W.

### Routing with Congestion Control and Load Balancing in Wireless Mesh Networks [133] (W. Song, 2006)
- Data retransmission RTSFailureCount & ACK retransmission ACKFailureCount & RTS retransmission RTSFailureCount
- Weighted Channel Usage (Data Retransmission, ACKFailureCount, RTSFailureCount)
- No
- Indirectly (via ACKs and RTSs)
- Yes
- Data and ACK retransmission are poor indications of the node congestion as retransmissions are not necessary lead to node congestion. Difficult to determine the optimal Channel Usage threshold.

- Average number of transmission opportunities (Instantaneous MAC utilization) & Instantaneous queue length.
- Queue Length plus MAC Utilization Threshold
- Indirect (via BO)
- Indirect (via MAC utilization)
- No
- BO is a poor indication of node congestion. Number of TXOPs is a poor indication of node congestion. Difficult to determine the optimal threshold of the congested level

### A Link-Quality and Congestion-Aware Cross layer Metric for Multi-Hop Wireless Routing [135] (G. Karbaschi, 2005)
- Number of Retransmission Attempts and hop count (Hc).
- ANR / LAB
- No
- Indirect (via Number of retransmissions)
- Yes
- Number of retransmission attempts is a poor metric to be used as a congestion measure.
In order to measure congestion locally at a node, two measurements are required which are a measure to the number of available transmission opportunities at a node and the forwarded traffic to the node (i.e. measuring how many packet arrived at the node and how many packets are leaving the node within a unit time). The current routing metrics fail to account for these parameters. The above routing metrics utilise either buffer occupancy or retransmission attempts as a measure for node congestion. Buffer occupancy is a poor metric to use because the buffer tends to fill and empty rapidly. Buffer occupancy is not a reliable metric for congestion because high buffer occupancy does not necessarily indicate node congestion. The number of retransmission attempts gives an indication of link quality but generally would not give a reliable indication to the node congestion. Excessive number of retransmission attempts may lead to node congestion, but this will depend on the number of packets entering and leaving the node within a unit of time.

The metric path \((MP)\) has been chosen for a routing performance comparison against the new \(AEF\) metric as the strategy of the \(MP\) and \(AEF\) metrics is to avoid congestion regions in the network by taking into account the bandwidth availability. The main difference between these two metrics is that the new \(AEF\) takes into account the local availability of the bandwidth at a node by providing a measure of the access contention at a node, while
the MP metric considers the link bandwidth and retransmission attempts along the path between the source-destination pair.

### 3.2.3 A New Congestion Aware Routing Strategy

Designing routing metrics is a critical issue for WMNs performance. The unique combination of static nodes with the shared nature of the wireless medium in mesh networks imposes specific requirements for the design of routing metrics. Routing metrics defined by the protocols are responsible for establishing the paths in the network. In the previous section, a review of several link quality routing metrics for wireless networks has been presented which highlight the advantages and disadvantage of each metric. The main drawback of these metrics is that they fail to account for local access contention at a node which an important factor in the cause of congestion. There is a need for a routing metric that can capture the congestion experienced locally at a network node as congestion can give rise to significant packet loss at a node. The proposed path selection rule based upon the $AEF$ metric provides an indication of the local congestion at a node by taking into account the load of the node and the access contention experienced by the node. The newly introduced $AEF$ routing metric routes network packets away from the congested region where packet loss is likely to occur.

### 3.2.4 Overview of Routing Protocols

WMNs are different from other architectures such as WLANs and WMANs. These network architectures utilize a single wireless link and hence have no need for a network layer [136]. While in WMNs and Ad Hoc networks the source and the destination nodes can be several wireless hops away from each other. Thus, the routing protocol is an important
factor in any WMN as it affects the entire performance of the network. In designing routing protocol for WMNs some factors should be taken into account such as interference, load distribution, avoiding congested regions, etc. which have a direct affect on the performance of WMNs. It has been shown that proactive routing protocols work well for wired networks as they provide up-to-date state information for all nodes in the network [137]. However due to the overhead associated with updating the information they scale poorly in WMNs. While the reactive routing protocols perform well in wireless environments due to continuously changing topology [138]. Also on-demand mechanisms can help bandwidth conservation as the bandwidth resource is scare in wireless environments. Therefore, reactive protocols have been widely adopted for WMNs [139]. A considerable number of routing protocols have been developed for Ad Hoc networks which can also be applied to WMNs. An overview of some of the more commonly used protocols such as proactive, Hybrid, and reactive routing are discussed in the following sections. The emphasis on the work presented in this thesis is on DSR protocol and will be discussed at the end of the section.

3.2.4.1 Destination Sequenced Distance Vector (DSDV) Routing

DSDV was one of the first proactive routing protocols introduced for Ad Hoc networks. DSDV is a table-driven protocol based upon the classical distributed Bellman-Ford algorithm [140] used in wired networks by including freedom from loops in routing tables [141]. It uses destination assigned sequence numbers to avoid the traditional counting to infinity problem associated with distance vector algorithms. Every mobile node in the network supports a routing table. This routing table holds the address of next hop, remaining hop count to the destination, and the sequence number of the last route
advertisement for that route. Each entry of the routing table is marked with a sequence number assigned by the destination node [142]. Nodes periodically transmit routing table updates throughout the network in a dynamically varying topology to maintain consistent tables. The sequence number was used in this protocol to avoid formation of routing loops as it enables the mobile nodes to distinguish inactive routes from new ones [141, 142, 143].

In this protocol, mobile nodes are periodically broadcast routing table updates using one of two different types of update packet. One is called “full dump” packet which carries the full routing table of a node. It can require multiple Network Protocol Data Units (NPDUs) when the routing table is large. This type of packet is transmitted infrequently to conserve network resources if the node experiences limited topological changes in relation to its neighbours. Smaller “incremental” packets represent the other type of packet. This type of packet is broadcasted to provide only that information which has changed since the last full dump was sent out by the node [142].

The mobile nodes maintain an additional table where they store the data sent in the incremental routing information messages [141, 142]. Any new learned routes will immediately be advertised by a node, and updated routes will cause an advertisement to be scheduled for transmission within a certain settling time (the time between the first route with a new sequence number and the shortest route) [143]. New route broadcasts carry the address of the destination, the number of hops to reach the destination, the sequence number of the information received regarding the destination, as well as a new sequence number unique to the broadcast [143]. The route labelled with the most recent sequence number is always used.
The mechanism of this protocol operates on the basis that each node in the network maintains a preferred neighbour for each destination. Every data packet carries in its header a destination node identifier. The received data packet by a node is forwarded to the preferred neighbour for its destination. The forwarding process continues until the packet reaches its destination [143]. In the case of two updates that have the same sequence number, the route with the smaller metric is selected to optimize the path. Moreover, nodes also keep track of the settling time of routes, or the weighted average time that routes to a destination will fluctuate before the route with the best metric emerges [144]. By delaying the broadcast of a routing update by the length of the settling time, nodes can reduce network traffic and optimize routes by eliminating those broadcasts that would occur if a better route was discovered in the near future [145]. One of the drawbacks of this metric is regularly updating of the routing table which consumes the available bandwidth even when the network is idle in addition to the power consumption caused by the periodic operation [140, 146]. In addition, this protocol is not appropriate for highly dynamic networks since a new sequence number is necessary for every topology changing [140]. The DSDV is suitable for Ad Hoc networks with small number of nodes.

3.2.4.2 Open Shortest Path First (OSPF)

OSPF is a link-state protocol in which routers send each other information about the direct connections and links which they have to other routers. It is designed to support routing in TCP/IP networks [147]. A router running OSPF maintains an identical database describing the topology. A routing table is calculated by constructing a shortest-path tree using the maintained database by the router. The OSPF algorithm [148] is a specification of a
hierarchical algorithm based on Dijkstra’s Shortest Path First (SPF) algorithm [149]. It is a link-state routing protocol that calls for the sending of link-state advertisements (LSAs) to all routers within the same hierarchical area. Routers use the SPF algorithm to calculate the shortest path to each node.

The OSPF routing protocol is composed of three algorithms: the Hello, Election, Flooding and Shortest-Path-First (SPF) [150]. The Hello, Election, and Flooding Protocols distribute and synchronize routing information within an autonomous system [147]. The first mechanism is finding neighbors. To do this, OSPF sends a "Hello" packet to each neighbor. Among the things in this packet is a list of neighbors from which the sender has recently received a Hello message. The Shortest-Path-First algorithm computes the shortest-path tree for each route using a method based on Dijkstra's algorithm [151]. In the Election algorithm, a Designated Router (DR) and a Backup Designated Router (BDR) are elected to distribute and synchronize topology information among routers on a broadcast network. The DR mechanism is used to reduce the number of the broadcasted messages needed to deliver topology information and hides this information from other routers within the autonomous system [147, 150]. The Flooding mechanism ensures that all routers within an area have identical topology information for that area. Topology information is exchanged between each pair of neighbouring routers in order to learn the most recent topology changes within the autonomous system. Using this mechanism, a router can obtain the new information by synchronizing its topology database with neighbouring routers [147].

In the OSPF, the best route is chosen by finding the lowest cost paths to a destination. All router interfaces (links) are given a cost. Each interface running OSPF is assigned a cost,
which is a unitless number based on factors such as throughput, round-trip time, and reliability, which are used to determine how easy or difficult it is to reach a destination. If two or more routes to a destination have the same cost, OSPF distributes traffic equally among the routes, a process that is called load balancing [152]. The cost of a route is equal to the sum of all the costs configured on all the outbound links between the router and the destination network.

This protocol enables the flexible configuration of IP subnets. Each route distributed by OSPF has a destination and mask [150]. Two different subnets of the same IP network number may have different sizes (i.e., different masks). This is commonly referred to as variable length subnetting [150]. A packet is routed to the best (i.e., longest or most specific) match. OSPF allows sets of networks to be grouped together. Such a grouping is called an area. The topology of an area is hidden from the rest of the system. This information hiding enables a significant reduction in routing traffic. Also, routing within the area is determined only by the area's own topology, lending the area protection from bad routing data [152].

3.2.4.3 The Wireless Routing Protocol (WRP)

The WRP is a table-based protocol with the goal of maintaining routing information among all nodes in the network [153]. It adopts a concept of second-to-last hop node to a destination. The algorithm of this protocol utilizes information about distance and second-to-last hop (predecessor) along the path to each destination [154]. Path-finding algorithms avoid the counting-to-infinity problem of distributed Bellman Ford algorithms by using that predecessor information, which can be used to infer an implicit path to a destination and
thus detect routing loops [155]. In this protocol, each node is responsible for maintaining four different tables which are distance table, routing table, link-cost table, and message retransmission list (MRL) table. The distance table of a node $A$ carries the distance of each destination node $B$ via each neighbour $C$ of $A$. It also contains the downstream neighbour of $C$ through which this path is realized. The Routing table of node $A$ contains the distance of each destination node $B$ from node $A$, the predecessor and the successor of node $A$ on this path. This table also contains a tag to identify if the entry is a simple path, a loop or invalid. The idea of listing predecessor and successor in a table is useful for avoiding counting-to-infinity problems and loops [156]. The task of the link-cost table is to store the cost of link to each neighbour of the node and the number of timeouts since an error-free message was received from that neighbour. The $MRL$ entries contain information such as the sequence number of the update message, a retransmission counter, an acknowledgement-required flag with one entry per neighbour, and a list of updates sent in the update message. This information is used to inform a node about which of its neighbours has not acknowledged its update message and to retransmit update message to that neighbour. The information is passed among a node and its neighbours by exchanging these routing tables using update messages [156]. Update messages contain a list of updates (the destination, the distance to the destination, and the predecessor of the destination) are periodically transmitted, as well as a list of responses indicating which nodes should acknowledge ($ACK$) the update. An idle Hello message is required to be sent to by the node within a specific time period to ensure connectivity when there is no change occurs in routing table since last update. Otherwise, the lack of messages from the node indicates the failure of that link; this may cause a false alarm. When a node $A$ receives a Hello message from a new node $B$, that new node is added to the $A$’s routing table, and the $A$ node sends
node B a copy of its routing table information. On receiving an ACK, the node updates its MRL. In the event of the loss of a link between two nodes, the nodes send update messages to their neighbours. The neighbours then modify their distance table entries and check for new possible paths through other nodes. The node also updates its routing table if the new path is better than the existing path. Any new path found is relayed back to the original nodes so that they can update their tables accordingly. A unique feature of this algorithm is that it checks the consistency of predecessor information reported by all its neighbours every time it detects a change in link of any of its neighbours [157]. This algorithm avoids routing loops by checking the status of direct link of all the immediate neighbours each time any update is done. Eliminating count-to-infinity problem and avoiding routing loops provide faster route convergence when link failure event occurs. However, loop freedom achievement makes the WRP protocol suffer from high overhead control traffic caused by the periodic and triggered exchange of routing tables and the reliance on ACK and Hello responses (caused by spurious retransmission of route tables if ACKs or Hellos are lost) [157]. Another drawback of this protocol, periodic Hello message consumes power and bandwidth. Also maintaining four tables requires a large amount of memory.

### 3.2.4.4 Zone Routing Protocol (ZRP)

This protocol was the first hybrid routing protocol with both a proactive and a reactive routing component. Each node proactively maintains routes within a local region (referred to as the routing zone) [80]. Nodes in the network need to know only the topology of their routing zone; i.e. the routing messages are only propagated locally. The ZRP divides the networks into several routing zones in which routing between members within a zone is performed via proactive methods called Intrazone Routing Protocol (IARP), and routing
between different routing zones is performed via reactive methods called Interzone Routing protocol \((\text{IERP})\) \cite{158}. The \(\text{IARP}\) performs routing among members of a zone. It learns the minimum distance and routes to all the nodes within the zone. The distance is referred to the zone radius where each node is required to know the topology of the network within its routing zone only and nodes are updated about topological changes only within their routing zone. The routing protocol for \(\text{IARP}\) zone is not defined and can include any number of proactive protocols, such as Distance Vector or link-state routing \cite{159}.

The \(\text{IERP}\) protocol is employed for discovering routes between different routing zones where a destination node is located in a different zone from that of the source node. The route discovery process operates by broadcasting a \(\text{RREQ}\) message to all border nodes within their routing zone. This process is repeated until the required node is discovered. Following this discovery, a \(\text{RREP}\) message is sent back to the source demonstrating the route.

The \(\text{ZPR}\) inherits advantages of the proactive and reactive protocols. Routes to nodes located outside the zone can be found by efficiently querying method. Also, routes within the routing zone can be found quickly. The route discovery requires a relatively small number of query messages as these messages are routed only to "peripheral" nodes \cite{160}. Unlike other proactive protocols, the \(\text{ZPR}\) limits broadcasting of information about topology changes to the neighbourhood of the change only. One of the drawbacks of this protocol is that the \(\text{IARP}\) is not specified which that means using different \(\text{IARP}\)s force the nodes to support several different routing protocols \cite{159}.
3.2.4.5 Temporally Ordered Routing Algorithm (TORA)

The TORA protocol is a highly adaptive loop-free distributed routing algorithm based on the link reversal algorithm [161]. It is well suited for high density dynamic mobile networks. This protocol is designed to discover routes on demand and it provides multiple routes for any desired source-destination pair. TORA protocol minimizes communication overhead by localizing algorithmic reaction to topological changes when possible. The TORA protocol is based on the concept of the localization of control messages to a small set of nodes near the occurrence of a topological change [162]. This can be done by maintaining routing information about adjacent (one-hop) nodes. This protocol maintains a destination-oriented directed acyclic graph (DAG) for each possible destination. Any node in this graph leads to the destination by following in logical direction which links have [163]. Each router simply tries to maintain information regarding the “direction” (or set of next-hop neighbours) for forwarding packets to a given destination. Thus, a node with a “route” to a given destination has one or more of its next-hop neighbours marked.

TORA protocol uses the notation of height to find the direction of each link. The height of the source node is defined as the largest value and the height of the destination node is the smallest value [164]. All nodes in the network make use of height when any node in the network attempts to communicate with another node. The logical links are considered to be directed from nodes with higher height towards nodes with lower height [161]. TORA functionality based on three basic phases, that is, route creation, route maintenance, and route erasure. During the route creation and maintenance phases, nodes use a height metric to establish a DAG rooted at destination (i.e. the destination is the only node with no downstream links) [165]. Then links will be assigned based on the relative height metric of
neighbouring nodes. Route maintenance will be used to reestablish a \textit{DAG} due to topology changing (during the times of mobility). \textit{TORA} also employs three control packets are used by each function, that is, Query (\textit{QRY}), Update (\textit{UPD}), and Clear (\textit{CLR}). The height is defined as a function of five parameters as follows:

\[
H_i = (\tau, o_{id}, r, \delta, i). \\
\text{(3.16)}
\]

Where \( \tau \) is a new reference level which represents the time of the link failure. It is defined each time a node loses its last outgoing link. The \( o_{id} \) parameter represents a unique identifier of the node that defined the new reference level. While \( r \) is the reflection indicator bit, \( \delta \) is the propagation ordering parameter, and \( i \) is the unique node identifier (\textit{ID}). The first three elements collectively represent the reference level. A new reference level is defined each time a node loses its last downstream link due to a link failure. \textit{TORA}’s route erasure phase essentially involves flooding a broadcast \textit{CLR} packet throughout the network to erase invalid routes [166].

Each node in the network runs a copy of \textit{TORA} for each destination. When a node attempts to find a route to a destination, it first broadcast a \textit{QRY} packet which carries the address of the destination for which it requires a route [167]. This packet will be propagated through the network until it reaches either the destination or an intermediate node having a route to the destination. The node that receives the \textit{QRY} packet will broadcast an \textit{UPD} packet listing its height with respect to the destination. The node that receives the \textit{UPD} packet sets its height to a value greater than the height of the neighbour from which the \textit{UPD} was received. This has an effect of creating a series of directed link from the source (\textit{QRY} packet originator) to the node that initially generated the \textit{UPD}. Nodes adjust their height to a local maximum with respect to its neighbour and transmit an \textit{UPD} packet when a routing
failure occurs. A node will attempt to discover a new route when it has no neighbour of finite height with respect to this destination. In the detection of a network partition, the node generates a CLR packet that rests the routing state and remove invalid routes from the network [168]. TORA builds a multipath routing structure and uses the availability of alternate paths to limit the reactions to topological changes. Thus, it is logical that the failure reactions for TORA may be less frequent and have a smaller scope than for a distance vector algorithm on average [169].

3.2.4.6 Optimized Link State Routing Protocol (OLSR)

The OLSR protocol is an IP routing protocol developed for Ad Hoc networks. It operates as a table driven and proactive protocol, which allows periodic exchange of information of network topology among all the nodes of the network [142]. This protocol is a proactive link-state routing protocol, which employs Hello and Topology Control (TC) messages to discover and then propagate link state information throughout the network. OLSR utilizes the multipoint relay (MPR) mechanism which represents the key concept of this protocol. In this mechanism, the nodes that periodically forward messages during the flooding process will be selected. The topology information collected by these nodes will be utilized to compute next hop destinations for all nodes in the network using shortest hop forwarding paths [170]. Employing such technique considerably reduces the message overhead in comparison to pure flooding method where every node has to transmit each received message when it receives the first copy of it. A node selects MPRs from among its one-hop neighbours with symmetric, i.e., bidirectional, links. The idea of selecting the route through MPRs automatically is to avoid the problems associated with data packet transfer over unidirectional links, such as not getting link layer acknowledgments for data packets at
each hop for link layers employing this technique for unicast traffic [171]. In the route calculation, the MPRs are used to form the route from a given node to any destination in the network. The protocol uses the MPRs to facilitate efficient flooding of control messages in the network [170].

Basically, the OLSR protocol is based on the following mechanisms: neighbour sensing based on periodic exchange of Hello messages, efficient flooding of control traffic uses the concept of MPRs, and computation of an optimal route using the shortest-path algorithm. The neighbouring sensing mechanism is used to detect the change in the neighbourhood of the node. For example, node A is called a neighbour of node B if these two nodes are directly linked. Node C is called a two-hop neighbour of A, if node C is a neighbour of node B and not a neighbour of node A, and there exists a symmetric link between A and B and an asymmetric link between B and C. In this mechanism, the node periodically transmits Hello messages. This message contains the address of the transmitter node, the list of its neighbour, including the link status (e.g. asymmetric or symmetric). A node thereby informs its neighbours of which neighbours it has confirmed communication. When a Hello message is received, a node produces description information about the links in its neighbourhood and about its two-hop neighbourhood. Each node maintains this information set which is valid for a limited time only and has to be updated to keep it valid. Referring to the MPRs mechanism, finding a mechanism which allows delivering topological information to each node without unnecessary duplication retransmissions (i.e. transmitting the same OLSR control message twice) is required. Hello messages are used for this purpose in order to provide topology information for the nodes [170].
The MPR concept is used to decrease the flooding overhead compared to the full flooding. In this concept, each node selects independently a set of nodes as MPRs. The node uses the chosen set to reach all its two-hop neighbours through its MPR relays. Each node in the network maintains a list of nodes which selected it as MPR. A broadcasted packet is retransmitted by a MPR node when it is received from a node for which it is located in the MPR set, further receptions of the same packet are dropped. The mechanism of the computation of an optimal route can be summarized as follows: In order to find an optimal route, all nodes with a non-empty set periodically sent a TC message. Each TC message consists of the address of its originator and the MPR set of that node. All MPRs of a node get the reachability information of that node. As a result, a partial topology graph will be received by all nodes through using that information and the links of their set of links to their MPR selectors. For computing the optimal path, the shortest path algorithm is applied to the partial topology graph. Topology information is only valid for a limited period of time in each node and will be removed from the graph when it is expired [172].

Using techniques such as MPR is one of the advantages of this protocol as it makes the protocol particularly suitable for large and dense networks. The larger and more dense a network, the more optimization can be achieved as compared to the classic link state algorithm [170, 173]. Another advantage of this approach is that connections are made quickly. Periodically discovering the network is one of the disadvantages of this approach. Because programs implementing OLSR are typically large and complex, continuous calculation and memory burdens may be too heavy for small computers.
3.2.4.7 Ad Hoc On-Demand Distance Vector Routing (AODV)

The AODV is a reactive routing protocol, it enables dynamic, self-starting, multi-hop routing between mobile nodes wishing to establish and maintain an Ad Hoc network [174]. The AODV builds upon the DSDV algorithm as it uses Bellman Ford algorithm to calculate the path. It is an improvement on DSDV algorithm by minimizing the number of required broadcasts [175]. This is realized by creating routes on a demand basis instead of maintaining a complete list of routes as in the DSDV algorithm.

The routing process of this protocol operates as follows, when a node intending to send a packet to a destination, if the sender node has no valid route to that destination, it will initiate a path discovery process to locate the destination node. It first broadcasts a route request packet (RREQ) to its neighbour. The neighbour node will forward the RREQ packet to their neighbours, and so on, until either the destination is reached or it has found an intermediate node that has a route to the destination [174].

A concept of destination sequence number was used by the AODV protocol to ensure all routes are loop-free and contain the most recent route information. Every node in the network maintains its own sequence number in addition to a broadcast ID. The broadcast ID is incremented for every RREQ the node initiates. This ID is used together with the IP address of the node to uniquely identify the broadcasted RREQ message. The source node includes in the RREQ message the most recent sequence number it has for the destination in addition to its own sequence number and the broadcast ID. The intermediate nodes can reply to the RREQ (sends a message backwards through a temporary route to the requesting node), only if they have a route to the destination whose corresponding destination
sequence number is greater than or equal to that contained in the \textit{RREQ}, otherwise they forward the \textit{RREQ} message [176]. During the process of forwarding the \textit{RREQ} message, intermediate nodes insert in their routing tables the address of the neighbour from which the first copy of the broadcast packet is received, thereby establishing a reverse path. In the case of receiving additional copies of the same \textit{RREQ} message, these additional copies will be discarded. In receiving the \textit{RREQ} message by the destination or an intermediate node with a fresh enough route, a route reply message (\textit{RREP}) will be sent back by the destination/intermediate node to the neighbour from which it fist received the \textit{RREQ} [176]. The \textit{RREP} is unicast in a hop-by hop fashion to the source. In the process of sending back a \textit{RREP} message to the source node, nodes along this path set up forward route entries in their route tables which point to the node from which the \textit{RREP} originated. With each route entry, a route timer is associated to delete an entry that is not used within the specific lifetime. When the source receives the \textit{RREP}, it records the route to the destination and can begin sending data. If multiple \textit{RREPs} are received by the source, the route with the shortest hop count is chosen [176].

Route maintenance operates as follows: the source node can re-initiate the route discovery process to find a new route to the destination when it moves. Moreover, if a node along the route moves, its upstream neighbour realizes the movement and propagates a link failure notification message (\textit{RERR}) to each of its active upstream neighbours to inform them of the removal of that part of the route [176]. The \textit{RERR} message is propagated by the nodes to their upstream neighbours until the source node is reached then the source node may regenerate a new route discovery process for that destination if it is still desired.
The concept of *Hello* message is one of the *AODV* aspects to maintain the local connectivity of a node. It is periodically broadcasted by a node to inform each node of other nodes in its neighbourhood. The *Hello* message technique presents greater knowledge of the network connectivity as it lists the other nodes from which a node has heard [146].

The main advantage of the *AODV* is that routes are established on demand and the distance vector routing algorithm is used to find the latest route to destination which that required no much memory or calculations [177]. This protocol produces no extra traffic for communication along existing links. One of the *AODV* disadvantages is that it requires more time to establish a connection. However, the periodic beaconing (*Hello* message) initiated by the protocol leads to unnecessary bandwidth consumption. Another disadvantage is that intermediate nodes can lead to inconsistent routes if the source sequence number is old and the intermediate nodes have a higher but not the latest destination sequence number, thereby having out of date entries [178]. In addition to that, generating multiple *RREP* s in response to a single *RREQ* packet can result in heavy control overhead, especially with dense networks.

### 3.2.4.8 Dynamic Source Routing (*DSR*)

The *DSR* protocol operates on-demand and it is composed of two mechanisms that work together to allow for the discovery and maintenance of source routes in the Ad Hoc network. The *DSR* employs an efficient route discovery mechanism. Route discovery is used to determine the route from source to destination. Routed packets contain the address of each node it traverses in order to get to its destination. When a node in the network using the *DSR* routing protocol attempts to send a packet to a destination node, it first queries its
Route Cache Table where the previously learned routes are preserved. If there is no route found in its cache, the sender node initiates route discovery procedure to find a new route to the destination node.

The route discovery procedure operates as follows: the sender node broadcasts a Route Request (RREQ) packet. Each node receiving a request message rebroadcasts it unless it is the destination or it has a route to the destination in its route cache [179]. If the intermediate node has no route to the destination node, it rebroadcasts the RREQ message after adding its address to the source route. If the intermediate node finds a route, it will not propagate the RREQ packet, but instead it sends a RREP to the source node by concatenating the recorded source route contained in the RREQ packet to the cached route to the destination node present in its route cache [81]. The intermediate node will discard the RREQ message if it has seen the RREQ before (i.e. message with the same request identification (ID)).

Each Route Request packet carries the identifications of the source and the destination nodes, unique request identification and a list of the addresses of the intermediate nodes, by which that Route Request packet has been forwarded [81], see Figure 3.6. When the destination node receives this Route Request message, it returns a Route Reply (RREP) message to the source node containing the path taken by the route request message as it is shown in Figure 3.6. When the source node receives this route reply message, it caches the path in its route cache in order not to repeat the route discovery process for each new packet destined to the same target node, for more details see [55]. Once this packet reaches the source node, then the source node will start sending data packets to the destination node. Intermediate nodes, will then perform passive learning by storing some information
from the route list (inside route reply packet header) into their route caches for future routing purposes [180].

![Diagram](image.png)

Figure 3.6: An example of the DSR route discovery mechanism.

During Route Discovery process, the source node stores a copy of the message in a local buffer called the Send Buffer. Send Buffer has a copy of every packet that cannot be transmitted by this node due to lack of a route. Each packet is time stamped and discarded after a specified time out period, if it cannot be forwarded [55]. For packets waiting in the Send Buffer, the node should occasionally initiate a new route discovery for the packet’s destination address. A new route discovery rate for the same destination node should be limited if the node is currently unreachable. This results in the waste of wireless bandwidth due to a large number of RREQs destined for the same destination which in turns results in high overhead. To reduce the overhead, the node goes into exponential back-off for the new route discovery of the same target. Packets are buffered that are received during the back-off. If the node attempts to send additional data packets to this same node more frequently
than this back-off limit, the subsequent packets should be buffered in the Send Buffer until a RREP is received. A new route discovery should not be initiated until the minimum allowable interval between new route discoveries for this target has been reached [81].

The DSR protocol supports a search ring approach where it limits the number of route discoveries to two attempts. In the initial attempt, The DSR uses a mechanism to send a nonpropagating RREQ with a hop limit of 1 (i.e. TTL = 1) to look for either the destination or some node with a route to the destination within its immediate neighbourhood. If no RREP is received (i.e. a route can not be found) within a timeout period, a new RREQ is sent by the sender with no hop limit which essentially floods the network. This dual-phase search has been extended to an expanding ring search by allowing the hop limit to increase in incremental steps. This process increases the average latency of the route discovery [55].

Due to the nature of broadcast transmission, many nodes around the broadcasting node receive the RREQ. Neighbouring nodes may attempt to send a RREP simultaneously result in what is called a RREP “storm” which causes local congestion and increases the rate of packet collisions in the network thereby wasting bandwidth. Having some nodes delay sending their RREPs may mitigate this problem. The delay time \(d\) is specified to be:

\[
d = H \times (h - 1 + r)
\]

(3.17)

Where \(h\) is the number of hopes of the returned route, \(r\) is a random number between 0 and 1, and \(H\) is a small constant delay to be introduced per hop.

Route Maintenance mechanism is used when an intermediate node is incapable of delivering the received packet to the next hop due to link/route failure. This node will first
salvage the packet by examining its route cache for another route to the same destination. If the route exists, the node replaces the broken source route on the packet’s header with the route from its cache and retransmits the packet. If this intermediate node has no route to the same destination, it will return a Route Error (RERR) to the source node to prevent it from sending more packets on the broken route. Any node hearing the RERR updates its route cache to remove a failed link. When the source node receives a Route Error packet, it will attempt to find alternative routes from its route cache. If alternative routes are not available, the source node will invoke Route Discovery again to find a new route for subsequent packets that it sends. Unfortunately, DSR produces a long delay when a route is rebuilt. 

Finding a route in a wireless network require considerable resources, such as time and bandwidth because it relies on broadcasting [180]. Routes may be shortened if one of intermediate nodes becomes unnecessary. For example, in Figure 3.7, if C overhears that A is forwarding a packet to B that is destined to C, then C sends a “Gratuitous” message (its RREP message) to original sender A. The RREP informs A to route packets as A-C-D instead of A-B-C-D, see Figure 3.7. In certain situations, caching of negative information can help DSR. For example, in Figure 3.8, if A knows that link C-D is broken, it can keep this information in its routing cache for a specified time (using a timer), e.g. by making the distance to routes through C as infinity. A will not use this path in response to any RREP it receives for subsequent RREQs. After the expiration of timer, the link can be added again in the route cache with correct hop counts, if link is repaired.
Consider the case shown in Figure 3.8, where link quality is varying with respect to time i.e. it is in a fade for some time. If the C-D link is in a fade, i.e. it is healthy for an interval and broken for another interval. By keeping the information that the link is broken, the node can prevent the addition of this link in its route cache when it becomes healthy again. It can keep this information in its routing cache for a specified times (using a timer) till the link become normal. After the expiration of timer, the link can be added again in the route cache with correct hop counts. This mechanism prevents oscillations in the route cache.
In DSR nodes cache learnt routes (through packets carrying either a RREP or a source route) in an attempt to reduce the amount of routing related traffic in the network. Likewise, nodes delete information from their cache as they learn previously existing links in the network have broken (through a route error or through the link-layer retransmission mechanism reporting a failure in forwarding a packet to its next-hop destination). The route discovery process is initiated only if the desired route cannot be found in the route cache. During the route discovery process, if the desired route is found in the route cache of an intermediate node, this node returns a RREP to the initiator itself rather than forwarding the RREQ. In the RREP, it sets the route record to list the sequence of hops over which this copy of the RREQ was forwarded to it, concatenated with its own idea of the route from itself to the target from its Route Cache [55]. The Route Cache process supports storing more than one source route for each destination. If a source node is employing a source route to some destination that includes intermediate node, the source node should shorten the route to the destination when it learns of a shorter route to intermediate node than the one that is listed as the prefix of its current route to the destination [55]. However, the cache process should still have the ability to switch to the older, longer route to the destination node if the shorter one is not valid.

The DSR is simple and is particularly suited to wireless networks. The main advantages of DSR over other popular protocols such as AODV and DSDV are:

- The DSR protocol can successfully discover and forward packets over paths that contain unidirectional links in addition to the bidirectional ones [55].
• It functions completely on demand, and it does not generate control overheads as it requires no periodic activity of any kind at any level within the network [55]. It reacts to changes in the environment only when necessary which allows the routing packet overhead of the protocol to automatically scale directly with the need for reaction to medium changes. This scalability dramatically lowers the overhead of the protocol by eliminating the need for any periodic activities, such as the route advertisement and neighbour detection packets that are present in other protocols [181].

• The DSR protocol can make use of multiple routes to any destination by employing a route cache table. The benefit of utilizing route cache is to reduce the need for the route discovery operation. It also allows each sender to select and control the routes used in routing its packets, for example, for use in load balancing or for increased robustness. Maltz has shown that DSR delivers excellent routing performance across a wide range of wireless network environments as he dissected the DSR into its component mechanisms to show how they combine to give DSR that performance [81].

• Unlike other protocols, DSR also capable of storing all usable routing information extracted from overhearing packets.

• The DSR protocol does not need to have a view of the entire network topology as the complete route is carried in the packet header [182]. It also eliminates the route inconsistency that the popular AODV or DSDV protocols might encounter. Inconsistency routing can occur in AODV protocol when the source node initiates
route discovery process with a destination sequence number older than the intermediate node has but it is not latest destination sequence number.

The *DSR* protocol has some other advantages such as easily guaranteed loop-free routing, rapid recovery when routes in the network change, allowing the network to be completely self-organizing and self-configuring, without the need for any existing network infrastructure or administration. The other advantage of this protocol is that the *DSR* utilizes source routes to control the forwarded packets through the wireless network. The key advantage of a source routing design is that intermediate nodes do not need to maintain consistent global routing information, since the packets themselves already contain all the routing decisions [181]. Every packet that carries a source route contains a description of a path through the network. Therefore, with a cost of no additional packets, every node overhearing a source route learns a way to reach all nodes listed on the route.

The main reason of employing the *DSR* protocol in this work is to take advantage of the *DSR* features described above as WMN stations are relatively stationary to minimize the routing overhead. In particular the unique features of this protocol are the on demand operation and cache route mechanism. The strategy of caching the discovered routes will reduce the need for the discovery mechanism and hence it reduces the routing overhead and the consumption of the network resources. The new *AEF* metric can be adopted by other routing protocols as it is based on the local information at a node. There is no practical reason why this metric could not be used with other routing protocols.
3.2.4.9 Modification to the DSR

The DSR protocol using the hop count metric fails to take into account certain link quality parameters such as interference, availability of bandwidth, link load, packet loss ratio, etc. which has an important impact on the performance of WMNs. To consider link quality information in the routing procedure, a new metric is required to be implemented as an alternative to the hop count metric. In this work, the AEF metric and its modified version ModAEF (described in section 4.3.3), have been introduced as new routing metrics. Those metrics can be used to find paths between the nodes in the network. The aim of the introduced metrics is to find paths with the least congestion in order to improve the global throughput of the network.

For this purpose, the route discovery mechanism of the standard DSR routing protocol has been modified by replacing the $H_c$ metric with the new AEF based metrics. In the case of employing AEF metric, the route selection procedure operates on the basis of finding the path with the highest minimum AEF value among the available paths between the source and destination nodes. This strategy is used to determine the bottleneck in terms of the level of congestion for each available route between the source-destination pair. The route with the highest minimum AEF value of the bottleneck link will be chosen by this strategy. The bottleneck link essentially determines the end-to-end throughput and delay time. The same route selection strategy has been applied when the ModAEF was employed as alternative routing metric to the $H_c$. The key feature of this modified route selection procedure is that it attempts to discover paths that have lower levels of congestion which can support high throughputs. The objective of this approach is to make use of MAC layer information at the
routing layer to enhance the global performance of the network. The use of the cross-layer techniques have been shown that a significant throughput improvement is achieved [183].

3.3 Summary

An introduction to WMNs is presented in this chapter giving some details about their characteristics and architecture. Due to the importance of the routing issue for WMNs, several routing metrics have been presented with critique of each. The most popular proactive and reactive routing protocols designed for wireless networks have been also presented in this chapter. Particular attention has been paid to the Dynamic Source Routing Protocol (DSR) as it is the subject matter of this thesis. The completely on demand operation of the DSR protocol and the route cache mechanism in addition to other features of the protocol, such as the salvage mechanism and supporting the use of multiple paths to any destination in addition to the support of using unidirectional links, resulted in the selection this routing protocol for this research work.
CHAPTER 4

SIMULATION MODEL

Overview
This chapter introduces the simulation model, the simulator settings, and assumptions that were implemented using the discrete event simulator OPNET Modeler 11 [184]. The OPNET simulation models have been developed to test and evaluate the performance of the AEF metric within the DSR routing protocol in a WMN environment. Due to the shortcomings exhibited in implementing the AEF as a cost metric for the DSR routing protocol, a further two modifications to the DSR routing protocol are introduced in this chapter. The examined scenarios in this work have been classified into several groups based on their configuration and these are presented in this chapter.

4.1 Justification for Adopting Simulation Approach
In this work, a distributed mechanism is involved which requires large scale testing. To achieve that, it was required to examine the AEF based route selection rule with large scale networks of 99 nodes randomly distributed across the network. It was also required to test the new metric with different number of gateways located in various positions in the network. It was also required to tune and change some parameters such as transmission range, transmission rate, packet size, and packet rate in order to examine the effectiveness
of the new $AEF$ metric. Further investigation of this metric was required in order to test its
effectiveness with different traffic types (e.g. Poisson and Pareto) and flow directions (i.e.
uplink flows, downlink flows, and bidirectional flows). This new metric was also examined
with the TCP/UDP transport protocols. In this work, it was required to generate 1000
random topologies for each scenario to properly investigate and analyze the performance of
the $AEF$ metric.

Attempting to meet all these requirements for the testing of the $AEF$ metric through an
experimental approach would be impractical given the scale and complexity of the test
network required. Also, it would be extremely time consuming to perform all these tests
experimentally. Therefore, it was decided to adopt an approach based upon computer
simulation as this represents a far more feasible and practical alternative to experimental
analysis. Furthermore, computer simulation allows for complete control of the simulation
environment, i.e. it eliminates any unpredictable results than can arise from random
variations in the signal propagation.

4.2 Network Modeling

The OPNET Modeler 11 is a popular software application for performing simulations on a
wide range of networks and is used both by the commercial and research communities. The
OPNET Modeler has been employed in this work to simulate the network following a
survey which showed it to be an efficient, well documented modelling package. Moreover,
it is relatively straightforward to develop and implement new modules, and is easy to
configure and simulate large scale test scenarios. It provides a comprehensive framework
for modeling wireless as well as wired networks.
It has been demonstrated by a number of researchers that OPNET gives accurate results compared to other network simulators [185, 186, 187]. OPNET modeler is an advanced package that allows the user to design and study communication networks, devices, protocols, and application [188, 189, 190]. It has been used to simulate different types of computer networks operating in different environments [191]. Lucio et al examined the accuracy of the OPNET modeler against the popular NS2 simulator using a network testbed [186]. In this work, several scenarios were evaluated. These scenarios were generated in the simulation tools and the network testbed. A constant bit rate (CBR) and a file transfer (FTP) session were used. It has been shown in these tests that NS-2 provides similar results compared to OPNET Modeler, but the “freeware” version of NS-2 makes it more attractive to a researcher. However, the complete set of OPNET Modeler modules provides more features than NS-2, and it therefore will be more attractive to network researchers. Chang [187] stated that the OPNET modeler is one of the most powerful software simulation packages following a comparison that he made against several other computer network simulators. He is also stated that the OPNET provides a comprehensive development environment for the specification, simulation and performance analysis of communication networks.

OPNET offers several modeling editors such as project editor, node editor, process editor, etc. Each editor enables the user to change such characteristics as the network size, node model, etc. It has a rich set of features allowing the user to model most available network technologies. The project editor in the Modeler was used to create simulation scenarios for the standard and modified DSR protocol.
Different WMN scenarios have been simulated using the OPNET modeler to evaluate and analyze the performance of the modified DSR protocol. The performance of all scenarios has been examined with the network node density varied from low density to high density and also with varying the traffic load on the network. All scenarios operate under IEEE 802.11b operation using direct-sequence spread spectrum (DSSS) modulation. One of the assumptions in this model is that the line rate adaptation is switched off, i.e. all nodes transmit at 11 Mbps. This assumption was made to remove the dependency of the throughput on the line rate in the analysis. In practice, nodes can transmit at different a rate which causes a reduction in the throughput and an increase in the delay time [192].

For the sake of simplicity Poisson traffic sources have been used in this model. However, it is recognized that network traffic is often far from Poisson and can sometimes exhibit self-similarity and long-range dependence [193, 194, 195, 196]. Poisson traffic is widely used for convenience as it easy to generate and to analyze [197]. The Poisson process represents an example of a traditional traffic model that exhibits only short-range dependence. Poisson does not result in high congestion or large increase in packet drop rates compared to the heavy-tailed traffic. Consequently the results obtained with using Poisson traffic will result in an overestimation of the performance improvement of the system.

4.2.1 Network Modeling Using OPNET

OPNET simulation models are organized in a hierarchy consisting of three main levels, see Figure 4.1, namely, the simulation network, node models, and process models. These three modeling environments are sometimes referred to as the modeling domains of OPNET. The simulation scenario or simulation network represents the top level of the hierarchy. It
describes the network layout, the nodes, and the configuration of attributes of the nodes comprising the scenario. The second level in the hierarchy refers to the node models. They consist of an organized set of modules describing the various functions of the node. The process models are the lowest level in the hierarchy.

Process models comprise finite state machines, definitions of model functions, and a process interface that defines the parameters for interfacing with other process models and configuring attributes. The process models rely on external files which contain a set of supporting functions or data structures. Finite state machine models are implemented using Proto C, which is a discrete event library based on C functions. The hierarchical structure of the models, coupled with support for C language programming, allows for easy development of communications or networking models.

Figure 4.1: Hierarchical architecture of the OPNET simulation modeler.
All the scenarios used in this work are designed in OPNET Modeler to avail of the rich set of features and libraries that it offers. The OPNET Modeler provides several options such as choosing the type and the size of the area, node placements in the area, etc. We have created test scenarios using a fixed number of randomly distributed nodes with varying network sizes in order that a range of sparse to dense networks can be simulated.

The Perl programming language has been used to generate the random topologies. Each topology comprises a single gateway and 99 nodes distributed randomly across the network coverage area. All generated topologies were imported into OPNET to run the simulated topologies. The generated results were exported to Matlab for analysis. The Matlab tool is a numerical computing environment. It was developed by the MathWorks, MATLAB provides matrix manipulation, plotting of functions and data, implementation of algorithms, etc.

### 4.2.2 DSR Model for WLAN Node

The DSR protocol has been chosen for this work as it is a simple and efficient on-demand routing protocol utilized in multi hop networks. The route discovery and the route cache of DSR protocol have been modified in this work. The objective of using a route cache is to avoid frequent route discovery where the node maintains a set of paths to each destination. The node chooses the path with the highest minimum $AEF$ to the destination, see Equation (4.9).

The Mobile Ad hoc Networks (MANET) routing protocols are simulated in the OPNET modeler. The MANET protocols are made a child process of the IP module (main IP
module spawn child process) as there is no individual module for the MANET, see Figure 4.2. In the node model of a wireless station of the OPNET modeler, the manet_mgr process of the MANET operates as a dispatcher process to spawn the appropriate routing protocol. It is located in the IP module (containing all the network layer functionality and handles all packet routing based on IP address among other things) as a child process of the ip_dispatch process. The dsr_rte process which represents the DSR routing process is created as a child process of manet_mgr process. When the DSR is configured on a node, the dsr_rte process is spawned by the manet_mgr process to run the DSR protocol on the node [184]. In order to apply the newly introduced routing metrics in the DSR protocol, the node model of the network scenarios has been updated by modifying the manet_mgr process, dsr_rte process, and wireless_lan_mac model.
In this modification, the packet headers of route request and route reply are also modified by including the newly introduced $AEF$ routing metrics. The estimated time for a node to transmit its load and its estimated access time to the medium are collected at the wireless_lan_mac module to be passed to the manet_mngr process which is located in the ip_dispatch process then to the dsr_rte process, see Figure 4.2. Within this process the $AEF$ of each node in the available routes between the source and the destination nodes is calculated in order to find the route with the highest minimum $AEF$ value according to the Equation (2.16).
In the above figure, the modeling under OPNET is divided into three levels. The upper level is the application layer, the second level is the network layer and lower level is the MAC/PHY layer. The network layer is the core of the node model, since it contains the DSR routing process model. Figure 4.3 shows the packet flow in the second level.

Packets that arrive at the network layer which may come from a higher layer application or from a low layer via the MAC and PHY layer of the radio (indicated by arrows 1 and 10 in figure 4.3). These packets will be processed by the “IP routing process” to be forwarded to its destination. If the packet has no route to forward to (arrow 6), it will be sent to
manet_mgr and dsr_rte processes to find a suitable route for it based on the specified routing procedure. It will then be sent to the ip_dispatch in order to be forwarded to its destination.

4.2.3 Implementing the Modification

In this study the path selection rule of the DSR protocol has been modified to include the new AEF metric as an alternative to the hop count metric. If the ip_dispatch process discovers that an interface is configured to run the MANET protocol then it invokes the manet_mgr process model, which is responsible for identifying and then invoking a specific MANET routing protocol such as DSR protocol. When the DSR protocol is invoked the dsr_rte process is spawned by the manet_mgr process. The OPNET implements the DSR protocol via dsr_rte and other external files such as dsr_pkt_support, dsr_route_cache, dsr_maintenance_buffer, dsr_route_discovery, dsr_notif_log_support, dsr_send_buffer, dsr_support, and manet_support.

In the modified version of the DSR protocol, the minAEF option field within the header of RREQ and RREP packets is used to carry the information of the new metric. The option is processed on a hop by hop basis. The minAEF option has three fields, option type, option length, and metric. Option type and option length fields store the same information as specified in the standard DSR. The metric field stores the metric data value. For this purpose, the dsr_rte process model has been modified in addition to the wlan_mac process model, see Figure 4.2. The minAEF option is added to the packet headers of route request, route reply, and source route. The headers of the route request message, route reply
message, and source route are modeled via an external header file `dsr_pkt_support` (in Figure 4.3). The `dsr_pkt_support` is responsible for creating the route request and route reply options in addition to the source route TLV (Type, Length, and Value). In the modified DSR routing version the `dsr_pkt_support` external file is modified to add the `minAEF` option to the header of the route request, route reply messages, and to the source route. The `dsr_rte` process is responsible for initializing the state variables and processes the arrived packet based on its TLV options set in the DSR packet. It is also responsible for initiating a route request message, sending out a reply message on recipient of route request packet to the source of the route request.

In the modified version of the path selection rule of the DSR protocol, when the source node wants to send out a packet but it does not have a route in its cache to the destination, the source node attaches the `minAEF` option field into the route discovery packet header before broadcasting the packet to the neighbouring nodes after setting it to 1. The typical size of the route discovery packet is 60 bytes. Initialising the route request packet and setting the `minAEF` to value of 1 is performed via the `dsr_rte` process model. Upon receiving a RREQ packet with a `minAEF` field attached, if the intermediate node has no route in its cache to the destination of this RREQ message, it will add its address and updates the `minAEF` field in the packet header as the packet gets forwarded to the destination node. Updating `minAEF` field occurs in the `dsr_pkt_support` file by measuring the AEF locally at a node.

Measuring the AEF is performed by retrieving parameters from the MAC layer via `wlan_mac` process which are the access time and the measured time of the transmitted
load. Over the interval of 1 second, the **wlan_mac** process measures the average time required to transmit a frame and the average time required to access the medium in order to transmit a frame. Under an assumption of statistical stationarity, it assumed that the *AEF* for the previous interval will also apply to the current interval. These measured parameters will be retrieved by the intermediate node traversed by the RREQ message in order to compute its *AEF* value, according to Equation 2.16, and compare it to the stored value in the *minAEF* field.

From a practical perspective the *AEF* metric can not be directly measured due to the difficulty of retrieving access MAC layer information (e.g. Backoff counter values etc) from the WLAN adapter. However, the measurement of the *AEF* metric can be performed indirectly, by analysing the time between transmitted frames on the medium. This allows the average backoff counter values and the average number of deferrals to be estimated.

When the destination node receives the route discovery packet, the route is reversed and placed in the newly created route reply packet as the original *DSR*. In addition, the measured *minAEF* along the path from the source node is also inserted into the route reply packet. As mentioned earlier, initiating the route reply and listing the recorded route by the RREQ message and the *minAEF* into the route reply is the responsibility of the **dsr_rte** process. In other words, the **dsr_rte** process model is modified to copy the *minAEF* field to the initiated route reply by the destination node.

If the intermediate node receives a route discovery packet and is able to find a route to that destination in its cache, the route will be retrieved from the cache and concatenated to the source route in the route request packet. The **dsr_rte** process is responsible for performing
this operation. The nodes along the new route locally calculate their AEF value following the same procedure outlined above. This is performed by retrieving the required information from the wlan_mac process. The minAEF field of the route reply is set to the minimum AEF value by selecting the lowest value of the nodes along the route. Then, a route reply will be sent back to the source node.

The received routes by the source node will be cached on the basis of max(minAEF), i.e. the route with the highest minAEF value will have the highest position in the cache table. In
other words, the route is inserted in the list of routes to a destination based on priority, the highest priority is assigned to the route with the maximum minAEF by storing it at the head of the list and the lowest priority route (with the minimum minAEF value) at the tail of the list. The dsr_route_cache external file (in Figure 4.3) is modified to enable sorting the stored routes on the basis of the minAEF metric, as instead of the minimum hop count metric in the original DSR protocol.

When a node learns a route, it inserts the route into the route cache. If there is no available space in the route cache, the node may delete an existing route to enable insertion of the new route. This process is based on various criteria as detailed below. Modification is made to the dsr_pkt_support file to determine the order of the priority of routes in the route cache which is detailed as follows:

1- Multiple routes to same destination
   a. If there are multiple routes to the same destination, then delete the route with the lowest minAEF.
   b. If there are multiple routes with the same minAEF to the same destination, then the least recently used entry is discarded.

2- No route to destination
   a. There exists no route to the destination in the cache table, then determine the destinations that have multiple routes and discard the one which has the lowest minAEF and is the least recently used.
   b. If all destinations have only one route, discard the route which is least recently used among all the destinations
In this study, each cached route is assigned an expiration timer of length 10 seconds. A route will be removed from the route cache when it exceeds the expiration time. Choosing the expiry timer shorter than 10 second results in an increase in the routing overhead. Increasing the expiry timer beyond 10 seconds will have a negative impact on the currency of the route. The route may be used for long time before being updated using the latest \textit{AEF} values of the nodes along the path. This can result in the route continuing to be used even after it becomes congested which leads to possibility of packet loss. The creation of the route cache and the sorting the inserted routes is responsibility of the \texttt{dsr\_route\_cache} file. All the modifications related to route cache are implemented in this file.

When the source node sends a data packet to the destination, the entire route is included in the packet header in addition to the \textit{minAEF} field. Adding the \textit{minAEF} field to the source route enables the intermediate nodes to cache the learned routes based on the \textit{minAEF} when it forwards the source route to the destination. This involved modifying the \texttt{dsr\_pkt\_support} file in order to add the \textit{minAEF} field to the source route.

In the route maintenance mechanism an intermediate node, which is forwarding a packet, may detect that the next hop along the route is broken. In this case if the node has another route to the packet's destination in its route cache the packet will be salvaged. The node replaces the original source route on the packet with a route from its route cache after updating the \textit{minAEF} field. The salvage procedure of the standard \textit{DSR} protocol is defined in the \texttt{dsr\_pkt\_support} external file. Each node in this new path locally calculates its \textit{AEF} value following the same procedure previously mentioned and updating the \textit{minAEF} field by assigning it the minimum value of the \textit{AEF} along the new path.
<table>
<thead>
<tr>
<th>File</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>wlan_mac</strong></td>
<td>Measure the average time required for a node to transmit its load over a specific time interval and the average access time over this interval.</td>
</tr>
</tbody>
</table>
| **dsr_rte**      | - The initialised RREQ message was modified by setting the $minAEF$ value to 1.  
|                  | - Copy the $minAEF$ value from the RREQ message to RREP message.             
|                  | - Copy the $minAEF$ value to the source route carried into the data packet.   
|                  | - Updating the $minAEF$ of the new route when an intermediate node concatenates its cached route to the destination upon receiving a RREQ. 
|                  | - Updating the $minAEF$ of the salvaged route.                               |
| **dsr_pkt_support** | - Adding $minAEF$ field to the header of RREQ, RREP, and source route.    
|                  | - Updating the $minAEF$ by intermediate nodes traversed by the RREQ message. |
|                  | - Modifying the memory allocation procedure.                                
|                  | - Modifying the Copy/Destroy procedure.                                     |
| **dsr_pkt_support** | Modifying the data structure that represents the RREQ option, RREP option, and source route TVL in order to include $minAEF$. |
| **dsr_route_cache** | - Modify the create route cache procedure for storing the discovered source routes by including $minAEF$ field in order to cache routes based on the $minAEF$. |
|                  | - Modifying the sorting mechanism for the learned routes based on the $minAEF$. |
|                  | - Modifying the priority procedure when a new route is learned.             |
4.2.4 Simulation Settings

A set of homogenous settings of the DSR protocol’s parameters have been applied for all network topologies utilized in this work. The table below presents the values of the DSR variables:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max Buffer Size</td>
<td>Infinity</td>
</tr>
<tr>
<td>Send Buffer Expiry Timer</td>
<td>30 s</td>
</tr>
<tr>
<td>Max Cached Routes</td>
<td>Infinity</td>
</tr>
<tr>
<td>Route Cache Expiry Timer</td>
<td>10 s</td>
</tr>
<tr>
<td>Request Table Size</td>
<td>64 nodes</td>
</tr>
<tr>
<td>Request Table Ids</td>
<td>16 identifiers</td>
</tr>
<tr>
<td>Max Request Retransmission</td>
<td>16 retransmissions</td>
</tr>
<tr>
<td>Max Request Period</td>
<td>10 s</td>
</tr>
<tr>
<td>Initial Request Period</td>
<td>0.5 s</td>
</tr>
<tr>
<td>Non Propagate Request Timer</td>
<td>0.03 s</td>
</tr>
<tr>
<td>Request Hold off Time</td>
<td>0.03 s</td>
</tr>
<tr>
<td>Request Period</td>
<td>500 ms</td>
</tr>
<tr>
<td>Non Prop Request Timeout</td>
<td>30 ms</td>
</tr>
<tr>
<td>Maintenance Buffer Size</td>
<td>50 packets</td>
</tr>
<tr>
<td>Maintenance Hold off time</td>
<td>0.25 s</td>
</tr>
<tr>
<td>Max Maintenance Retrans.</td>
<td>2 retransmissions</td>
</tr>
<tr>
<td>Maintenance ACK Timer</td>
<td>0.5 s</td>
</tr>
</tbody>
</table>

4.3 Free Space Propagation

In this study the free space propagation model has been used and consequently any path losses due to surface reflections or multipath fading are not considered. The free space
model represents signal propagating through open space with no interactions from the environment. The free space path loss model calculates the difference in the power between the receiver and transmitter as a function of their separation. The field strength of an electromagnetic wave in free space is inversely proportional to the distance, i.e. it decreases in inverse proportion to the square of the distance to the transmitter. This results in the receiver input power fading with the square of the distance. In omni-directional antennas the received power can be described on the basis of the law of free-space propagation (also known as the Free Space Loss, FSL). In [198], an ideal point-shaped source is described as, a so-called isotropic radiator of signal energy, transmits its power $P_0$ uniformly in all directions $\Theta$. The constant spatial power density is $P_{iso} = \frac{P_0}{4\pi}$. In this isotropic case the power density flow $F$ through a sphere with radius $d$ is [198]:

$$F = \frac{P_0}{4\pi d^2} \left[ W/m^2 \right]$$  \hspace{1cm} (4.1)

In the normal case an antenna transmits the main part of the power $P_T$ (index $T$: Transmitter) in preferred directions (main and minor lobes). The antenna gain $G_T$ puts this in relation to the isotropic radiation. The product EIRP = $P_TG_T = P_0$ is called EIRP (effective isotropically radiated power). An antenna with gain $G_T$, which transmits in the mean the total power $P_0$, transmits into the direction $\theta$ the power density [198]:

$$P_{TX} = \frac{4\pi}{P_0G_{TX}}$$  \hspace{1cm} (4.2)

The corresponding power flow density (power per unit area) through a sphere with radius $d$ is:
\[ F = \frac{P_T G_T}{4\pi d^2} \]  

\( (4.3) \)

The power \( P_R \) (Index \( R \): Receiver) an antenna can take from the electromagnetic waves is the product of \( F \) and the effective antenna area which can be expressed as follows by the wavelength \( \lambda \) and the gain \( G_R \) of the receiver antenna [199]:

\[ P_R = P_T G_T G_R \left( \frac{\lambda}{4\pi d} \right)^2 \]  

\( (4.4) \)

The term \( \left( \frac{\lambda}{4\pi d} \right)^2 \) is referenced as free-space pathloss because it describes the spatial diffusion of the transmitted energy over the path of length \( d \). In a logarithmic representation the difference \( P_T - P_R \) corresponds to an expression \(-10 \log \frac{P_R}{P_T} \). In this representation the free-space loss \( L_F \) results (with \( c = \lambda f \)) in:

\[ L_F = -10 \log (G_T) - 10 \log (G_R) + 20 \log (f) + 20 \log (d) - 20 \log \left( \frac{c}{4\pi} \right) \]  

\( (4.5) \)

In the case of an isotropic antenna the last expression reduces to:

\[ L_F = -20 \log \left( \frac{\lambda}{4\pi d} \right) = -20 \log \left( \frac{P_R}{P_T} \right) \]  

\( (4.6) \)

The FSL channel model was utilised in this study due to the limitation of the current license of OPNET available as it is the only channel model available. As mentioned above, the FSL model considers only signal fading caused by distance. It ignores the affects of the environment such as reflection and multipath fading which results in overestimating of the performance of the network. Employing a channel model that takes into account the environment affects such as multipath fading will reduce the effective value of the node density factor. This will have a significant affect on the performance of the network due to its impact on the routing decision of the AEF path selection rule. In other words, multipath
fading will reduce the connectivity of the network and hence the contention will be reduced. That means the AEF value will appear higher than it actually is. This will have an impact on the routing decision of the routing protocol and hence the performance of the network will be affected. That will degrade the performance of the network in terms of global throughput.

4.4 Transmission Control Protocol (TCP) and User Datagram Protocol (UDP)

The TCP mechanism provides for a reliable, ordered delivery of a stream of bytes between a source-destination pair. TCP is the protocol that the majority of Internet applications rely on, applications such as the World Wide Web, e-mail, and file transfer. Other applications, which do not require reliable data stream service such as real time applications, may use the UDP which provides a datagram service that emphasizes reduced latency over reliability. UDP is connectionless and unreliable which means that it does not establish a virtual circuit like TCP, nor does it demand an acknowledgement. It merely sends out the message. TCP provides a point-to-point channel for applications that require reliable communications.

TCP performance is dependent on a subset of algorithms and techniques such as flow control and congestion control. Flow control determines the rate at which data is transmitted between a sender and receiver. Congestion control defines the methods for implicitly interpreting signals from the network in order for a sender to adjust its rate of transmission. Timeouts and retransmissions are used to address error control in TCP protocol. Although delay could be substantial, particularly if real-time applications are implemented, the use of both techniques offers error detection and error correction thereby guaranteeing that data will eventually be sent successfully. However in practice, most TCP
deployments have been carefully optimised in the context of wired networks. Ignoring the properties of wireless networks can lead to TCP implementation with poor performance. The TCP assumes that packet losses are always due to network congestion. But while this assumption is valid in wired networks, it is not true in wireless networks. In wireless networks, there are several causes for data packets to be lost, including losses caused by routing failures, by network partitions, and by high bit error rates. Performing congestion control in these cases (i.e. when employing TCP) does yield poor performance [200]. Moreover, the effects of interactions among TCP, MAC and routing algorithm are non-trivial to this end-to-end performance [201]. It has been shown in the work introduced by [201] that when the TCP is implemented in a wireless network, the global throughput of the network is decreased rapidly when the hop number of a route is increased. This is caused by several factors, such as MAC layer collision and inappropriate route recovery timer of the routing protocol.

In this study, the UDP traffic protocol was implemented and preferred over the TCP to avoid any possible adverse interaction between the new route selection rule based upon the AEF metric and the TCP.

4.5 Density Factor (DF)

A node density factor (DF) is used in this work as a measure of the average number of the nodes located within the transmission range of a node. It can be defined as follows:

\[ DF = \frac{\pi R^2 D - 1}{\pi R} \]  

(4.7)

Where \( R \) is the transmission range of the node in the network and \( D \) is the node density. It is assumed here that the nodes use omni directional antennas resulting in a circular
coverage area. In other words, an average node density is assumed across the network. $D$ can be defined as follows:

$$D = \frac{\text{Number of nodes}}{\text{Area}}$$  \hspace{1cm} (4.8)

Where $\text{Area}$ is the size of the area of the network and $\text{Number of nodes}$ denotes to the total number of distributed nodes across the network. The factor -1 in equation (4.7) represents the sender node itself.

4.6 Modified DSR Routing Discovery Mechanism

The rationale for modifying the DSR protocol is to make it better suited to the WMN environment based upon IEEE 802.11 WLAN technology. The WLAN medium is a shared medium where nodes must contend for accessing the medium using DCF MAC mechanism. Since the DCF is a “listen before talk” mechanism, a high level of contention for access to the medium will result in a low availability of bandwidth at a node. This in turn limits the maximum throughput that can be achieved. Unfortunately, the DSR protocol fails to explicitly consider the contention experienced locally at a node which is an important omission in WMNs based upon the IEEE 802.11 standard. In this case, the access efficiency factor $\text{AEF}$ measured locally at a node is used as an indicator of the level of contention experienced locally at that node. By incorporating the contention factor into the DSR routing mechanism, the overall performance of the network can be significantly improved as it is demonstrated in chapter 5. The performance of the modified DSR is investigated through a series of simulations performed on the OPNET modeler package. In this regard, three modifications have been made to the DSR routing discovery mechanism.
4.6.1 First Modification to the DSR Routing Discovery Mechanism

In this modification a new metric to support the DSR routing mechanism has been developed. The intention of this modification is to explicitly consider local congestion at the node and avoid routing traffic through congested regions. Specifically, the DSR protocol was modified by replacing the hop count ($H_c$) metric with an Access Efficiency Factor ($AEF$) metric. In this modification, the strategy of the algorithm is to determine the path based on the following selection rule:

$$\max_i \{\min_k \{\eta_f(l_{ki})\}\}$$  \hspace{1cm} (4.9)

Where $l_{ki}$ is a node $k$ in route $i$ that represents the link transmissions from node $k$ in route $i$. Equation (4.9) describes the strategy of finding the route with the highest minimum $AEF$ value which attempts to avoid routing through congested areas in the network. The original intention was to find the routes that have the highest capacity, i.e. to find routes capable of supporting large throughputs. However subsequent analysis showed that the effect of this route selection rule was to avoid congested nodes. The rationale of using the new route selection strategy is to find the bottleneck of each available route by determining the link with the lowest $AEF$ value. Then, the route with the highest minimum $AEF$ will be selected in order to optimize the global throughput of the network.

Figure 4.5 illustrates the operation of the route selection mechanism for the standard DSR protocol and the modified DSR protocol. In this figure, the original DSR protocol selects route $B$ as the hop count of this route is smaller than the hop count of the other paths (route $A$ and route $C$). While the modified DSR protocol chooses route $A$ over routes $B$ and $C$ as it selects the route with the highest minimum link capacity.
Exploring the network performance using AEF metric showed a significant improvement in the global throughput (defined as the total number of data bits per second received by the gateway node). However, this throughput improvement also has some drawbacks associated with it. One of the drawbacks is an increase in the average delay time (which is the average time required to transmit a packet from the source node to the destination node) of the network. Analysis carried out in chapter 5 will show that, the routing mechanism implementing AEF metric avoids congested areas by routing packets away from the congestion and hence the dropped packets at the nodes are reduced.

Routing packets around the congested areas requires using long transmission paths which results in an increase in the end-to-end delay. The other is that the smaller routed packets are penalized over the large packets in the sense that the smaller packets streams are routed away from the direct paths towards the gateway node. This means that the smaller packets streams take longer paths than the larger packet streams to reach their destination. This is
because the $AEF$ is dependent on the size of the routed packets. As described in section 2.3, the $AEF$ metric is defined by $BW_{load}$ component in addition to the $BW_{access}$ according to the Equation 2.10:

$$\eta_a = \frac{BW_{load}}{BW_{access}}$$  \hspace{1cm} (4.10)

Where $BW_{load}$ can be shown to be dependent on the packet size and packet rate according to:

$$BW_{load} \propto MSDU_{size} \times MSDU_{rate}$$ \hspace{1cm} (4.11)

Similarly,

$$BW_{access} \propto MSDU_{rate} \times T_{access}$$ \hspace{1cm} (4.12)

Consequently:

$$\eta_a \propto \frac{MSDU_{size}}{T_{access}}$$ \hspace{1cm} (4.13)

Where the $MSDU_{size}$ is the packet size generated by the sources in the network and the $MSDU_{rate}$ is the packet rate. $T_{access}$ represents the contention time for accessing the medium.

According to Equation (4.13), a station which is forwarding a smaller packet size will have a lower $AEF$ value, i.e. it will make this station appear to be more congested than it actually is. The modified path selection rule responds by routing packets away from the node, i.e. the routed packets will take longer transmission paths. This could be a problem if the network carries voice traffic. The selection of long paths for small packet streams is considered to be a drawback of the $AEF$ metric as it will result in increased transmission delays that decrease the quality of the voice.

In response to this drawback associated with implementing the $AEF$ metric in the $DSR$ protocol, an upper limit on the route lengths has been imposed to limit the increase in the global delay time of the network as will be explained in the next section. In order to
overcome the protocol drawback associated with small packets, a modification to the \( AEF \) metric has been introduced in section 4.5.3.

### 4.6.2 Second Modification to the DSR Routing Discovery Mechanism

To deal with the increase in the average delay time of the network when the \( AEF \) metric is employed, another modification to the routing mechanism has been implemented by incorporating the hop count in addition to the \( AEF \) in the routing discovery mechanism. The objective of using the hop count parameter in the routing discovery of the modified DSR protocol and limiting it to an upper bound is to control the average delay time of the network. This will allow the network administrator to impose an upper limit on the delay time. In other words, by tuning the hop count limit \( (H_{CL}) \) the network manager can trade-off throughput against delay according to network performance targets.

### 4.6.3 Third Modification to the DSR Routing Discovery Mechanism

A further modified version of the \( AEF \) (\( ModAEF \)) is introduced to address the drawback associated with the \( AEF \) metric. The modified \( AEF \) routes the larger packet streams away from the direct routes (i.e. usually more congested routes, this observation will be demonstrated later in Section 5.3.1) while the small packets tend to take shorter (i.e. more direct) routes towards the gateway node. In order to counter this, the \( AEF \) was modified by dividing it by the \( MSDU_{size} \) of the routed packet as follows:

\[
ModAEF = \frac{AEF}{MSDU_{size}} \quad (4.14)
\]
According to the Equation (4.14), a station with a small routed packet will have a bigger $\text{ModAEF}$ value than that of a larger packet. As a result the path length taken by the short packets will be on average shorter than that for large packets.

An analysis was carried out in respect of the network performance in Chapter 5 by using $\text{ModAEF}$ as a metric for the routing discovery mechanism. The results have shown the penalty applied to the large routed packets over the small packets by taking on average longer paths to reach the destination. To deal with this unfairness, an $\alpha$ parameter has been introduced as a tuning parameter to control the routing of the packets in the network, as shown below:

$$\text{ModAEF} = \frac{AEF}{MSDU_{size}^\alpha}$$

(4.15)

Tuning $\alpha$ allows the network operator to control the route lengths differences between the large and small routed packets. The examination also showed that the percentage throughput improvement (global throughput improvement against the standard DSR) is reduced for the modified DSR using $\text{ModAEF}$ metric. The use of $\text{ModAEF}$ exhibits less average delay time compared to the use of $AEF$.

### 4.7 Methodology

The performance of the network has been examined for different topologies by comparing the performance of the modified DSR protocol against the standard DSR protocol. The OPNET modeler was used to carry out this simulation. In this work, the investigation of the performance of the DSR modifications in terms of the global throughput and average delay time for different wireless mesh network scenarios have been carried out.
To evaluate the performance of implementing the newly introduced \textit{AEF} and \textit{ModAEF} metrics for the \textit{DSR} routing protocol, different mesh network scenarios have been designed. To perform this evaluation, the effect of the network density, packet size variations, packet rate variations, traffic type, and number of available gateways in the network, on the performance of the network have been studied. To validate the performance of the modified routing metric based upon the \textit{AEF}, a comparison has been made with the well known \textit{ETT} metric. These scenarios have been classified into groups based on the various network aspects (e.g. packet rate, packet size, etc.) as shown in the Table 4.3.

The OPNET modeler has been employed using the default IEEE 802.11b radio setting where the physical layer set to a direct-sequence spread spectrum (DSSS) with 11 Mbps data rate. OPNET has been used to simulate the performance of the modified \textit{DSR} protocol. The examined network scenarios have been classified into several groups of scenarios based on their configuration. All scenarios consist of 1000 randomly generated topologies. Each topology comprises 99 nodes randomly distributed across the network and a single gateway at a fixed location, the only exception is the group \textit{G}, see section 4.6.7, as it aims to examine the performance under a different number of gateways, see Figure 4.6. In this simulation model the area of the network and the number of nodes are assumed to be fixed. Different values of \textit{DF} are realized by varying the value of the transmission range. All the randomly distributed nodes generated Poisson traffic. The group \textit{F} is the only exception since nodes generate Pareto traffic. Pareto distribution is the simplest heavy-tailed distribution. The goal of implementing Pareto distribution is to investigate the network
performance with heavy-tailed traffic. Such traffic has implications for congestion control and traffic performance.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>No. GWs</th>
<th>Packet Size ($P_z$)</th>
<th>Packet Rate ($Pr$)</th>
<th>Metric</th>
<th>Traffic type</th>
<th>Scenario</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A$</td>
<td>1</td>
<td>512</td>
<td>5</td>
<td>$AEF$</td>
<td>Poisson</td>
<td>$DF$ variation</td>
</tr>
<tr>
<td>$B-1$</td>
<td>1</td>
<td>512</td>
<td>2.5</td>
<td>$AEF$</td>
<td>Poisson</td>
<td>$Pr$ variation</td>
</tr>
<tr>
<td>$B-2$</td>
<td>1</td>
<td>512</td>
<td>10</td>
<td>$AEF$</td>
<td>Poisson</td>
<td>$Pr$ variation</td>
</tr>
<tr>
<td>$C-1$</td>
<td>1</td>
<td>256</td>
<td>5</td>
<td>$AEF$</td>
<td>Poisson</td>
<td>$Pz$ &amp; $Pr$ variation</td>
</tr>
<tr>
<td>$C-2$</td>
<td>1</td>
<td>256</td>
<td>10</td>
<td>$AEF$</td>
<td>Poisson</td>
<td>$Pz$ &amp; $Pr$ variation</td>
</tr>
<tr>
<td>$C-3$</td>
<td>1</td>
<td>256</td>
<td>20</td>
<td>$AEF$</td>
<td>Poisson</td>
<td>$Pz$ &amp; $Pr$ variation</td>
</tr>
<tr>
<td>$D$</td>
<td>1</td>
<td>1500-512 - 128</td>
<td>5</td>
<td>$AEF$</td>
<td>Poisson</td>
<td>$Pz$ variation</td>
</tr>
<tr>
<td>$E$</td>
<td>1</td>
<td>256 - 512</td>
<td>10 - 5</td>
<td>$AEF$</td>
<td>Poisson</td>
<td>$Pz$ &amp; $Pr$ variation</td>
</tr>
<tr>
<td>$F$</td>
<td>1</td>
<td>512</td>
<td>5</td>
<td>$AEF$</td>
<td>Pareto</td>
<td>Traffic Type</td>
</tr>
<tr>
<td>$G$</td>
<td>1-4</td>
<td>512</td>
<td>5</td>
<td>$AEF$</td>
<td>Poisson</td>
<td>Gateway variation</td>
</tr>
<tr>
<td>$H$</td>
<td>1</td>
<td>512</td>
<td>5</td>
<td>$AEF$</td>
<td>Poisson</td>
<td>Downlink Stream</td>
</tr>
<tr>
<td>$I$</td>
<td>1</td>
<td>512</td>
<td>5</td>
<td>$AEF$</td>
<td>Poisson</td>
<td>TCP Traffic</td>
</tr>
<tr>
<td>$J$</td>
<td>1</td>
<td>512</td>
<td>5</td>
<td>$AEF – Hc$</td>
<td>Poisson</td>
<td>No. Path Limitation</td>
</tr>
<tr>
<td>$K-1$</td>
<td>1</td>
<td>512 - 256</td>
<td>5</td>
<td>$ModAEF$</td>
<td>Poisson</td>
<td>Modified version of $AEF$</td>
</tr>
<tr>
<td>$K-2$</td>
<td>1</td>
<td>1500-128</td>
<td>5</td>
<td>$ModAEF$</td>
<td>Poisson</td>
<td>$\alpha$ variation</td>
</tr>
<tr>
<td>$L$</td>
<td>1</td>
<td>512</td>
<td>5</td>
<td>$ETT &amp; MP$</td>
<td>Poisson</td>
<td>$ETT &amp; MP$ metrics</td>
</tr>
</tbody>
</table>
4.7.1 Scenarios of Group A

In this group of scenarios, the effect of the number of nodes that are located within the transmission range of a node on the performance of the modified DSR protocol has been investigated. For this purpose, ten scenarios of different network densities \((DF = 1, 1.5, 2, 2.5, 3, 4, 5, 6, 8, 10)\) have been formed. In all scenarios, each source has an average rate of 5 packets per second with a packet size set to 512 bytes. This per node condition insures that the network remains unsaturated as the routing protocol will not function effectively under saturation.

4.7.2 Scenarios of Group B

An investigation of the impact of the packet rate variations on the behaviour of the network using the \(AEF\) as criteria for the routing discovery mechanism is performed here. In this regard, two subgroups of scenarios \((B-1 \text{ and } B-2)\) have been established with different packet rates, each of which set to a different value as follows: 2.5 and 10 packets per second, see Table 4.3. The generated packet size for all scenarios is set to 512 bytes. Each subgroup consists of ten groups of scenarios of different network densities \((DF = 1, 1.5, 2, 2.5, 3, 4, 5, 6, 8, \text{ and } 10\) nodes).

4.7.3 Scenarios of Group C

In this group of scenarios, a further examination of the impact of the network load variations on the performance of the modified DSR protocol has been performed. The scenarios of this group are classified into three subgroups as shown below:
4.7.3.1 Scenarios of Group C-1

The influence of the variations of the network load on the performance of the AEF metric is explored here. Scenarios of $DF = 1, 1.5, 2, 2.5, 3, 4, 5, 6, 8, \text{ and } 10$ nodes, have been developed. The generated packet size for all scenarios under this group is set to 256 bytes and the packet rate is set to 5 packets per second.

4.7.3.2 Scenarios of Group C-2

More investigation has been made of the performance of the modified DSR protocol by setting the packet rate to 10 packets per second and the packet size is set to 256 bytes for all examined scenarios. Ten scenarios of $DF = 1, 1.5, 2, 2.5, 3, 4, 5, 6, 8, \text{ and } 10$ nodes, have been established for this purpose.

4.7.3.3 Scenarios of Group C-3

Ten different scenarios of $DF = 1, 1.5, 2, 2.5, 3, 4, 5, 6, 8, \text{ and } 10$ nodes, have been assigned to this group. All sources in these scenarios have set their packet sizes to 256 bytes and the packet rate to 20 packets per second.

4.7.4 Scenarios of Group D

The effect of mixed packet sizes at the nodes with uniform packet rate for all nodes is examined here. Ten scenarios of mixed packet sizes (128, 512, and 1500 bytes) generated by the sources with a fixed packet rate sets to 5 packets per second. Each scenario in this group sets to a specific network density $DF$ value as follows: 1, 1.5, 2, 2.5, 3, 4, 5, 6, 8, and 10. For each topology in this scenario, the nodes have been divided into three sets based on the sizes of generated packets. Each set consists of 33 nodes. The nodes in one of the sets
generate packets of size 128 bytes and the nodes in the other sets generate packets of size 512 bytes and 1500 bytes.

4.7.5 Scenarios of Group E

Ten scenarios of mixed packet sizes (512 and 256 bytes) generated by the source nodes and mixed packet rates (5 and 10 packets per second) with a $DF$ set to 1, 1.5, 2, 2.5, 3, 4, 5, 6, 8, and 10. In these scenarios, the nodes have been classified into two sets based on the packet sizes and packet rates generated by the nodes. One of the sets comprises of 50 nodes and the other one comprises of 49 nodes. The packet size and packet rate for each set of nodes adjusted to a different value as follows: packet size of 256 bytes with packet rate 10 packets per second and packet size of 512 bytes with packet rate 5 packets per second.

4.7.6 Scenarios of Group F

This scenario uses Pareto traffic sources to examine the performance of the modified $DSR$ protocol. Network traffic often exhibits self-similarity and dependencies over a long range of time scales [202, 203]. This is to be contrasted to Poisson traffic in its arrival and departure process. As mentioned before, Pareto distribution is the simplest heavy-tailed distribution [204]. It can be justified as realistic based on the observations of long-range dependence in some aggregate packet traffic streams [205]. The rationale of employing such traffic is to model the network traffic with one closer to the self-similarity characteristic. Under this group three scenarios of $DF = 2$, 4, and 6, have been formed. In all scenarios, the generated packet sizes at the nodes are set to 512 bytes and packet rates are set to 5 packets per second.
4.7.7 Scenarios of Group $G$

In this group of scenarios, an examination of the performance of the $AEF$ metric by applying different number of gateways to the network has been carried out. Four scenarios have been formed for this investigation. In one of the scenarios, a single gateway has been located in the centre of the network. In other scenarios two gateways, three gateways, and four gateways have been assigned to each of them. In these three scenarios the gateways are located in the edges of the network, see Figure 4.6. All scenarios are provided with 99 nodes randomly distributed across the network. The generated packet size at the nodes is set to 512 bytes and the packet rate to 5 packets per second.

![Figure 4.6: Network topology examples of different number of gateways nodes (in red).]
4.7.8 Scenarios of Group H

An examination of the performance of the new $AEF$ metric has been carried out by implementing a downlink traffic stream in WMNs. This examination is also carried out for WMNs that uses bidirectional traffic flow. For this purpose two different scenarios have been established, each of which consists of 1000 topologies with one gateway and 99 nodes randomly distributed across the network. The nodes of each topology have been divided into four sets of nodes as follows: a set of 25 nodes with transmission line rate of 11 Mbps, another set comprising 25 node with transmission line rate of 5.5 Mbps, a third set also consisting of 25 nodes with transmission line rate of 2 Mbps, and a fourth set consisting of 24 nodes with transmission line rate of 1 Mbps. Mixed line rates have been employed in these scenarios in order to take into account the dependency of the throughput on the line rate. The packet sizes have been set to 512 bytes and the $DF = 2$ for all topologies of these scenarios. For the bidirectional flow scenario, the nodes have been classified into two sets based on the packet rates generated by the nodes. One of the sets consists of 50 nodes that generate 5 packets per second and the other one consists of 49 nodes which receive 5 packets per second from the gateway. While for the downlink traffic flow scenario, the generated packet rate is set to 5 packets per second for all nodes.

4.7.9 Scenarios of Group I

To study of impact of using TCP traffic based Reno algorithm in WMNs when the new path selection mechanism based on the $AEF$ metric is employed in the $DSR$ protocol, a scenario of 1000 randomly distributed topologies comprising one gateway and 99 nodes has been established. For each topology, the $DF = 2$, the packet size is set to 512 bytes, and the packet rate is set to 5 packets per second. For each topology, four transmission line rates
have been used by dividing the nodes of each topology into four sets, one set consists of 25 nodes with a transmission line rate of 11 Mbps, a second set consisting of 25 nodes with 5.5 Mbps, a third set consisting of 25 nodes with 2 Mbps, and a fourth set consisting of 24 nodes with 1 Mbps.

4.7.10 Scenarios of Group J

The scenarios of this group include the hop count limit \((H_{CL})\) in addition to the \(AEF\) metric in the routing mechanism. The hop count is not employed as a metric for routing mechanism but instead is used to enforce an upper limit on the route lengths of the available routes between the source and the destination pair. The objective of using the \(H_{CL}\) is to limit the average delay time in the network. Tuning the \(H_{CL}\) will allow the network administrator to control the end-to-end delay time by setting the \(H_{CL}\) to an upper limit that satisfies some network requirements.

To examine the \(H_{CL}\) variations on the performance of the modified \(DSR\) protocol in terms of the global throughput and delay time, four scenarios have been created using various \(H_{CL}\) values \((H_{CL} = \infty, 7, 6, 5)\). The packet lengths are set to 512 bytes in all scenarios and the average packet rate is set to 5 packets per second.

4.7.11 Scenarios of Group K

The \(ModAEF\) metric used here is intended to investigate the penalty imposed on the small packets routed through the network over the larger packets as discussed in the section 4.5.3. In this group of scenarios, the \(ModAEF\) metric has been employed to investigate its
performance against that of the $AEF$ metric and to compare the lengths of the paths taken by the long and short packet sizes.

4.7.11.1 Scenarios of Group $K$-1

Three different scenarios of $DF = 2, 4, \text{ and } 6$ have been created using $ModAEF$ ($\alpha$ value has been set to 1) as a metric for the routing mechanism to evaluate the performance of the modification to the standard $DSR$. In all topologies, the packet sizes are set to 512 and 256 bytes and the packet rate is set to 5 packets per second. For each topology in this scenario, the nodes have been divided into two sets based on the generated packet size. One of the sets consists of 50 nodes with packet size of 512 bytes and the other set consists of 49 nodes with packet size of 256 bytes. These examinations have been carried out to compare the performance of the implementation of the $ModAEF$ metric in $DSR$ protocol with the performance of implementing the $AEF$ metric.

4.7.11.2 Scenarios of Group $K$-2

To analyze the effect of the packet size variations on the route length of the routed packets in the network, six scenarios have been formed using $ModAEF$ as a metric for the routing discovery mechanism with different setting for the $\alpha$ parameter ($\alpha = 0.2, 0.4, 0.6, 0.8, 1.5, 2$). In these scenarios, 50 nodes generate packets of size 128 bytes while the rest of the nodes (i.e. 49 nodes) generate packets with size of 1500 bytes. All the nodes in the network set their packet rates to 5 packets per second.

Based on these simulations, the relationship between the $DF$ and the global throughput of the network can be examined in order to study the effect of node density on the global
throughput of the network. The relationship between the $H_c$ and $DF$ has also been studied to investigate the role that the $DF$ plays in determining the lengths of the routes.

4.7.12 Scenarios of Group $L$

To evaluate the performance of the newly introduced metrics, the $ETT$ [104] and the $MP$ metrics [18] have been implemented in the $DSR$ protocol. Scenarios of different network densities have been designed using the $AEF$ metric on first instance and $ETT$ in the other as routing discovery criteria. Similarly, different network scenarios have been established using the $AEF$ metric on one instance and $MP$ in the other as routing discovery criteria. For this purpose, three different scenarios of $DF = 2, 4,$ and $6$ have been developed to compare the performance of the modified $DSR$ routing algorithm based on the $AEF$ metric against the $DSR$ based on the $ETT$. Three different scenarios of $DF = 2, 4,$ and $6$ have also been designed to compare the performance of the newly introduced path selection rule based on the $AEF$ metric against the path selection rule based on the $MP$ metric. The employed packet sizes in these scenarios are set to 512 bytes and the rate is set to 5 packets per second. In each network topology, the nodes have been divided into four sets of nodes one of which consist of 25 nodes with a transmission line rate of 11 Mbps, the second set consists of 25 nodes with a transmission line rate of 5.5 Mbps, the third set consists of 25 nodes with a transmission line rate of 2 Mbps, and the fourth set consists of 24 nodes with a transmission line rate of 1 Mbps.

4.8 Modeling Assumption

In this section, the many assumptions made for the simulation model in this work are described and justified. The main model assumptions are as follows:
• The area of the network and the number of nodes are fixed. The boundary edges of the plane are open (i.e. no reflections) and all nodes are randomly distributed within the plane. In this model, in order to realize different node density ($DF$) values, the transmission range of the network nodes are adjusted to achieve the required $DF$ value.

• Each node is identical with homogenous parameter settings.

• The simulator operates using a single fixed channel.

• Measuring the $AEF$ metric is carried out over the interval of 1 second. Based on the tests performed for the $AEF$ calculations for different intervals (0.5 second, 2 second, and 5 second), it has been found that as the calculation time interval for the $AEF$ metric is increased over 1 second the performance of the network is reduced in terms of global throughout and average delay time, see Figures L.1, L.2, L.3, and L.4 in Appendix L. This indicates that the accuracy of calculating the $AEF$ metric is degraded as the time interval is increased over 1 second. It has also been found the differences between the performance of the network when the calculation of the $AEF$ based on the interval of length 1 second and interval length of 0.5 second is negligible, see Figure L.5 and L.6 in Appendix L. For convenience, a 1 second interval was selected.

• This model assumes that the nodes employ omni-directional antennas. This results in a circular coverage area with no multipath fading. The $DF$ value affects two factors: the level of contention and level of connectivity. Reducing the $DF$ value causes a reduction in the level of the contention in the network. As a result, the performance of the network will be enhanced. At the same time, reducing the $DF$
value results in reduction in the level of the connectivity in the network which can adversely affect the performance of the network in terms of the global throughput due to the reduced number of available paths between the source-destination pair.

- The Free Space Loss (FSL) channel model was used for all network topologies. That means some propagation effects will not be taken into account such as fading, shadowing, and path attenuation. Under the current license of OPNET available it was not possible to investigate any link cost models other than FSL. Free-space propagation is considered to be unrealistic in wireless communications, because in reality obstacles and reflective surfaces will always appear in the propagation path. Along with attenuation caused by distance, a radiated wave will also lose energy through reflection, transmission and diffraction due to obstacles. The obtained results from using the FSL model will be overestimated.

- For the sake of the simplicity, the model assumes all nodes operate at a uniform transmission rate in the majority of the simulated network scenarios considered. Adapting such an assumption is to remove the dependency of the throughput on the transmission rate in the analysis. Reference [193] showed that nodes communicating with a single a gateway with different transmission rates cause throughput degradation [193] due to reduced transmission opportunities. Nodes with higher line rates have to wait longer for nodes with lower line rates to complete their transmissions. This will degrade the performance of the network.

- Poisson traffic sources are implemented in the majority of the simulated network scenarios of this model. It is widely used for convenience as it is easy to generate and to analyze [202]. As explained earlier, real network traffic often exhibits self-
similarity and long range dependence in contrast to Poisson traffic. However, with self-similar data, large values tend to come in clusters, and clusters of clusters, etc. This can have detrimental consequences for network performance due to the increases in the level of congestion in the network.

- Packet acknowledgement and retransmission attempts (a maximum of 4 attempts are allowed) are included in the model operation.
- Short preamble is employed in order to reduce the MAC overhead and hence improve the network performance.
- A user datagram protocol (UDP) traffic stream was employed in this study. The UDP protocol can be described as a connection-less protocol that does not require a connection between two points before the packets are sent. On the other hand the transmission control protocol (TCP) protocol requires the establishment of a connection between the source and destination before sending the data. It has been shown by [206] that the maximum possible level network performance in terms of global throughput can be achieved when the UDP is used. Based on the simulation of stationary scenarios introduced by Sun et al [201], Sun stated that the interactions among TCP, MAC, and routing protocol have a significant impact on the performance of the network. To avoid this interaction between the new routing selection mechanism based upon the \textit{AEF} metric and the TCP, the UDP is used in this study to avoid the conflict due to employing two flow control mechanisms in the network.
- In this study, the uplink traffic flow has been adopted due to the ease of implementing and analyse in the OPNET modeler. It will be shown later that
implementing uplink, downlink, and bidirectional flows produce more or less similar results, see Figures (5.16 and 5.18). It has been recognized that traffic flow in wireless networks tends to be highly asymmetric. In other words, the downlink traffic load usually greatly exceeds the uplink load. However, at Layer 2 when omni-directional antennas are used, the notion of uplink and downlink does not really apply. In other words, at Layer 2 within a wireless mesh node, the direction of the flow is largely irrelevant in terms of network performance, i.e. the node essentially broadcasts its frames in all directions.

4.9 Summary

This chapter gives an overview of the OPNET modeler with a description of the modification introduced to the DSR routing protocol, the model settings of the simulator, and the assumptions used in implementing the model. This modeler has been developed to implement the AEF as a cost metric for the DSR routing mechanism. The ModAEF metric has been introduced in this chapter to correct for the one of the limitations of the AEF owing to its dependency on the packet size. To overcome the other shortcoming exhibited by the modified DSR based on the AEF, the $H_{CL}$ is introduced to impose an upper limit on the available transmission paths between the source-destination pair. The simulation methodology and test scenarios by the OPNET modeler have been classified into several groups based on their configuration and are presented in this chapter.
CHAPTER 5

RESULTS AND DISCUSSION

Overview
This chapter presents and analyses the simulation results of incorporating a new path selection rule (based upon the $AEF$ metric and its subsequent refinements) in the routing mechanism of the $DSR$ protocol. The OPNET modeler has been employed in this work to analyse and evaluate the performance of the modified $DSR$ routing algorithm based upon the $AEF$ metric against the standard metric ($H_c$ metric) of the $DSR$ protocol. The performance of the modified version of $AEF$ ($ModAEF$ metric) was also examined against the $H_c$ metric and compared to the performance of the original $AEF$ metric. Imposing a hop count limit ($HCL$) on the length of the discovered transmission paths has also been examined in this chapter. The performance comparisons of the modified $DSR$ routing algorithm based upon the $AEF$ metric and the $DSR$ routing algorithm based upon the $ETT$ metric in one instance and the $DSR$ routing algorithm based upon the $MP$ metric in other have been performed.

5.1 Operation of the Modified $DSR$ Path Selection Rule
In this section, the operation of the modified $DSR$ path selection rule using the $AEF$ as the cost metric is analysed. The strategy behind the modification to the path selection rule of
the standard $DSR$ is to find the optimum path by selecting the path with the highest minimum $AEF$ value, see section 4.5.1, in order to avoid routing packets through areas of high congestion. The original intention was to find paths capable of supporting large throughputs. However, subsequent analysis revealed the actual operation of this mechanism resulted in congestion avoidance. Figure 5.1 demonstrates the basic operation of forwarding packets at a mesh node. The number of packets received by a node is determined by routing decisions made by the path selection rule. On the other hand, the number of packets transmitted by a node is determined by the availability of transmission opportunities. The availability of transmission opportunities is limited by the level of contention which is in turn determined by the number of other stations operating in the vicinity of the station also contending for access.

Figure 5.1: Basic packet forwarding operation at a node.
Packet dropping occurs when the number of packet arrivals exceeds the availability of transmission opportunities. In other words, packet dropping occurs as a result of the transmit buffer being full, which is a consequence of the arrival rate exceeding the service rate for prolonged intervals of time. Network congestion causes packets to remain in the transmit buffers for longer periods of time which causes the queue length to grow ultimately leading to packet loss (due to a finite buffer capacity).

![Figure 5.2: Operation of the modified DSR path selection rule based upon $AEF$ metric.](image)

The routing decision is made at the source node on the basis of a path selection rule which is in turn based upon a path selection metric, see Figure 5.2. The path selection metric implemented in this work is the $AEF$ value measured locally at a node which is determined
by the level of contention experienced by the node and by the load transmitted by the node (which is in turn determined by the routing decisions made by the path selection rule), see Equation 4.10. The modified DSR path selection rule finds the bottleneck in the available paths and then picks the path with the highest minimum $AEF$ value, see Equation 4.9. In other words, it selects routes containing the bottleneck node that is least likely to become congested. Employing such a routing mechanism helps to avoid routing packets through congested nodes. Figure 5.3 is an example of how the modified path selection rule results in the avoidance of congested nodes. In this example, if nodes Node$_j$ and Node$_i$ constitute the bottleneck of two available paths between the source and the destination nodes, the source node will select the path with the highest minimum $AEF$ value, i.e. the modified path selection rule at the source node will compare the $AEF$ value of each of these bottlenecks and then select the route with the least worst bottleneck. Effectively, this means that the route with higher capacity will be chosen.

The decision of selecting a route is made at the source node after receiving the RREP packets either from the destination node or from an intermediate node which has a route to the destination. The source node will store all the received routes in its route cache based upon the highest minimum $AEF$ value, i.e. the cached routes will be sorted based on the highest $\text{minAEF}$. The route with the highest $\text{minAEF}$ value will be stored on the top of the list and the one with the lowest $\text{minAEF}$ value will be inserted at the bottom of the list. Cached routes are assigned an expiration timer of length 10 seconds in order to refresh the cached routes.
The effect of other parameters such as packet size, packet rate, and transmission range on the performance of the modified path selection rule are investigated in this work. The interaction between these parameters and the local $AEF$ value at a node can be demonstrated in Figure 5.4. Increasing the transmission range of the nodes means an increase in the $DF$ value, i.e. increasing the number of potentially interfering nodes that are located within the reception range of a node. This will lead to an increase in the contention for transmission opportunities and hence an increase in the congestion in the network. On the other hand, increased contention will result in an increased $BW_{access}$ component which leads to a reduction in the $AEF$ value at the node, see Equation 4.10. The modified path selection rule will attempt to avoid routing through such nodes (i.e. nodes with a low $AEF$).
value) which results in longer transmission paths being taken by the streamed packets and hence an increase in the end-to-end delay. The effect of packet rate variation is also investigated in this work. It has been observed that increasing the packet rate degrades the performance of the network. This is due to increased contention and hence an increased level of congestion.

Increasing the routed packet sizes enhances the overall performance of the network due to the increased efficiency through using large packet sizes. Hence the capacity of the network will be maximized and the global throughput of the network will be enhanced, see section 5.2.2. In general, by using large packets, the most efficient use of the transmission opportunity will be made especially if the number of transmission opportunities is limited.
The analyses in this work show that the route selection rules based upon the $AEF$ metric exhibit better load distribution across the network nodes than the standard $DSR$ path selection rule, see Figure 5.5. This figure shows the PDF of the load distribution for standard $DSR$ routing algorithm and the modified $DSR$ routing algorithm for networks where $DF = 2$ with one gateway, packet rate is set to 5 packets per second, and packet size is set to 512 bytes.
Figure 5.5: Load distribution for the modified DSR against the standard DSR for networks of $DF = 2$.

A better load distribution resulting from the modified DSR path selection rule improves the network performance, due primarily to avoiding the creation of heavily loaded nodes. This mechanism reduces the amount of dropped packets at the nodes. To verify the throughput improvement of implementing the modified DSR routing algorithm over the standard DSR, the CDF of the packet loss ratio for 100 network topologies where $DF = 2$ are plotted, see Figure 5.6. For these topologies, the packet size is set to 512 bytes and the packet rate is set to 5 packets per second.
From this figure, it be seen that for the standard \textit{DSR} protocol 50\% of the random topologies experience a packet lost ratio less than 22\% compared with the modified \textit{DSR} where 50\% of the random topologies experience a packet lost ratio less than 7.5\%. It also can be seen that for a packet lost rate less than 10\%, no topologies using the standard \textit{DSR} routing algorithm achieve this performance compared to the modified \textit{DSR} routing algorithm, where 82\% of the random topologies experience a packet lost ratio less than 10\%. Based on this result, it can be seen that, the modified \textit{DSR} routing algorithm exhibits a significant reduction in the packet lost ratio compared to the standard \textit{DSR} due to avoiding routing through highly congested nodes. Figures 5.7 and 5.8 demonstrate the load distribution mechanism of the standard \textit{DSR} using the hop count metric and the modified \textit{DSR} using the \textit{AEF} metric for an arbitrarily selected network topology. In this topology,
using the standard DSR as the routing protocol results in high packet loss. The analysis shows that high packet loss occurs at nodes N_13, N_23, N_86, and N_89 in Figure 5.7 due to the buffer overflow. This is because the standard DSR is concerned with finding the shortest path without taking into account the congestion at the node. It leads to a number of heavily congested nodes in the vicinity of the gateway node, see Figure 5.7. Applying the modified DSR path selection rule reduces the packet lost by avoiding routing packets through these congested nodes, see Figure 5.8. In this example, it can be seen that the load of N_13, N_23, N_86, and N_89 is noticeably reduced and hence the number of dropped packets is reduced (see Table 5.1). The reduced load indicates the avoidance of routing through these congested nodes and alternative routes are found to forward the load. It can also be seen that the load of the gateway neighbour nodes (nodes that lie within the reception range of the gateway) such as nodes N_28, N_32, and N_38 has been increased while the load of N_23 is reduced where the modified DSR routing algorithm has been employed. Based on these analyses, it is clear that the modified path selection rule spreads the traffic across multiple gateway neighbour nodes which reduces the level of congestion. In other words, it exhibits better load distribution than the standard DSR which consequently reduces the packet loss across the network and hence the throughput of the network is optimized.

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Standard DSR</th>
<th>Modified DSR</th>
</tr>
</thead>
<tbody>
<tr>
<td>N_13</td>
<td>101</td>
<td>43</td>
</tr>
<tr>
<td>N_23</td>
<td>106</td>
<td>94</td>
</tr>
<tr>
<td>N_28</td>
<td>18</td>
<td>84</td>
</tr>
<tr>
<td>N_32</td>
<td>49</td>
<td>76</td>
</tr>
<tr>
<td>N_38</td>
<td>35</td>
<td>89</td>
</tr>
<tr>
<td>N_86</td>
<td>93</td>
<td>70</td>
</tr>
<tr>
<td>N_89</td>
<td>96</td>
<td>57</td>
</tr>
</tbody>
</table>
Figure 5.7: Load distribution for a particular topology using the standard DSR protocol for networks for $DF = 2$. 

Note: The number in bracket indicates the number packet transmitted.
Figure 5.8: Load distribution for a particular topology using the modified DSR protocol for networks for $DF = 2$.

The above analysis demonstrates the operation of the modified DSR routing mechanism by finding the bottleneck node for each selected route based on the $AEF$ value and then chooses the path with the least worst bottleneck node. This results in a better load distribution across the network by avoiding routing packets through heavily loaded nodes,
see Figure 5.8, which in turn reduces the contention around congested nodes. This reduction in the contention maximizes the global throughput of the network due to reduced packet loss at a congested node. In this example, the number of packets received by the gateway is 208 packets per second where the standard DSR protocol is implemented. While the number of packets received by the gateway is increased to 343 packets per second as a result of the modified DSR routing algorithm.

It has been demonstrated in this section that the modified DSR path selection rule exhibits greater load distribution across the nodes than the standard DSR. This does not imply a greater spatial distribution of the load across the network. A better load distribution results in reduction in the level of contention in the network and hence increases the possibility of winning a sufficient number of transmission opportunities. This will enhance the performance of the network in terms of the average global throughput. A path selection rule based on the AEF metric increases the average length of selected paths as it streams the traffic through long transmission paths in order to avoid highly congested areas. Consequently, the average delay time will be increased.

5.2 Performance Investigation of the Modified DSR Path Selection Rule

The modifications to the DSR routing mechanism in this work can be categorized into three stages. In the first stage, the modification to the DSR protocol involves a modification to the route discovery mechanism by using the AEF as an alternative metric to the hop count ($H_C$) and finds the route with the highest minimum AEF value. While in the second stage, the hop count limit ($H_{CL}$) is included in the modified routing mechanism to impose an upper limit on the path length of the available routes. Including the $H_{CL}$ in the routing
mechanism will allow the network operator to control the average delay time of the network in order to meet some network requirements. The third stage of the modification has been made by developing a modified version \((\text{ModAEF})\) of the \(\text{AEF}\) to be used in the newly introduced path selection rule, see section 4.5.3, to remedy a shortcoming of the \(\text{AEF}\) metric where it penalizes small routed packets over the large packets by streaming the smaller packets away from the direct path to the gateway node. That means, the smaller packet streams take longer paths than larger ones to reach their destination, resulting in greater delays. The \(\text{ModAEF}\) metric attempts to counter the dependence of the \(\text{AEF}\) on the packet size. The performance of this modified path selection rule is also evaluated against the path selection rule based upon the \(\text{ETT}\) metric of the \(\text{DSR}\) protocol.

In this work, 1000 random topologies for each scenario with one receiver (gateway node) and 99 senders (mesh nodes) have been generated. The gateway node has a fixed location in all topologies, the only exception is the group \(G\), see section 4.6.7, where it investigates the performance under a different number of gateways, see Figure 4.6. The simulator was run twice for each topology, once with the standard \(\text{DSR}\) followed by the modified \(\text{DSR}\). The global throughput was recorded for each 10 minute simulation run in order to calculate the percentage improvement for the particular topology. For each scenario the probability distribution function (PDF) and complementary cumulative distribution function (CCDF) of the global throughput improvement \((T_p)\) and the average delay time increment \((D_{inc})\) for the modified \(\text{DSR}\) against the standard \(\text{DSR}\) for all network topologies examined have been calculated.
The simulation results will be introduced in this chapter following the classifications of the
various test scenarios outlined in chapter 4.

5.2.1 Simulation Results of Implementing the Modified DSR Path Selection Rule

In this section, the experimental results for group A scenarios for the modified DSR
employing the AEF metric in the routing discovery mechanism are introduced. The
performance of the newly introduced route selection mechanism against the standard
mechanism has been examined for different network densities in terms of the global
throughput and average delay time. The performance of the modified DSR has been
examined for $DF = 1, 1.5, 2, 2.5, 3, 4, 5, 6, 8,$ and $10$ nodes. Each source in this group of
scenarios has an average packet rate ($P_r$) of 5 packets per second with a packet size ($P_z$) set
to 512 bytes. Figure 5.9 demonstrates the CCDF of the average global throughput
improvement ($T_p$) for the network scenarios with densities $DF = 1, 1.5, 2, 2.5,$ and $3$, for the
modified DSR. While Figure 5.10 demonstrates the CCDF of the $T_p$ for the network
scenarios of densities $DF = 4, 5, 6, 8,$ and $10$. The PDFs of all the other scenarios of this
group are given in Appendix A.
Figure 5.9: CCDFs of the percentage throughput improvement for \( DF = 1, 1.5, 2, 2.5, \) and 3 scenarios \([P_z = 512 \text{ B}, Pr = 5 \text{ pps}]\).

Figure 5.10: CCDFs of the percentage throughput improvement for \( DF = 4, 5, 6, 8, \) and 10 scenarios \([P_z = 512 \text{ B}, Pr = 5 \text{ pps}]\).
By using the CCDF for all the examined scenarios, it was possible to obtain the fraction of stations \((F_r)\) that exhibit a probability of percentage throughput improvement \((P_T)\) greater than or equal to 30\% and 50\% (for the purpose of comparing performances these two percentage improvement values have been adopted), see Table 5.2.

**Table 5.2 Probability Percentage Throughput Improvement for All Examined Scenarios with Different DF Values.**

<table>
<thead>
<tr>
<th>Density Factor (DF)</th>
<th>(P_T [\text{Improvement} \geq 30%])</th>
<th>(P_T [\text{Improvement} \geq 50%])</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>63%</td>
<td>43%</td>
</tr>
<tr>
<td>1.5</td>
<td>72%</td>
<td>52%</td>
</tr>
<tr>
<td>2</td>
<td>77%</td>
<td>56%</td>
</tr>
<tr>
<td>2.5</td>
<td>73%</td>
<td>47%</td>
</tr>
<tr>
<td>3</td>
<td>70%</td>
<td>43%</td>
</tr>
<tr>
<td>4</td>
<td>66%</td>
<td>37%</td>
</tr>
<tr>
<td>5</td>
<td>64%</td>
<td>32%</td>
</tr>
<tr>
<td>6</td>
<td>60%</td>
<td>30%</td>
</tr>
<tr>
<td>8</td>
<td>50%</td>
<td>9%</td>
</tr>
<tr>
<td>10</td>
<td>39%</td>
<td>3%</td>
</tr>
</tbody>
</table>

Using Table 5.2, the relationship between the node density factor \(DF\) and the percentage fraction of stations that exhibit a throughput improvement greater than or equal to 30\% and 50\% have been plotted, see Figure 5.11. This figure demonstrates that the highest \(F_r\) value occurs at \(DF = 2\) as the best balance between connectivity and contention appears at this value. In Figure 5.11, when the \(DF\) value exceeds 2 the \(F_r\) value decreases which means that an increased number of interfering nodes results in a reduction in the percentage fraction of stations that exhibit throughput improvement greater than or equal to 30\% and 50\%. It can also be observed from Figure 5.11 that reducing the value of the \(DF\) to less than 2 results in a reduction in \(F_r\) because of the reduced level of connectivity. Reduced
connectivity also results in a reduction in the global throughput improvement as the number of the available paths between the source and the destination is reduced.

![Fraction of Stations Satisfying the Specified Throughput Improvement](image)

Figure 5.11: Probability of percentage throughput improvement as a function of node density factor \([Pz = 512 \text{ B}, Pr = 5 \text{ pps}]\).

Figures 5.12 and 5.13 show the CCDF of the average delay time increment \((D_{inc})\) for the network densities of \(DF = 1, 1.5, 2, 2.5,\) and 3, and \(DF = 4, 5, 6, 8,\) and 10, for the modified DSR against the standard one. The PDFs of these network scenarios may be found in Appendix A.
Figure 5.12: CCDFs of the percentage delay increment for $DF = 1, 1.5, 2, 2.5, \text{ and } 3$ scenarios [$P_z = 512 \text{ B, } Pr = 5 \text{ pps}$].

Figure 5.13: CCDFs of the percentage delay increment for $DF = 4, 5, 6, 8, \text{ and } 10$ scenarios [$P_z = 512 \text{ B, } Pr = 5 \text{ pps}$].
By using the CCDF for all the examined scenarios, the fraction of stations \((F_r)\) that exhibits a probability percentage delay increment \((P_d)\) greater than or equal to 20% and 30% (for the purposes of the comparing performance these two percentage improvement values have been adopted) can be obtained and these are given in Table 5.3.

<table>
<thead>
<tr>
<th>Density Factor (DF)</th>
<th>(P_d[\text{Increment } \geq 20%])</th>
<th>(P_d[\text{Increment } \geq 30%])</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>24%</td>
<td>6%</td>
</tr>
<tr>
<td>1.5</td>
<td>32%</td>
<td>13%</td>
</tr>
<tr>
<td>2</td>
<td>33%</td>
<td>18%</td>
</tr>
<tr>
<td>2.5</td>
<td>42%</td>
<td>24%</td>
</tr>
<tr>
<td>3</td>
<td>47%</td>
<td>29%</td>
</tr>
<tr>
<td>4</td>
<td>54%</td>
<td>33%</td>
</tr>
<tr>
<td>5</td>
<td>55%</td>
<td>34%</td>
</tr>
<tr>
<td>6</td>
<td>57%</td>
<td>37%</td>
</tr>
<tr>
<td>8</td>
<td>66%</td>
<td>47%</td>
</tr>
<tr>
<td>10</td>
<td>70%</td>
<td>49%</td>
</tr>
</tbody>
</table>

The relationship between the \(DF\) and the percentage fraction of stations that exhibits increment in the average delay time greater than or equal to 20% and 30% has been also plotted, see Figure 5.14. In this figure, as the \(DF\) value is increased the percentage fraction of stations that exhibit delay increments greater than 20% and 30% is also increased due to an increased level of contention. In other words, increasing the level of contention leads to an increasing level of congestion in the network which is in turn leads to a reduced throughput and increased delay time.
In all the simulation scenarios considered here it has been shown that the newly introduced route selection rule based upon the AEF metric significantly improves the global throughput of the topology. However, this improvement in the throughput of the network is accompanied by an increase in the average delay time. This is because the modified routing algorithm avoids congested areas by routing packets away from the congestion, i.e. by taking longer transmission routes. Therefore the congestion avoidance strategy of the modified routing mechanism results in an increase in the average delay time of the network.
5.2.2 Effect of Load Variations

Further analysis of the impact of the network load variations on the performance of the modified DSR protocol is carried out in this section. The influence of the packet rate and packet size variations on the performance of the network when the AEF metric implemented in the modified routing mechanism of the DSR protocol has been analysed in this section. Following the classifications of the tested scenarios outlined in chapter 4, the simulation results of groups B, C, D, and E are introduced in this section.

Increasing the packet rate will increase the contention level in the network. In other words, the contention for transmission opportunities will be increased and hence the level of congestion is increased in the network. This increase in the contention results in an increased $BW_{access}$ component which leads to a reduction in the AEF value at the node, see Equation 4.10. The new path selection rule will attempt to avoid routing through such nodes (i.e. nodes with a low AEF value). This will result in longer transmission paths being selected and hence an increase in the end-to-end delay. Reducing the routed packet sizes degrades the overall performance of the network due to the reduced efficiency through using small packet sizes. Using small packet sizes results in less efficient use of the transmission opportunity especially if the number of transmission opportunities is limited. This will minimise the network capacity and hence the global throughput of the network is reduced.

For group B and C scenarios, the influence of the packet rate variations on the performance of the modified routing mechanism has been examined. The simulation results of group B scenarios are separated into two set of scenarios (B-1 and B-2), while the output of the
simulation for the group C scenarios is divided into three sets of scenarios (C-1, C-2, and C-3), the $P_z$ and $P_r$ are the criteria for this classification.

Both groups of scenarios have been tested with different $DF$ values ($DF = 1, 1.5, 2, 2.5, 3, 4, 5, 6, 8,$ and $10$). The $P_z$ is kept constant at 512 bytes and the $P_r$ is varied between 2.5 and 10 packets per second for group B scenarios. Scenarios B-1 represents a $P_z$ of 512 bytes and $P_r$ of 2.5 packets per second. While Scenarios B-2 represents a $P_z$ of 512 bytes and $P_r$ of 10 packets per second. For group C scenarios, the $P_r$ rate is set to 5, 10, and 20 packets per second for C-1, C-2, and C-3 sets respectively. In order to study the effect of varying the $P_z$ on the performance of the modified DSR protocol, smaller packet sizes (256 bytes) are assigned to the scenarios of this group compared to the size of the packets employed in the scenarios of the previous groups which is 512 bytes.

The analysis of the results for the average global throughput improvement and the delay time increment are plotted in the format of CCDFs for all scenarios of these groups, see Appendices B and C. The PDFs for all scenarios of this group may also be found in these Appendices. Utilizing the CCDFs for the scenarios of group B (Figures B.1.1, B.1.2, B.2.1, and B.2.2 in Appendix B) and group C (Figures C.1.1, C.1.2, C.2.1, C.2.2, C.3.1, and C.3.2 in Appendix C), the fraction of stations ($F_r$) that exhibit a probability of percentage throughput improvement ($P_T$) greater than or equal to 30% and 50% can be obtained and demonstrated in Table 5.4. While the fraction of stations ($F_r$) that exhibits a probability percentage delay increment ($P_D$) greater than or equal to 20% and 30% can be obtained from the CCDFs of group B (Figures B.1.13, B.1.4, B.2.13, and B.2.14 in Appendix B) and
group C (Figures C.1.13, C.1.14, C.2.13, C.2.14, C.3.13, and C.3.14 in Appendix C), see Table 5.5.

### Table 5.4 Probability Percentage Throughput Improvement for All Examined Scenarios for Different DF Values.

<table>
<thead>
<tr>
<th>DF</th>
<th>B-1</th>
<th>B-2</th>
<th>C-1</th>
<th>C-2</th>
<th>C-3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$P_T \geq 30%$</td>
<td>$P_T \geq 50%$</td>
<td>$P_T \geq 30%$</td>
<td>$P_T \geq 50%$</td>
<td>$P_T \geq 30%$</td>
</tr>
<tr>
<td>1</td>
<td>74%</td>
<td>49%</td>
<td>61%</td>
<td>40%</td>
<td>68%</td>
</tr>
<tr>
<td>1.5</td>
<td>78%</td>
<td>58%</td>
<td>63%</td>
<td>42%</td>
<td>72%</td>
</tr>
<tr>
<td>2</td>
<td>79%</td>
<td>59%</td>
<td>65%</td>
<td>44%</td>
<td>73%</td>
</tr>
<tr>
<td>2.5</td>
<td>73%</td>
<td>47%</td>
<td>60%</td>
<td>40%</td>
<td>67%</td>
</tr>
<tr>
<td>3</td>
<td>71%</td>
<td>45%</td>
<td>59%</td>
<td>38%</td>
<td>66%</td>
</tr>
<tr>
<td>4</td>
<td>70%</td>
<td>41%</td>
<td>57%</td>
<td>34%</td>
<td>62%</td>
</tr>
<tr>
<td>5</td>
<td>67%</td>
<td>37%</td>
<td>56%</td>
<td>29%</td>
<td>60%</td>
</tr>
<tr>
<td>6</td>
<td>65%</td>
<td>35%</td>
<td>54%</td>
<td>27%</td>
<td>58%</td>
</tr>
<tr>
<td>8</td>
<td>58%</td>
<td>13%</td>
<td>44%</td>
<td>2%</td>
<td>45%</td>
</tr>
<tr>
<td>10</td>
<td>48%</td>
<td>8%</td>
<td>30%</td>
<td>0%</td>
<td>35%</td>
</tr>
</tbody>
</table>

### Table 5.5 Probability Percentage Delay Increment for All Examined Scenarios for Different DF Values.

<table>
<thead>
<tr>
<th>DF</th>
<th>B-1</th>
<th>B-2</th>
<th>C-1</th>
<th>C-2</th>
<th>C-3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$P_D \geq 20%$</td>
<td>$P_D \geq 30%$</td>
<td>$P_D \geq 20%$</td>
<td>$P_D \geq 30%$</td>
<td>$P_D \geq 20%$</td>
</tr>
<tr>
<td>1</td>
<td>24%</td>
<td>5%</td>
<td>27%</td>
<td>9%</td>
<td>24%</td>
</tr>
<tr>
<td>1.5</td>
<td>26%</td>
<td>6%</td>
<td>31%</td>
<td>12%</td>
<td>27%</td>
</tr>
<tr>
<td>2</td>
<td>28%</td>
<td>7%</td>
<td>33%</td>
<td>13%</td>
<td>29%</td>
</tr>
<tr>
<td>2.5</td>
<td>38%</td>
<td>19%</td>
<td>44%</td>
<td>25%</td>
<td>41%</td>
</tr>
<tr>
<td>3</td>
<td>46%</td>
<td>27%</td>
<td>51%</td>
<td>32%</td>
<td>47%</td>
</tr>
<tr>
<td>4</td>
<td>53%</td>
<td>32%</td>
<td>57%</td>
<td>37%</td>
<td>55%</td>
</tr>
<tr>
<td>5</td>
<td>54%</td>
<td>33%</td>
<td>59%</td>
<td>40%</td>
<td>56%</td>
</tr>
<tr>
<td>6</td>
<td>56%</td>
<td>33%</td>
<td>60%</td>
<td>40%</td>
<td>58%</td>
</tr>
<tr>
<td>8</td>
<td>65%</td>
<td>44%</td>
<td>70%</td>
<td>50%</td>
<td>68%</td>
</tr>
<tr>
<td>10</td>
<td>68%</td>
<td>47%</td>
<td>72%</td>
<td>52%</td>
<td>71%</td>
</tr>
</tbody>
</table>
In the previous groups, each topology has been examined with equal-sized packets at a different packet rates. The simulation results of the examined topologies with varied packet sizes and uniform packet rate are introduced in group $D$ scenarios. For this purpose, the simulation results of ten scenarios of $DF = 1, 1.5, 2, 2.5, 3, 4, 5, 6, 8$, and $10$, with packet sizes set to 128, 512, and 1500 bytes and packet rates are set to 5 packet per second have been demonstrated for this group. For more details about this group see section 4.6.4. The CCDFs and PDFs of the global throughput improvement and average delay time for the all network scenarios for the modified $DSR$ are shown in Appendix $D$, see Figures D.1, D.2, D.13, and D.14.

To investigate the performance of the modified $DSR$ protocol, further examination has been made by employing different packet sizes and packet rates in the network which is introduced in the group $E$ scenarios. The simulation results of scenarios with different $DF$ values using packet sizes 256 and 512 bytes and packet rates 10 and 5 packets per second have been presented here. For each topology, the nodes are divided into two sets: one of the sets consists of 50 nodes with generated packet size of 256 bytes and packet rate of 10 packets per second and the other set consists of 49 nodes where the generated packet sizes are set to 512 bytes with packet rate is set to 5 packets per second. Ten scenarios have been established with $DF$ sets to 1, 1.5, 2, 2.5, 3, 4, 5, 6, 8, and 10. The CCDF (Figures E.1, E.2, E.13, and E.14) and PDF of the global throughput improvement and average delay time for all the network scenarios for the modified $DSR$ may be found in Appendix $E$. By utilizing the CCDF for the examined scenarios of these two groups, it was possible to obtain the fraction of stations ($F_r$) that exhibit a probability of percentage throughput improvement ($P_T$) greater than or equal to 30% and 50%, see Table 5.6.
TABLE 5.6 PROBABILITY PERCENTAGE THROUGHPUT IMPROVEMENT FOR ALL EXAMINED SCENARIOS FOR DIFFERENT DF VALUES.

<table>
<thead>
<tr>
<th>DF</th>
<th>$P_T \geq 30%$</th>
<th>$P_T \geq 50%$</th>
<th>$P_T \geq 30%$</th>
<th>$P_T \geq 50%$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>65%</td>
<td>42%</td>
<td>68%</td>
<td>48%</td>
</tr>
<tr>
<td>1.5</td>
<td>69%</td>
<td>45%</td>
<td>71%</td>
<td>51%</td>
</tr>
<tr>
<td>2</td>
<td>71%</td>
<td>47%</td>
<td>70%</td>
<td>50%</td>
</tr>
<tr>
<td>2.5</td>
<td>70%</td>
<td>46%</td>
<td>67%</td>
<td>45%</td>
</tr>
<tr>
<td>3</td>
<td>68%</td>
<td>44%</td>
<td>66%</td>
<td>40%</td>
</tr>
<tr>
<td>4</td>
<td>65%</td>
<td>40%</td>
<td>64%</td>
<td>36%</td>
</tr>
<tr>
<td>5</td>
<td>61%</td>
<td>34%</td>
<td>63%</td>
<td>31%</td>
</tr>
<tr>
<td>6</td>
<td>57%</td>
<td>27%</td>
<td>59%</td>
<td>28%</td>
</tr>
<tr>
<td>8</td>
<td>48%</td>
<td>5%</td>
<td>48%</td>
<td>7%</td>
</tr>
<tr>
<td>10</td>
<td>34%</td>
<td>1%</td>
<td>35%</td>
<td>2%</td>
</tr>
</tbody>
</table>

The fraction of stations ($F_r$) that presents a probability percentage delay increment ($P_D$) greater than or equal to 20\% and 30\%, see Table 5.7.

TABLE 5.7 PROBABILITY PERCENTAGE DELAY INCREMENT FOR ALL EXAMINED SCENARIOS FOR DIFFERENT DF VALUES.

<table>
<thead>
<tr>
<th>DF</th>
<th>$P_D \geq 20%$</th>
<th>$P_D \geq 30%$</th>
<th>$P_D \geq 20%$</th>
<th>$P_D \geq 30%$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>32%</td>
<td>14%</td>
<td>30%</td>
<td>12%</td>
</tr>
<tr>
<td>1.5</td>
<td>36%</td>
<td>17%</td>
<td>33%</td>
<td>16%</td>
</tr>
<tr>
<td>2</td>
<td>38%</td>
<td>20%</td>
<td>35%</td>
<td>18%</td>
</tr>
<tr>
<td>2.5</td>
<td>43%</td>
<td>25%</td>
<td>44%</td>
<td>26%</td>
</tr>
<tr>
<td>3</td>
<td>48%</td>
<td>29%</td>
<td>50%</td>
<td>30%</td>
</tr>
<tr>
<td>4</td>
<td>55%</td>
<td>34%</td>
<td>56%</td>
<td>36%</td>
</tr>
<tr>
<td>5</td>
<td>60%</td>
<td>41%</td>
<td>58%</td>
<td>39%</td>
</tr>
<tr>
<td>6</td>
<td>62%</td>
<td>43%</td>
<td>60%</td>
<td>41%</td>
</tr>
<tr>
<td>8</td>
<td>70%</td>
<td>51%</td>
<td>69%</td>
<td>49%</td>
</tr>
<tr>
<td>10</td>
<td>72%</td>
<td>53%</td>
<td>71%</td>
<td>52%</td>
</tr>
</tbody>
</table>
In comparison with the results for the group A presented in section 5.2.1, the results of group B show that the lowest throughput improvement occurred for the case when the packet rate has been increased to 10 packets per second. According to these results, fixing the size of the generated packets and increasing the packet rate causes a reduction in the performance of the network. The global throughput improvement has decreased as the $P_r$ increases. This reduction is related to the increase in the level of contention in the network and hence the congestion will be increased. This increase in the congestion introduces an increase in the average delay time of the network due to the congestion avoidance strategy of the modified DSR path selection rule. This congestion avoidance routing mechanism streams the routed packets over longer transmission paths. In other words, increasing the contention between nodes leads to a reduction in the $AEF$ value at the nodes, see Equation 4.13. The modified path selection rule attempts to avoid routing through these nodes (nodes with lower $AEF$ value) by finding alternative longer routes. Consequently, the end-to-end delay time is increased.

Increasing the packet rate also raises the possibility of collisions in the network which leads to an increase in the retransmission attempts in the network. Retransmissions cause the packets to remain longer in the buffer (while awaiting a successful transmission) resulting in a reduced service rate (i.e. transmission rate) leading to a higher probability of buffer overflow and subsequent packet loss. As a consequence, the performance of the network is reduced in terms of a decreased throughput and increased delay time, see Figures B.1.1, B.1.2, B.2.1, B.2.2, and B.25 in Appendix B.
In comparison to the results of group $A$, reducing the packet size leads to a reduction in the global throughput of the network. This is due to the reduced efficiency of using small packets, i.e. less efficient use of the transmission opportunities, see Figure C.25 in Appendix C. In addition, reducing the packet size results in a reduction in the $AEF$ value at the nodes since the $AEF$ is proportional to the packet size (packet size dependence), see Equation 4.13. This will make a nodes appear to be more congested than it actually is which causes packets to be routed away from the node, i.e. the routed packets will take longer transmission paths, see Figure C.26 in Appendix C and Figure D.26 in Appendix D.

It should also be noted that packets of small size incur a relatively large overhead due to UDP and IP headers resulting in the inefficient use of the network resource [207, 208]. Based on these results, it can be seen that reducing the packet size or increasing the packet rate introduces a reduction in the global throughput improvement. This reduction in the throughput improvement is accompanied by an increase in the average delay time, see Figure C.26 in Appendix C.

It is also noted that using mixed packet sizes and packet rates in the network has shown an affect on the performance of the modified routing mechanism. In comparison with the group $A$ scenarios, Figures E.1, E.2, E.13, E.14, E.25, and E.26 in Appendix E demonstrate that the global throughput has been reduced and average delay time has been increased when the packet rate is increased and the packet size is reduced for 50 nodes of the network. Again, this performance degradation is due to the increased congestion in the network resulting in higher dropped packets. The increased average delay time arises from
the congestion avoidance strategy of the modified DSR path selection rule by routing the traffic along longer transmission paths to avoid congested nodes.

Even the scenarios of these groups exhibit a reduction in the global throughput compared to the results of group $A$, but still show a significant improvement in the global throughput of the network compared to the standard DSR protocol. As explained previously, implementing such a path selection rule using the AEF as a link cost metric helps the packet streams to avoid congested areas.

5.2.3 Effect of Traffic Type

Real traffic usually exhibits self-similarity and long-range-dependence properties. It is usually more bursty (unevenness or variations in the traffic flow) than Poisson traffic. Self-similar traffic exhibits high-variability and persistence of clustering (consecutive payloads consisting of identical attributes of data packets, i.e. back-to-back packets of similar attributes) which have a negative impact on network performance, as it leads to increase congestion in the network. With Poisson traffic, clustering occurs in the short term but smoothes itself out over the long term. With long-tail traffic, the bursty behaviour may itself be bursty, which in turn could intensify the clustering phenomena, and resulting in network performance degradation. Traffic self-similarity adversely affects performance measures such as queue size and packet-loss rate. The queue length distribution of long-tail traffic decays more slowly compared to Poisson sources which possesses exponentially decaying tails [209]. It has been shown by Park et al that as self-similarity increases, the network throughput declines gradually and queuing delay increases more dramatically [210]. An extremely large buffer capacity is required as self-similarity is increased in order
to achieve a constant level of throughput or packet loss. However, increased buffering leads to large queuing delays and thus self-similarity significantly increases the steepness of the trade-off curve between throughput/packet loss and delay [210]. For modeling the network with heavy-tailed traffic, Pareto traffic source is used here as it is considered the simplest heavy-tailed traffic model. Three different scenarios of $DF = 2, 4, \text{ and } 6$, have been established. The generated packet sizes at the sources are set to 512 bytes and packet rate is set to 5 packets per second. The objective of this test is to analysis the performance of the modified DSR protocol using heavy-tailed traffic.

This section presents and discusses the simulation results of the global throughput improvement and average delay time increment for the network scenarios with $DF = 2, 4, \text{ and } 6$, using Pareto Traffic source. The results of these scenarios are expressed in the format of a CCDF, see Figures F.1, F.2, F3, F.7, F.8, and F.9 in Appendix F. The PDFs of these scenarios can be also found in Appendix F. Based on the CCDF for all these examined scenarios, it is possible to extract the fraction of stations ($F_r$) that exhibit a probability of percentage throughput improvement ($P_T$) greater than or equal to 30% and 50%, see Table 5.8. Also the fraction of stations ($F_r$) that presents a probability percentage delay increment ($P_D$) greater than or equal to 20% and 30% can be obtained, see Table 5.9.

<table>
<thead>
<tr>
<th>Density Factor($DF$)</th>
<th>$P_T$ [Improvement $\geq 30%$]</th>
<th>$P_T$ [Improvement $\geq 50%$]</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>73%</td>
<td>53%</td>
</tr>
<tr>
<td>4</td>
<td>60%</td>
<td>31%</td>
</tr>
<tr>
<td>6</td>
<td>52%</td>
<td>23%</td>
</tr>
</tbody>
</table>

 TABLE 5.8 PROBABILITY PERCENTAGE THROUGHPUT IMPROVEMENT FOR ALL EXAMINED SCENARIOS FOR DIFFERENT $DF$ VALUES.
<table>
<thead>
<tr>
<th>Density Factor(DF)</th>
<th>(P_d[\text{Increment} \geq 20%])</th>
<th>(P_d[\text{Increment} \geq 30%])</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>37%</td>
<td>18%</td>
</tr>
<tr>
<td>4</td>
<td>48%</td>
<td>27%</td>
</tr>
<tr>
<td>6</td>
<td>64%</td>
<td>44%</td>
</tr>
</tbody>
</table>

Figures F.1, F.2, and F.3 in Appendix F, show a comparison with the performance of group A and group F scenarios in term of \(T_p\). Based on those figures, a degradation in the performance of the modified DSR protocol has been shown by the bursty traffic (Pareto traffic) in comparison to the Poisson traffic. Employing such bursty traffic causes more congestion to occur in the network due to it increasing the queuing delay. With long-range dependent traffic sources, a trade-off relationship exists between queuing delay and packet loss rate, the high increase in queuing delays at relatively low levels of utilization and slow decay of queue lengths implies a high level of packet loss [211, 212]. The modified path selection rule performs effectively even when the heavy-tailed traffic is employed. It outperforms the standard path selection rule in terms of global throughput, see Figures F.7, F.8, and F.9 in Appendix F.

### 5.2.4 Effect of the Number of Available Gateways

An evaluation of the performance of the modified routing mechanism of the DSR protocol using the \(AEF\) as a cost metric has been carried out by using different number of gateways. For this purpose, four scenarios of \(DF = 2\) with different number of gateways have been established. In one of the scenarios, a single gateway has been located in the centre of the network. In other scenarios two gateways, three gateways, and four gateways have been
allocated to each of them and located at the edges of the network, see Figure 4.6. The generated packet size at the nodes is set to 512 bytes and the packet rate to 5 packets per second.

Figures G.1 and G.6 in Appendix G, show the CCDF of the global throughput improvement and average delay time increment for the network scenarios of $DF = 2$ with different gateways for the modified DSR. The PDFs of these scenarios are given in Appendix G. By using the CCDF for all the examined scenarios of this group, it was possible to obtain the fraction of stations ($F_r$) that exhibit a probability of percentage throughput improvement ($P_T$) greater than or equal to 30% and 50%, see Table 5.10. The fraction of stations ($F_r$) that exhibits a probability percentage delay increment ($P_D$) greater than or equal to 20% and 30% can be also obtained, see Table 5.11.

### Table 5.10 Probability Percentage Throughput Improvement for All Examined Scenarios.

<table>
<thead>
<tr>
<th>No of Gateway (GW)</th>
<th>$P_T$ [Improvement $\geq$ 30%]</th>
<th>$P_T$ [Improvement $\geq$ 50%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>77%</td>
<td>56%</td>
</tr>
<tr>
<td>1 (centered)</td>
<td>68%</td>
<td>48%</td>
</tr>
<tr>
<td>2</td>
<td>60%</td>
<td>41%</td>
</tr>
<tr>
<td>3</td>
<td>49%</td>
<td>29%</td>
</tr>
<tr>
<td>4</td>
<td>37%</td>
<td>20%</td>
</tr>
</tbody>
</table>

### Table 5.11 Probability Percentage Delay Increment for All Examined Scenarios.

<table>
<thead>
<tr>
<th>No of Gateway (GW)</th>
<th>$P_D$ [Increment $\geq$ 20%]</th>
<th>$P_D$ [Increment $\geq$ 30%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>33%</td>
<td>18%</td>
</tr>
<tr>
<td>1 (centered)</td>
<td>31%</td>
<td>14%</td>
</tr>
<tr>
<td>2</td>
<td>27%</td>
<td>9%</td>
</tr>
<tr>
<td>3</td>
<td>20%</td>
<td>4%</td>
</tr>
<tr>
<td>4</td>
<td>11%</td>
<td>0%</td>
</tr>
</tbody>
</table>
The results show that increasing the number of gateways results in a reduction in the global throughput improvement which is accompanied by a reduction in the average delay of the network, see Figures G.1 and G.6 in Appendix G. As the number of gateway nodes is increased the level of congestion is reduced around the gateway node. Adding more gateway nodes to the network generates shorter routes, so it increases the overall performances; but it also gives more possibilities for the path selection rule to choose which gateway to route traffic towards. Introducing more gateways to a network will produce smaller number of congestion regions and will result in an enhanced network performance. The modified path selection rule of the DSR protocol works most effectively in WMNs where there are localized regions of high congestion, i.e. congested nodes, are increased. Therefore, the introduction of additional gateway nodes reduces the number of regions of localized congestion. As a consequence, the benefit of using the modified DSR over the standard DSR is reduced. Figures G.1 and G.6 in Appendix G indicate that locating a gateway node at the centre of the network exhibits a reduction in the average delay time increment and global throughput improvement compared to a single gateway positioned at the edge of the network. This is also related to the level of congestion around the gateway. The level of congestion around a gateway positioned in the centre of the network is less than the level of congestion around a gateway positioned at the edge of the network. This is due to increased number of available transmission paths to the gateway.

The analysis illustrates that the number of available gateway nodes affects the distribution of the load across the network as shown in Figure 5.15. Increasing the number of gateways in a network improves the overall performance of the network through a better load
distribution. This results in a reduction in the level of interference in the network and hence improves the overall performance of the network.

![PDF of load distribution](image)

Figure 5.15: Load distribution for the modified DSR using different number of gateways for network with $DF=2$.

5.2.5 Uplink and Downlink Traffic Stream

In this section, two simulation scenarios have been established in order to investigate the performance of the new AEF path selection rule with differently directed traffic streams. One of the scenarios employs downlink traffic streams where the traffic load directed from the gateway node towards the nodes distributed across the network. The other scenario employs bidirectional traffic streams where nodes in the network send and receive packets
to and from the gateway node. The simulation results of these two scenarios are compared to the simulation results of the case when only uplink traffic stream is implemented.

The network nodes of each topology in these scenarios are divided into four sets, the first set is consists of 25 nodes with transmission rate 11 Mbps. The second set consists of 25 nodes with transmission rate of 5.5 Mbps. The third set consists of 25 nodes with transmission rate of 2 Mbps. The fourth set consists of 24 nodes with transmission rate of 1 Mbps. Different line rates have been used here in order to take into account consideration the dependency of the throughput on the network. In the bidirectional flow scenario, the nodes have been classified into two sets based on the packet rates generated by the nodes. One of the sets consists of 50 nodes that generate 5 packets per second and the other one consists of 49 nodes which receive 5 packets per second from the gateway. While the packet rate at the nodes of the downlink traffic flow scenario is set to 5 packets per second. The packet size is set to 512 bytes and $DF = 2$ for all topologies of these scenarios. The simulation results in the form of the CCDFs for the examined scenario where the downlink traffic stream is employed against the uplink traffic stream in terms of the global throughput improvement and average delay time increment are presented in Figures 5.16, 5.17, 5.18, and 5.19.
Figure 5.16: CCDF of the percentage throughput improvement for downlink traffic stream and the uplink traffic stream scenarios.

Figure 5.17: CCDF of the percentage delay increment for the downlink traffic stream and the uplink traffic stream scenarios.
The above figures show that simulation results of implementing downlink and uplink traffic streams are essentially similar. This is because the new AEF path selection rule reacts to any changes in the network regardless of the traffic directions. This is also applied to the case where bidirectional traffic flows are present, see Figures 5.18 and 5.19.

Figure 5.18: CCDF of the percentage throughput improvement for downlink and uplink traffic stream and the uplink traffic stream scenarios.
The traffic flow in wireless networks tends to be highly asymmetric where the downlink traffic load usually greatly exceeds the uplink traffic load. At layer two, the notion of uplink and downlink streams does not really apply when omni-directional antennas are used since the node essentially broadcasts its frames in all directions. This has been demonstrated through the examination applied to different network scenarios where uplink, downlink, and bidirectional traffic flows were employed.

The new route selection rule results in a distributed routing scheme therefore there is a lack of coordination in terms of route selection, i.e. there is no communication between nodes regarding the choice of selected routes. A potential drawback of this path selection rule is
when multiple source nodes share the same intermediate node to forward their traffic. This may cause congestion at that intermediate node and the algorithm will react to this by attempting to select an alternative route. Potentially they can simultaneously react in the same way by selecting the same alternative route. As each node is unaware of the route selected by its neighbours and potentially a group of nodes can pick the same route this gives rise to further congestion at another intermediate node. A route flip-flopping between intermediate nodes can occur as a result. This condition can be observed in the sparse networks where the number of the available routes is very limited. Multiple source nodes might share the same intermediate node in order to forward their traffic. In addition, this condition might also occur when there are some nodes located at the edges of the network where the number of available routes is limited. However, this condition could be avoided by using another path selection rule that is based on finding the average access efficiency factor rather than using the access efficiency factor of the bottleneck node of the discovered routes. Using the average $AEF$ may allow the route selection mechanism to choose a route containing a severely congested bottleneck node as the presence of such a node may be masked within the calculation of the average value. That means a higher probability of packet loss occurring. The main reason behind the use of the $\text{minAEF}$ metric is to identify the bottleneck node of the available routes. Finding the route with the highest $\text{minAEF}$ will allow the route selection mechanism to select the route with the least worst bottleneck.

### 5.2.6 Transport Protocol

The impact of using a transport protocol based on flow control mechanism on the performance of the $DSR$ protocol based on the $AEF$ metric has been investigated. In this regards, the TCP protocol has been implemented for a scenario of 1000 topologies of $DF =$
2, packet size 512, and packet rate 5 packets per second. For each network topology, the nodes in the network classified into four sets, one consists of 25 nodes with a transmission line rate of 11 Mbps, the second set also consists of 25 nodes but with transmission line rate of 5.5 Mbps, the third set consists of 25 nodes with a transmission line rate of 2 Mbps, and the fourth set consists of 24 nodes with a transmission line rate of 1 Mbps. To more accurately model a typical wireless network where a number of different line rates will be used.

The global throughput and the average delay time were recorded for each 10 minute simulation run. The CCDFs of the global throughput improvement and the average delay time increment for the modified DSR routing algorithm based upon the AEF metric against the standard DSR for all network topologies examined have been calculated. The results are compared to the CCDFs of the global throughput improvement and average delay time increment for the modified DSR against the standard DSR using when the UDP traffic is used, see Figures 5.20 and 5.21.
Figure 5.20: CCDF of the percentage throughput improvement for TCP traffic against UDP.

Figure 5.21: CCDF of the percentage delay increment for TCP traffic against UDP.
The simulation results indicate that the TCP traffic results in a significant reduction in the performance of the modified DSR protocol based upon the AEF metric in terms of global throughput improvement. This is due to the flow control mechanism used by the TCP protocol which results in a conflict with the congestion avoidance mechanism based upon the AEF metric introduced in this work. This is because the TCP reacts faster than the AEF path selection rule to the occurrence of congestion in the network. The TCP mechanism is based on the round trip time which is of the order of a few milliseconds while the AEF which is based on a 1 second update time. When the TCP detects the occurrence of congestion it halves the transmission rate which consequently reduces the global throughput of the network. The reduction in the average time increment is a consequence of the round trip time mechanism used by the TCP protocol which results in reduction in the delay time of the network.

5.2.7 Incorporation of the Hop Count Parameter

A modification to the DSR protocol has been made in this work by including the AEF parameter in addition to the hop count in the routing discovery mechanism. The hop count is not employed as a metric for routing mechanism, instead it is used to eliminate long paths that have been selected by the AEF based routing mechanism, i.e. it enforces an upper limit on the route lengths of the available routes between the source and the destination node pairs. The goal of the work in this section is to analyse the performance of the modified path selection rule with the hop count limit ($H_{CL}$) against the standard DSR for all examined scenarios. The $H_{CL}$ is set to a different value for each scenario. The idea behind imposing a limit on the hop count is to control the delay time in the network.
In this section, the examination of the performance of the modified DSR applied to different network scenarios of $DF = 2$ with different hop count limits ($H_{CL} = \infty, 7, 6, \text{and } 5$). For each scenario the CCDF and PDF of the throughput improvement and the delay increase for all network topologies examined have been calculated.

The CCDFs of the four scenarios using various hop count limit values ($H_{CL} = \infty, 7, 6, 5$), which represent the global throughput improvement and the average global delay time increase of the modified DSR routing algorithm against the standard DSR routing algorithm with different hop count limits are presented in Appendix $H$, see Figures H.1 and H.5. The packet lengths are set to 512 bytes in all scenarios and the average packet rate is set to 5 packets per second. The PDFs of these scenarios are also shown in Appendix $H$. Using the CCDFs of the global throughput improvement, the fraction of stations ($F_r$) that exhibit a probability of percentage throughput improvement ($P_T$) greater than or equal to 30% and 50% can be obtained and are shown in Table 5.12.

<table>
<thead>
<tr>
<th>Hop Count ($H_{CL}$)</th>
<th>$P_T$ [Improvement $\geq 30%$]</th>
<th>$P_T$ [Improvement $\geq 50%$]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\infty$</td>
<td>77%</td>
<td>56%</td>
</tr>
<tr>
<td>7</td>
<td>67%</td>
<td>45%</td>
</tr>
<tr>
<td>6</td>
<td>60%</td>
<td>35%</td>
</tr>
<tr>
<td>5</td>
<td>40%</td>
<td>20%</td>
</tr>
</tbody>
</table>
By using the CCDFs of the average global delay time increment, the fraction of stations \( (F_r) \) that exhibits a probability percentage delay increment \( (P_D) \) greater than or equal to 20% and 30% can be demonstrated in Table 5.13.

Table 5.13. PERCENTAGE DELAY INCREMENT FOR ALL EXAMINED SCENARIOS FOR DIFFERENT DIFFERENT HOP COUNT LIMITS.

<table>
<thead>
<tr>
<th>Hop Count ( (H_{CL}) )</th>
<th>Increment ( (P_D) ) ≥ 20%</th>
<th>Increment ( (P_D) ) ≥ 30%</th>
</tr>
</thead>
<tbody>
<tr>
<td>∞</td>
<td>33%</td>
<td>18%</td>
</tr>
<tr>
<td>7</td>
<td>28%</td>
<td>10%</td>
</tr>
<tr>
<td>6</td>
<td>20%</td>
<td>0%</td>
</tr>
<tr>
<td>5</td>
<td>0%</td>
<td>0%</td>
</tr>
</tbody>
</table>

The above results show that by using the newly introduced path selection rule based upon the \( AEF \) metric with different \( H_{CL} \) value significantly enhances the average global throughput of the network. This throughput improvement is associated with an increase in the delay time. Furthermore, assigning different values to the \( H_{CL} \) allows the delay time to be controlled by eliminating longer transmission paths and hence limiting the delay.

Figures H.1 and H.5 in Appendix H demonstrate the throughput improvement and the delay increment of the modified \( DSR \) routing algorithm against the standard \( DSR \) algorithm when the \( H_{CL} \) limit is set to ∞, 7, 6, and 5. Reducing the \( H_{CL} \) leads to a reduction in the global throughput and delay time, see Tables 5.11 and 5.12. This reduction in the percentage throughput improvement is caused by the lack of available paths between the source-destination pair. Reducing the \( H_{CL} \) value will bring about a higher level of congestion. Consequently, the global throughput of the network is reduced through more dropped
packets. However, reducing the $H_{cl}$ value will reduce the average path length which leads to reducing average delay time of the network. This modification allows the network operator to trade-off path throughput and end-to-end delay to meet network requirements.

5.3 Modified Version of $AEF$

An examination of the modified route selection path based upon the $AEF$ metric with different $DF$ values demonstrates that the smaller routed packets are taking longer paths compared to the larger packets in order to reach the gateway node, see Figure 5.22. This is because the routing algorithm based upon the $AEF$ metric routes the smaller packets away from the direct route to the gateway node (owing to the dependency of the $AEF$ metric on packet size). It will make the node which is forwarding small packet sizes appear to be more congested than it actually is. The modified path selection rule responds by routing packets away from the node, i.e. the routed packets will take longer transmission paths. This process could be considered a shortcoming of the modified path selection rule based upon the $AEF$ when the network carries voice traffic (owing to the small packet sizes usually associated with packetized speech). Long paths taken by the small packet streams might result in voice quality degradation owing to the increased delays incurred in their transmission. Ideally, all packets should be treated equally, irrespective of their size.

Figures 5.22 and 5.23 show the path lengths taken by the streamed packets of size 128 bytes and 1500 bytes. Based on these figures, the modified path selection rule based upon the $AEF$ metric penalizes the small packets over the larger ones as they take longer paths to reach their destination. It also can be observed from these figures, the difference in the hop count increases with the density factor. Owing to the dependency of the $AEF$ on the packet size, nodes with small packets appear more congested than they actually. On the other
hand, increasing the $DF$ value leads to increase the level of congestion. Therefore, the nodes with small packets appear highly congested and hence the modified algorithm will stream the small packets through longer paths. To deal with this penalization of streams comprising small packet sizes, a modified version of the $AEF$ ($ModAEF$) has been introduced.

Figure 5.22: Hop Count against Density Factor ($DF$) using the modified path selection rule based upon the $AEF$ metric for networks where $DF = 2$. 
In the figures above, the hop count is plotted against $DF$ for scenarios of $DF = 1, 2, 3, 4, 5,$ and 6. Each scenario comprises 1000 random topologies using a single gateway and 99 nodes randomly distributed across the network. The nodes are divided into two sets (50 and 49 nodes). One set of nodes generates packets of size 128 bytes, the other set generates packets of size 1500 bytes. The packet rate is set to fixed value 5 packets per second for all scenarios of this group. Figures 5.22 and 5.23 represent the hop count against the $DF$ and the PDF of the hop count respectively using $AEF$ metric. These figures show that the small packet streams (128 bytes) have incurred on average a greater hop count value to reach the gateway node than the larger packet streams (1500 bytes).
5.3.1 Simulation Results Obtained for ModAEF

This section introduces a modification to the DSR protocol by using a modified version of AEF (called ModAEF) in order to deal with the shortcoming arising from the dependence of the AEF on the packet size, see section 4.5.3. A further modification to the modified DSR routing algorithm has been introduced by employing the ModAEF metric as an alternative to the AEF metric. In this modified algorithm, the selected path is identified by choosing the path with the highest minimum ModAEF value.

A number of different simulation studies on the performance of the path selection rule based upon the ModAEF metric have been carried out and compared to the performance of the path selection rule based upon the AEF metric. A modification to the new path selection criterion is incorporated in this modification to achieve better results. The analysis focuses on the improvement in the average global throughput. The concomitant increase in the average delay was also analyzed. The OPNET modeler has been employed to investigate the performance of the modified DSR protocol on a series of randomly generated network topologies of different node densities.

Three different scenarios where DF is set to 2, 4, and 6, have been established using the ModAEF as a cost metric to investigate the performance of this metric against the AEF. In these scenarios, the generated packet sizes for 50 nodes in the network are set to 512 bytes and 49 nodes are set to 256 bytes, the packet rate is set to 5 packets per second, and the $\alpha$ parameter has been set to 1, see Equation 4.15. The CCDF of the global throughput improvement for the network scenarios of group $K-1$, see section 4.6.11.1, with densities $DF = 2, 4,$ and 6, for the modified DSR routing algorithm using the ModAEF metric against
the standard DSR are presented in Appendix I, see Figures I.1, I.2, and I.3 respectively in Appendix I. Using these CCDFs for this group of scenarios, the fraction of stations ($F_r$) that exhibit a probability of percentage throughput improvement ($P_T$) greater than or equal to 30% and 50% can be obtained and presented in Table 5.14.

TABLE 5.14 PERCENTAGE THROUGHPUT IMPROVEMENT FOR ALL EXAMINED SCENARIOS FOR DIFFERENT DF VALUES.

<table>
<thead>
<tr>
<th>Density Factor (DF)</th>
<th>$P_T$ [Improvement $\geq$ 30%]</th>
<th>$P_T$ [Improvement $\geq$ 50%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>64%</td>
<td>42%</td>
</tr>
<tr>
<td>4</td>
<td>53%</td>
<td>26%</td>
</tr>
<tr>
<td>6</td>
<td>44%</td>
<td>17%</td>
</tr>
</tbody>
</table>

The simulation results for the examined scenarios show that using the ModAEF as a metric for the modified routing mechanism exhibits a lower global throughput compared to the AEF metric. This verifies that the path selection rule based upon the AEF metric outperforms the path selection rule based upon the ModAEF in term of global throughput. On the other hand, the path selection rule based upon the ModAEF exhibits a reduced average delay time compared to the path selection rule based upon the AEF, see Figures I.7, I.8, and I.9 in Appendix I. These figures also show that the routing algorithm based upon the ModAEF shows a significant improvement in terms of the global throughput compared to the standard DSR routing algorithm. This improvement in the throughput is associated with an increase in the average delay time.
The simulation results of the average delay time for the scenarios where $DF = 2, 4,$ and $6$, for the modified $DSR$ using the $ModAEF$ against the standard $DSR$ have been plotted in the format of the CCDF, see I.17, I.18, and I.19 in Appendix I. Based on the CCDF for these simulations, the fraction of stations ($F_r$) that exhibits a probability percentage delay increment ($P_D$) greater than 20% and 30% can be demonstrated in Table 5.15.

<table>
<thead>
<tr>
<th>Density Factor ($DF$)</th>
<th>$P_D[Increment \geq 20%]$</th>
<th>$P_D[Increment \geq 30%]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>27%</td>
<td>12%</td>
</tr>
<tr>
<td>4</td>
<td>43%</td>
<td>24%</td>
</tr>
<tr>
<td>6</td>
<td>44%</td>
<td>26%</td>
</tr>
</tbody>
</table>

The simulation results for this group of scenarios show that the path selection rule based upon the $ModAEF$ metric outperforms the path selection rule based upon the $AEF$ metric in terms of average delay time. This improved delay performance is accompanied by a reduction in the global throughput of the network. Figures 5.24 and 5.25 demonstrate that the path selection rule based upon the $ModAEF$ routes the large packet streams away from the direct paths to the gateway node. On the other hand it tends to route the small packets through the direct paths to the gateway node. The use of the $AEF$ as a congestion metric is not ideal owing to its dependence of packet size. The $ModAEF$ metric attempts to correct for this dependency. The $AEF$ metric provides an indirect measure of the contention experienced at a node. In order to remove the dependence on the packet size, the $AEF$ ought to be replaced with the contention level experienced at a node. Direct measure to the local contention at a node provides a measure to the availability of transmission opportunities. In
other words, the number of the available transmission opportunities of a node is limited by the level of contention which is in turn determined by the number of other stations operating in the vicinity of the station also contending for access. Essentially the $\alpha$ factor serves to reduce the penalization of small packets by artificially allowing more small packets to take more direct paths to the gateway, i.e. usually more congested routes, which results in greater packet loss and hence a reduced global throughput. Without the ModAEF, these small packets would normally be directed away from the congested regions. The decrease in the delay time corresponds to more direct paths to the gateway and usually a more direct route which reduces the delay.

Following a similar scenario setup of section 4.6.11.2, the nodes in the network are classified into two sets one with 50 nodes and the other with 49 nodes. The generated packet size is set to 128 bytes in one of the sets and 1500 bytes in the other one. The packet rate is set to 5 packets per second in all scenarios of this group. Six scenarios with $DF = 1, 2, 3, 4, 5, \text{ and } 6$ value have been established using the ModAEF metric in the routing discovery mechanism. Based on these simulations, the relationship between the $DF$ of the network and the hop count taken by the routed packet streams can be plotted, see Figure 5.24. While, Figure 5.25 demonstrates the PDF of the hop count for these scenarios.
Figure 5.24: Hop Count ($H_c$) against Density Factor ($DF$) using $ModAEF$ metric for networks where $DF = 2$.

Figure 5.25: PDF of the Hop Count ($H_c$) using $ModAEF$ metric for networks where $DF = 2$. 

174
A comparison has been made between the path selection rule based upon the \textit{AEF} and the path selection rule based upon the \textit{ModAEF} in terms of the path lengths of the streamed packets is shown in Figure 5.24. In this figure, the average number of hops taken by the small routed packets when the \textit{AEF} metric is used is higher than the number of hops taken by the large packets. This figure also demonstrates that the average number of hops taken by the large packets when the \textit{ModAEF} metric is employed is greater than the number of hops taken by the small packets to reach the gateways node.

\subsection*{5.3.2 Examination of the Effect of Packet Size Variation}

In this section, an investigation of the packet size effects on the path lengths of the packet streams has been carried out. Six scenarios with different $\alpha$ value have been established using the \textit{ModAEF} metric. The nodes in each topology have been divided into two sets, one with 50 nodes and the other with 49 nodes. The packet size is set to 128 bytes for one of these sets and 1500 bytes for the other set. The packet rate is set for 5 packets per second for all topologies. Figures 5.26 and 5.27 present the hop count value against the $\alpha$ factor when the \textit{ModAEF} metric is employed for scenarios with $\alpha = 0.2, 0.4, 0.6, 0.8, 1.5, 2$. This examination has been applied to moderate and dense networks (where $DF = 2$ and 4 respectively) in order to investigate the effect of the $DF$ in relation to the packet size variations (different $\alpha$ values).

The objective of this investigation is to examine the effect of the packet size on the path length of the routed packets. Based on Equation 4.14, as the packet size is increased the \textit{ModAEF} value is reduced and hence the longer the path that will be taken by the packet. The figures below illustrate $\alpha$ variation affect on the path length of the packet streams in
terms of hop count. It can be seen that increasing the value of the $\alpha$ factor results in an increase in the path lengths for the large sized packet streams. Increasing the $\alpha$ value has the opposite effect on the small routed packets. This shows that the routing algorithm based upon the $ModAEF$ metric streams the large packets away from the direct to the gateway node, whereas the small packets have taken shorter paths. Figures 5.26 and 5.27 show that varying the packet size to a value (by adjusting the $\alpha$ parameter value) affects the path lengths taken by the routed packets. In Figure 5.26, for these particular packet sizes, when the $\alpha$ value is set to less than 0.26 (the intersection point) the routing algorithm streams the packets of size 128 bytes away from the direct path and the packets of size 1500 bytes will be directed towards the gateway node. Consequently, the small routed packets will take longer paths than the large ones. While the intersection point for the $DF = 4$ appears at a higher value of $\alpha = 0.38$. Tuning the $\alpha$ parameter allows the network operator to control the traffic in the network. For example, if video streams are dominant in the network, the network operator can tune $\alpha$ in order to give priority to the large packet streams over smaller ones.
Figure 5.26: Hop Count ($H_c$) against $\alpha$ factor for scenarios where $DF = 2$.

Figure 5.27: Hop Count ($H_c$) against $\alpha$ factor for scenarios where $DF = 4$. 
5.4 Dynamic Behaviour of the New Metric

A number of preliminary investigations were made regarding the settling time of the DSR routing protocol based upon the new AEF metric. Figure 5.28 demonstrates the settling time for the algorithm for network topologies with one gateway and $DF = 1$, one gateway and $DF = 2$, two gateways and $DF = 2$, and three gateways with $DF = 2$. The packet size is set to 512 bytes for all analyzed topologies and the packet rate is set to 5 packets per second. It can be seen here that the modified DSR routing algorithm takes a significant amount of time (approximately 180 seconds) to attain a steady-state condition, i.e. for the algorithm to converge to a set of stable transmission paths. This is due to the reactive nature of this algorithm where it continuously reacts to changes in the network conditions including those changes resulting from its own routing decisions.

However, the length of the settling time depends on several factors including the topology, the nature of the load and the initial conditions. For this particular example, the effect of the network density on the settling time of the system has been performed using the same initial conditions. It has been shown that the lower the $DF$ value the shorter the settling time is. This is due to the contention effect, reducing the network density results in reducing the level of congestion in the network which leads to reduced congestion across the network. This will lead to a faster convergence to a set of stable routes. Varying the number of gateways in the network has been also investigated. Increasing the number of gateways will reduce the level of congestion in the network which results in more stable routes emerging. Consequently, the system will converge faster to a set of stable routes. As a result, the settling time of the routing mechanism will be reduced.
Due to the limitation of the current version of OPNET, it was not possible to make any step changes to simulation parameters during the simulation run, i.e. it is not possible to switch a node off or to change its transmit power etc. while the simulation is running. Therefore, the only investigation into the dynamic behaviour of the AEF path selection rule was performed by moving one of the nodes (which is located two hops away from the gateway) from its original position towards the boundary of the network.
Figure 5.29 demonstrates the settling time for the algorithm when the node moves away for the same network topology with one gateway and $DF = 2$. In this network, the node moved out of range of the network (and therefore was essentially removed from the network) within 10 seconds. This figure shows that the throughput settled down within about 5 seconds which indicates that the settling time can be fast enough to react to changes in the network topology.

![Figure 5.29: Normalised throughput at the gateway node against the time interval for network of one gateway and $DF = 2$ when one node is removed from the network.](image)

### 5.5 Performance Evaluation of the AEF metric against the ETT and MP

A performance evaluation of the modified DSR routing algorithm based upon the AEF metric against the DSR routing algorithm using the ETT in one instance as a cost metric and
using the $MP$ metric in other are introduced in this section. To carry out this evaluation, scenarios of $DF = 2, 4, \text{ and } 6$ for one gateway node has been formed. The simulation results of the $DSR$ based upon the $ETT$ metric have been compared to the simulation results of the modified $DSR$ based upon the $AEF$ metric. Also the simulation results of the $DSR$ based upon the $MP$ metric have been compared to the simulation results of the modified $DSR$ based upon the $AEF$ metric. Scenarios of 1000 random topologies have been established with one receiver (i.e. one gateway node) and 99 senders randomly distributed across the network. The employed packet sizes in this scenario are set to 512 bytes and the rate is set to 5 packets per second. The nodes in the network have been divided into four sets of nodes one of which consist of 25 nodes with a transmission rate of 11 Mbps, the second set consists of 25 nodes with a transmission rate of 5.5 Mbps, the third set consists of 25 nodes with a transmission rate of 2 Mbps, and the fourth set consists of 24 nodes with a transmission rate of 1 Mbps. The simulation was run four times for each topology: first with standard $DSR$, followed by the modified $DSR$ using the $AEF$ as a cost metric, then with $DSR$ using $ETT$ as the cost metric, finally with $DSR$ using $MP$ as the cost metric. The global throughput was recorded for each 10 minute simulation run in order to calculate the percentage improvement for the particular topology. The CCDF of the global throughput improvement and the average delay time increment for the modified $DSR$ routing algorithm based upon the $AEF$ metric against the standard $DSR$ for all network topologies examined have been calculated. Similarly, the CCDF of the global throughput improvement and the average delay time increment for the $DSR$ using the $ETT$ as cost metric against the standard $DSR$ have been also calculated for all examined network topologies. The simulation results of the global throughput improvement for this scenario are plotted in the format of CCDFs and presented in Figures J.1, J.2, and J.3 in Appendix J. Also, the CCDF of the global
throughput improvement and the average delay time increment for the DSR using the MP as cost metric against the standard DSR have been also calculated for all examined network topologies. The simulation results of the global throughput improvement for this scenario are plotted in the format of CCDFs and presented in Figures K1, K.2, and K.3 in Appendix K.

The CCDFs of these scenarios can be examined to determine the fraction of stations ($F_r$) that exhibit a probability of percentage throughput improvement ($P_T$) greater than or equal to 30% and 50% can be obtained and presented in Table 5.16.

<table>
<thead>
<tr>
<th>DF</th>
<th>$P_T \geq 30%$</th>
<th>$P_T \geq 50%$</th>
<th>$P_T \geq 30%$</th>
<th>$P_T \geq 50%$</th>
<th>$P_T \geq 30%$</th>
<th>$P_T \geq 50%$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>72%</td>
<td>49%</td>
<td>47%</td>
<td>17%</td>
<td>63%</td>
<td>27%</td>
</tr>
<tr>
<td>4</td>
<td>66%</td>
<td>37%</td>
<td>36%</td>
<td>10%</td>
<td>50%</td>
<td>8%</td>
</tr>
<tr>
<td>6</td>
<td>60%</td>
<td>30%</td>
<td>28%</td>
<td>0%</td>
<td>40%</td>
<td>5%</td>
</tr>
</tbody>
</table>

An examination of the results show that the path selection rule based upon the AEF metric outperforms the path selection rule based upon the ETT and MP metrics in terms of the global throughput. On the other hand, this newly introduced path selection rule shows a higher delay time compared to the path selection rule based upon the ETT. The CCDFs of the average delay time for the scenarios of this group have been shown in Figures J.4, J.5, and J.6 in Appendix J. At the same time, the new path selection rule based upon the AEF metric exhibits almost the same delay increment against the standard DSR routing.
algorithm as the path selection rule based upon the $MP$ metric does. The CCDFs of the average delay time for the scenarios of the $MP$ metric have been shown in Figures K.4, K.5, and K.6 in Appendix $K$. Based on these CCDFs, the fraction of stations ($F_r$) that exhibits a probability percentage delay increment ($P_D$) greater than or equal to 20% and 30% can be demonstrated in Table 5.17.

<table>
<thead>
<tr>
<th>$DF$</th>
<th>$P_D \geq 20%$</th>
<th>$P_D \geq 30%$</th>
<th>$P_D \geq 20%$</th>
<th>$P_D \geq 30%$</th>
<th>$P_D \geq 20%$</th>
<th>$P_D \geq 30%$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>29%</td>
<td>10%</td>
<td>21%</td>
<td>0%</td>
<td>32%</td>
<td>2%</td>
</tr>
<tr>
<td>4</td>
<td>51%</td>
<td>31%</td>
<td>35%</td>
<td>18%</td>
<td>52%</td>
<td>34%</td>
</tr>
<tr>
<td>6</td>
<td>54%</td>
<td>32%</td>
<td>38%</td>
<td>19%</td>
<td>57%</td>
<td>30%</td>
</tr>
</tbody>
</table>

The newly introduced path selection rule based on the $AEF$ metric has shown a significant improvement in the global throughput of the network compared to the path selection rule based upon the $ETT$ and $MP$ metrics. This enhancement is accompanied by an increased delay time. Unlike the path selection rule based upon the $ETT$ and $MP$ metrics, the path selection rule based upon the $AEF$ metric takes into account the interference affect which has a large influence on the performance network, i.e. it avoids routing through heavily congested nodes. In addition, avoiding congestion regions results in longer transmission paths and hence the end-to-end delay is increased. Based on these simulation results, it can be shown that the new $AEF$ metric is a viable alternative routing metric to more traditional link quality based metrics.
5.6 Performance Comparison of the Routing Metrics Examined in the Thesis

The basic strategy for conducting the simulation study is to compare three routing metrics the $H_c$, $ETT$, $MP$, and $AEF$ within the $DSR$ routing protocol in WMN environments. From the results of the simulation tests, it can be seen that the $AEF$ outperforms the $H_c$ metric in terms of network throughput. This is because the $H_c$ metric only concerns itself with finding the shortest path between the source and the destination nodes regardless of how efficient the route is. It does not account for other factors that have a critical affect on the performance of the network, such as congestion, packet loss, and bandwidth availability. In other words, using this metric can lead to data packets being routed through highly congested routes which can lead to high packet loss.

The simulation examinations also verify that the $AEF$ metric is more effective than $ETT$. It outperforms the $ETT$ metric when it has been implemented in the $DSR$ protocol in WMNs. The $ETT$ metric is a link aware metric that finds a path based on the probability of successful packet delivery and bandwidth of each link. There are several drawbacks associated with the $ETT$ metric which cause performance degradation. It has no explicit consideration of the interference in the network which is a critical issue for the network performance. Due to the dependency of the $ETT$ on the loss probability, the probe packets may not experience the same loss rate as data packets since they are small and sent at lowest possible data rate (1 Mbps in case of IEEE 802.11b). Furthermore, the metric has no direct consideration of the link load or data rate. Two links with different data rates may have the same loss rate [87]. Moreover, the $ETT$ is a link quality metric operates by finding a route with the lowest sum of the link $ETTs$ along the path to the destination. This means that a route with the worst bottleneck (a highly congested link) might be chosen by the
routing mechanism which could lead to a dramatic reduction in the overall network performance. However, the main drawback of the ETT metric is that it does not account for local congestion at a node.

Based on the simulation analysis, the new AEF metric also outperforms the MP metric in terms of global throughput and average delay time when it has been implemented in the DSR protocol in WMNs. The MP metric takes into account the available bandwidth (AB) as well as the number of retransmissions (NR) to improve the WMN performance. The MP metric was introduced as a congestion measure for the WMNs. Measuring the number of packets that arrive at the node and number of packets transmitted by the node within a unit time is required to accurately measure the congestion locally at a node. The MP metric fails to account for both these parameters. Instead, it utilises the AB and NR as a measure for node congestion. The number of retransmission attempts can be used as an indication of link quality but generally does not give a reliable indication of the node congestion. However, excessive number of retransmission attempts may lead to node congestion, but this will depend on the number of packets entering and leaving the node within a unit time. The main shortcoming of this metric is that it takes no explicit consideration of the local congestion at a node as it does not directly takes into account the contention experienced by the node. In other words, the MP metric take no consideration to the number of available transmission opportunities at a node. The number of the available transmission opportunities of a node is limited by the level of contention which is in turn determined by the number of other stations operating in the vicinity of the station also contending for access. Measurements of the number of available transmission opportunities at a node and the forwarded traffic received at the node within a unit of time are required to determine the
probability of congestion at a node. The $AEF$ metric explicitly considers the local congestion at a node. Besides, it chooses the path with the least worst bottleneck, i.e. it finds a path capable of supporting the highest throughput. Based on these simulation results, it has been demonstrated that the congestion avoidance strategy is more effective than link quality optimization in finding high throughput paths in WMN environments.

5.7 Summary
The operation of the modified $DSR$ path selection rule has been explained in this chapter by demonstrating the basic operation of forwarding packets at a node and the process of congestion avoidance is also demonstrated. The affect of varying various network parameters such as network density, packet rate, packet size, traffic type, and number of gateway nodes, on the performance of the modified path selection rule has also been investigated in this chapter. The main shortcoming of the $AEF$ based path selection rule is the increased delay time due to the congestion avoidance mechanism which results in longer transmission paths being taken. To overcome this shortcoming, a hop count limit is incorporated into the routing algorithm to eliminate long transmission paths in order to allow the network manager to trade-off throughput against delay.

Due to the dependency of the $AEF$ on the packet size, the smaller routed packets take longer transmission paths compared to the large ones. A modified version of the $AEF$ metric ($ModAEF$) is presented in this work to correct for the dependency of the $AEF$ on the packet size. This could be considered to be a major shortcoming associated with this metric when voice applications are used. The $ModAEF$ is employed by the modified $DSR$ path selection rule as an alternative metric to the $AEF$ to remedy this shortcoming. The
performance evaluation of the modified path selection rule based upon the $ModAEF$ against the standard $DSR$ is demonstrated in this chapter. A performance comparison of the $ModAEF$ metric against the $AEF$ metric is also introduced in this chapter. Employing the $ModAEF$ as a cost metric for the modified path selection rule exhibits a significant improvement in the throughput compared to the standard $DSR$. In comparison to the $AEF$, the $ModAEF$ showed a reduction in the global throughput and delay time of the network. However, the newly introduced path selection rule using the $AEF$ as a cost metric performs effectively in WMNs. It is concerned with finding paths between the source and the destination nodes that can avoid the congested regions in the network. Congestion avoidance leads to an overall improvement in the network performance in terms of throughput. This congestion avoidance algorithm based on the $AEF$ metric outperforms the standard hop count, the well known $ETT$, and the $MP$ metrics within the $DSR$ routing protocol in WMN environment.
A routing algorithm that takes into account the variability of the wireless link quality is required to be introduced to address some characteristics of the wireless mesh networks such as the relatively stationary topologies and shared wireless medium, since the hop count metric is not aware of the nature of the wireless link. The shortest path metric is concerned with finding a path between source-destination pair regardless of how efficient the path is. As it is not aware of the nature of the wireless link, a link of low quality could be chosen resulting in degradation in the performance of the network. A cross-layer technique should be employed for routing to consider factors such as interference, bandwidth availability, etc., from various layers allowing information exchanged between protocol layers, to help in finding reliable and efficient paths to enhance the performance of the network.

Due to the shared nature of the wireless medium, a wireless link in a mesh network does not have a dedicated bandwidth since nodes in the vicinity may also contend for the same bandwidth. Therefore, an effective routing metric must be able to capture the contention for access to the medium between competing flows. The DSR protocol has been modified to make it better suited to the WMN environment. In this modification, a metric \((A EF)\) that reflects the level of contention experienced locally at a node is incorporated into the route
discovery mechanism. Since the nodes in the network contend for access to the wireless medium using the IEEE 802.11 DCF MAC mechanism, a high level of contention for access to the medium will result in a low availability of bandwidth at a node. The $AEF$ which is an indicator to the level of congestion at a node has been introduced as an alternative metric to the hop count for the routing selection mechanism. In this modification, the selected path is identified by finding a path with the highest minimum $AEF$ value. The modified $DSR$ routing mechanism is based upon avoiding congested nodes where packet loss is likely to occur. The objective of this work is to utilize locally generated MAC layer information at the routing layer to improve the global performance of the network.

The OPNET modeler has been employed to examine a series of randomly generated network topologies which are classified under different types of scenarios. In these scenarios, the performance of the path selection rule based upon the $AEF$ metric has been examined against the standard path selection rule of the standard $DSR$ protocol under various node densities, packet rates, packet sizes, traffic types, and number of gateway nodes. In this work, 1000 topologies for each scenario with one gateway and 99 nodes randomly distributed across the network have been generated. Each topology was simulated twice over a 10 minute interval for each run. One simulation used the original $DSR$ routing algorithm while the other employed the modified $DSR$ routing algorithm. The average throughput and delay time were recorded for each run and the percentage throughput improvement and delay increment for the particular topology were calculated.
Through computer simulation using the OPNET modeler, it has been demonstrated that significant enhancement in throughput can be achieved through the use of this modified DSR routing algorithm. For example, for topologies of a moderate network density such as \( DF = 2 \), it has been shown that about 56% of the network nodes exhibit a probability of percentage throughput improvement greater than or equal to 50%, and about 77% of the stations exhibit a probability of percentage throughput improvement greater than or equal to 30%, see Table 5.2. However, this improvement in the throughput is also accompanied by an increase in the delay time. As an example of this, the delay time exhibited by the same network topologies mentioned above, about 18% of the stations in the network present a probability of percentage delay increment greater than or equal to 30%, and about 33% of the nodes exhibit a probability of percentage delay increment greater than or equal to 20%, see Table 5.3. The increase in the delay time could be considered a shortcoming of this approach under some circumstances, such as if the network were carrying voice traffic which might lead to the voice quality degradation.

To overcome the drawback of this approach, a hop count limit has been introduced into the path selection rule. The use of a hop count limit allows the network administrator to control the delay time of the network by imposing an upper limit on the length of the selected transmission paths. The hop count limit can be tuned (in order to impose a maximum permissible network delay) to satisfy the network requirements. Different scenarios have been established in this work for this purpose of assessing the performance of the modified path selection rule with different hop count limit. The analysis showed that tuning the hop count limit to a lower value reduces the global throughput and delay time of the network. The throughput reduction is due to a reduction in the available transmission paths between
the source-destination pair and hence increased contention for access to the medium. However, incorporating the hop count limit allows the network operator to trade-off throughput against delay.

An analysis applied to the performance to the path selection algorithm based upon the $AEF$ metric has highlighted another shortcoming associated with this algorithm. Adopting the $AEF$ metric as a local congestion metric at a network node is not ideal owing to its dependency on the packet size. This has the unfortunate consequence that small packets tend to take longer paths towards the gateway node compared with large packet sizes. This could be considered as a drawback of the metric when the network is carrying voice services. This has been corrected by developing a modified version of the $AEF$ metric (called $ModAEF$) that explicitly considers the size of the packet. A tuning factor ($\alpha$) has also been introduced to allow the operator determine the level of the weighting that should be applied to the packet size to correct for this dependence. Based on the results of this analysis, the $ModAEF$ streams the large packets away from the direct paths to the gateway while it streams the smallest packets along more direct paths to the gateway node. The routing selection mechanism identifies the best path by selecting the path with the highest minimum $ModAEF$ value.

A number of different simulation studies have been performed to analyse the behaviour and performance of the modified $DSR$ routing algorithm using the $ModAEF$ as the cost metric when compared to the standard $DSR$ routing algorithm. Based on these analyses, it has been shown that the $AEF$ metric outperforms the $ModAEF$ in terms of throughput. On other hand, the $ModAEF$ metric exhibits less average delay time than $AEF$ metric. However,
employing the ModAEF as cost metric exhibits significant improvement in the throughput compared to the standard DSR. For example, when the network topologies of $DF = 2$ were examined using the ModAEF metric, about 42% of stations exhibit a probability of percentage throughput improvement greater than or equal to 50%, and about 64% of the stations exhibit a probability of percentage throughput improvement greater than or equal to 30%, see Table 5.14. For this particular example, by implementing the ModAEF in the routing algorithm it exhibits a 25% reduction in the global throughput improvement compared to the use of the AEF. On the other hand, the ModAEF outperforms the AEF metric in terms of delay time by exhibiting a 33% reduction in the delay time. About 12% of the stations exhibit a probability of percentage delay increment greater than or equal to 30%, and about 27% of the nodes exhibit a probability of percentage delay increment greater than or equal to 20%, see Table 5.15.

Finally, the performance of the modified DSR routing algorithm based upon the AEF metric has been evaluated against the DSR routing algorithm based upon the ETT metric and the MP metric using mesh nodes with different transmission rates. The modified version of the DSR protocol outperforms the DSR protocol using the ETT and MP as a cost routing metric in terms of the global throughput improvement. The overall performance of the network can be significantly improved by implementing the AEF metric in the route selection mechanism.

The objective of this work is to develop a new routing metric that explicitly takes into account the local availability of the bandwidth at a node. This new metric is also provides a measure of the local contention for access at a node. The main contribution of this work is
in adopting a local congestion metric that explicitly account for the congestion experienced locally at a network node. In this work, a new cross-layer routing metric and path selection rule for WMNs is introduced that explicitly considers the local availability of bandwidth at a node. It demonstrated how this cross-layer approach to routing can lead to a significant improvement in WMN performance through reduced node congestion. It also introduced a viable alternative routing metric to more traditional link quality based metrics. Identifying the critical role played by the access contention in determining routing protocol performance is another contribution of this work. In addition to this, it highlighted the dependence of network capacity on packet size and shows how this can be managed within the new \textit{AEF} metric.

The new path selection mechanism is based upon avoiding congested nodes where packet loss is likely to occur and which will result in a reduced throughput. It exhibits better load distribution across the network due to avoiding routing through congested nodes and hence significantly maximizes the global throughput of the network. It has been demonstrated that the modified routing algorithm based on the \textit{AEF} outperforms the standard \textit{Hc}, the well known \textit{ETT}, and \textit{MP} metrics within the \textit{DSR} routing protocol in WMNs. On the other hand, due to the dependency of the \textit{AEF} metric on the packet size, this metric cannot be considered an ideal congestion metric. Ideally, the \textit{AEF} needs to be replaced with a metric that only reflects the congestion experienced at a node. This dependency on packet size is not necessarily a drawback of the new \textit{AEF} metric. Since, from a network perspective, the capacity of the network will dependence of the size of the packets being transmitted on the network where the greater the packet size, the greater the capacity, i.e. the maximum global throughput of the network. This dependence on packet size is also shared by the \textit{AEF}
metric, so in a sense the $AEF$ also captures this dependence which can lead to improved routing decisions. In fact, by implementing the $\alpha$ tuning factor in the modified $AEF$ metric, this dependence can be controlled and this can lead to optimized network performance.

6.1 Conclusions

The main findings from the simulations carried out in this work can be summarised as follows:

- The $DSR$ routing mechanism based on the $AEF$ shows a significant improvement over the $H_{c}$, $ETT$, and $MP$ metrics in terms of throughput due to the explicit consideration of the congestion experienced locally at a node.

- Two shortcomings arise due to the use of such a metric in the routing mechanism of the $DSR$ protocol, the end-to-end delay increment and the penalization of small packets over the large packets in the routing decisions.

- The increase in the end-to-end delay is related to the congestion avoidance strategy of the modified $DSR$ routing mechanism which results in routing packets being routed along long transmission paths in order to avoid congestion.

- To overcome this drawback, a further modification to the modified $DSR$ routing mechanism has been adopted. This is achieved by incorporating a hop count limit in the routing mechanism of the $DSR$ protocol to impose an upper limit on the length of the transmission paths. Utilizing the hop count limit allows the network manager to trade-off path throughput and end-to-end delay.

- The other shortcoming of the modified $DSR$ is related to the penalization of the small routed packets arising from the dependency of the $AEF$ on the packet size.
To overcome this penalization of small packets, a modified version of the \( AEF \) (\( ModAEF \)) metric has been utilized as an alternative to the \( AEF \) metric. Using the \( AEF \) as a congestion metric is not ideal owing to its dependence of the packet size. Ideally, the \( AEF \) needs to be replaced with a metric that reflects the level of contention only experienced at a node. However, the \( ModAEF \) has been introduced to correct for this dependency.

A tuning factor (\( \alpha \)) has also been introduced as a tuning parameter for the \( ModAEF \) metric, to allow the network manager to determine the weighting that should be applied to the packet size to correct for this dependence. Utilizing the \( \alpha \) factor reduces the penalization of small packets by artificially allowing more small packets to be streamed along more direct routes towards the gateway node.

Based on the analysis presented in this work, it has been verified that the \( AEF \) is a simple and effective routing metric that can be utilised in WMN environments.

### 6.2 Suggestions for Future Work

In addition to the comparison carried out in this work between three routing metrics within a WMN environment when using the \( DSR \) routing protocol – namely \( H_c \), \( ETT \), and the \( AEF \) based path selection rule, some other research issues have been identified that could be addressed to further investigate the reliability and effectiveness of the \( AEF \) based path selection rule, as follows:

- Further investigations of the dynamic behaviour of the \( DSR \) routing protocol when using the \( AEF \) metric is required. However, a number of preliminary investigations were made
into the settling time of the new routing protocol, see Figures 5.28 and 5.29. It can be seen here that the modified DSR routing algorithm takes a significant amount of time (approximately 180 seconds) to attain a steady-state condition, i.e. for the algorithm to converge to a set of stable transmission paths. This is due to the reactive nature of this algorithm where it continuously reacts to changes in the network conditions including those changes resulting from its own routing decisions. However, the length of the settling time depends on several factors including the topology, the nature of the load and the initial conditions. Based on these investigations, the settling time of the throughput of the network can be fast enough to react to changes in the network topology, see Figure 5.29.

- Network traffic routing plays a critical role in determining the performance of a WMN. The performance of the modified DSR protocol using the AEF metric could be further investigated under real world traffic with a wide range of packet sizes, rates, and types, in order to examine the method under more realistic traffic patterns.

- Investigate the use of transmit power control (TPC) at the network nodes to maximise the number of the gateway neighbour nodes. The basic idea behind using the TPC is to mitigate the impact of interference [213]. To gain a better understanding to the behaviour of the modified DSR routing protocol using the AEF metric, it will be beneficial to implement an algorithm that can identify the gateway neighbour nodes and then use the power control to trade-off connectivity against congestion avoidance to improve the performance of the WMNs.

- Unlike the standard DCF mechanism, the IEEE 802.11e EDCA mechanism introduces unfairness into the system, by allowing certain nodes win more transmission opportunities
than other nodes. The EDCA mechanism could be employed to prioritise the congested nodes over noncongested neighbours, thereby allowing them to win more transmission opportunities in order to reduce the level of congestion. The challenge here is to tune the EDCA parameters to ensure that effective prioritisation occurs.

- Each simulator has its own strengths and weaknesses. For example, a comparison has been made by Lucio et al between OPNET modeler and NS2 in terms of accuracy of bandwidth estimation for the pure CBR-type traffic. They have shown that, NS2 performed better than OPNET modeler using the default modeler package [214]. It will be useful to validate the effectiveness of the modified path selection rule based on the AEF metric with other simulators such as the widely used NS2.

- Future work should provide an experimental validation of a hardware test-bed using the newly introduced AEF routing metric for the DSR protocol. The purpose of using a simulator is to provide proof of concept. It is likely that the performance gains presented in this work will be less in an experimental hardware test-bed due the basic assumptions regarding the channel model and surrounding environment.

- The performance of the modified path selection rule based upon the AEF metric could be examined with a more realistic channel model, by implemented a more sophisticated channel model that considers some propagation effects such as fading, shadowing, and path attenuation in order to prove its effectiveness. Taking into account such parameters is to apply a more realistic examination, in which it is likely that the gained performance in this work will be less when such parameters have been included.
It is worthwhile to note that the main problem of the AEF metric is the dependence on the packet size which leads to penalize small routed packets over the large packets, by routing the smaller packets away from the direct paths to the gateway node, i.e. smaller packet sizes will be treated unfairly. Accordingly, the AEF is not the ideal metric to be used as a congestion metric as it provides an indirect measure of the contention experienced at a node. Ideally, the AEF ought to be replaced with a metric that only considers the contention level experienced at a node. A direct measure to the local contention at a node provides a measure of the availability of transmission opportunities. In other words, the number of the available transmission opportunities of a node is limited by the level of contention which is in turn determined by the number of other stations operating in the vicinity of the station also contending for access.

Owing to the throughput dependency on the transmission rate, it may be worth considering implementing a modification to the line rate adaptation algorithm to explicitly consider congestion. Nodes communicating with different transmission rates causes throughput degradation because nodes with higher line rates have to wait longer for nodes with lower transmission rates to complete their transmissions.
Publications


References


international conference on Computer communication. vol. 1, pp. 36-49, India (Mumbai), 2002.


Available at:
http://www.scansourcesecurity.com/MicroSites/ScanSourceSecurity/quickstart/docs/Firetide

Available at: www.gsd.inesc-id.pt/~ler/reports/ricardopinto-midterm.pdf (Last accessed: 10/06/2011)


Proceeding of the 2nd IEEE Workshop on Mobile Computing Systems and Applications,

for Mobile Ad Hoc Networks,” Internet Draft, Internet Engineering Task Force, 15 April
(Last accessed: 10/06/2011).


Available at: http://portal.acm.org/citation.cfm?id=352164  (Last accessed: 10/06/2011).


the degree of master of computer science. Faculty of computer science and information technology University of Malaya, Kuala Lumpur, May 2009.


Available at: www4.ncsu.edu/~mlsichit/Research/Publications/wwcChallenges.pdf
(Last accessed: 10/06/2011).


Available at: http://www.ietf.org/rfc/rfc3626.txt (Last accessed: 10/06/2011)


Available at: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.20.5753 (Last accessed: 10/06/2011)


Appendix

Appendix A

Figure A.1: PDF of the percentage throughput improvement for $DF=1$ scenario [$Pz = 512$ B, $Pr = 5$ pps].
Figure A.2: PDF of the percentage throughput improvement for $DF = 1.5$ scenario [$P_x = 512$ B, $Pr = 5$ pps].

Figure A.3: PDF of the percentage throughput improvement for $DF = 2$ scenario [$P_x = 512$ B, $Pr = 5$ pps].
Figure A.4: PDF of the percentage throughput improvement for $DF = 2.5$ scenario [$P_z = 512$ B, $Pr = 5$ pps].

Figure A.5: PDF of the percentage throughput improvement for $DF = 3$ scenario [$P_z = 512$ B, $Pr = 5$ pps].
Figure A.6: PDF of the percentage throughput improvement for $DF = 4$ scenario [$P_z = 512$ B, $P_r = 5$ pps].

Figure A.7: PDF of the percentage throughput improvement for $DF = 5$ scenario [$P_z = 512$ B, $P_r = 5$ pps].
Figure A.8: PDF of the percentage throughput improvement for $DF = 6$ scenario [$P_z = 512$ B, $Pr = 5$ pps].

Figure A.9: PDF of the percentage throughput improvement for $DF = 8$ scenario [$P_z = 512$ B, $Pr = 5$ pps].
Figure A.10: PDF of the percentage throughput improvement for $DF = 10$ scenario [$P_z = 512$ B, $Pr = 5$ pps].

Figure A.11: PDF of the percentage delay increment for $DF = 1$ scenario [$P_z = 512$ B, $Pr = 5$ pps].
Figure A.12: PDF of the percentage delay increment for $DF = 1.5$ scenario [$P_z = 512$ B, $P_r = 5$ pps].

Figure A.13: PDF of the percentage delay increment for $DF = 2$ scenario [$P_z = 512$ B, $P_r = 5$ pps].
Figure A.14: PDF of the percentage delay increment for $DF = 2.5$ scenario [$P_z = 512$ B, $Pr = 5$ pps].

Figure A.15: PDF of the percentage delay increment for $DF = 3$ scenario [$P_z = 512$ B, $Pr = 5$ pps].
Figure A.16: PDF of the percentage delay increment for $DF = 4$ scenario [$P_z = 512$ B, $Pr = 5$ pps].

Figure A.17: PDF of the percentage delay increment for $DF = 5$ scenario [$P_z = 512$ B, $Pr = 5$ pps].
Figure A.18: PDF of the percentage delay increment for $DF = 6$ scenario [$P_z = 512$ B, $Pr = 5$ pps].

Figure A.19: PDF of the percentage delay increment for $DF = 8$ scenario [$P_z = 512$ B, $Pr = 5$ pps].
Figure A.20: PDF of the percentage delay increment for $DF = 10$ scenario [$P_z = 512$ B, $P_r = 5$ pps].
Appendix B-1

Figure B.1.1: CCDF of the percentage throughput improvement for $DF = 1, 1.5, 2, 2.5, \text{ and } 3$ scenarios [$P_z = 512 \text{ B}, P_r = 2.5 \text{ pps}$].

Figure B.1.2: CCDF of the percentage throughput improvement for $DF = 4, 5, 6, 8, \text{ and } 10$ scenarios [$P_z = 512 \text{ B}, P_r = 2.5 \text{ pps}$].
Figure B-1.3: PDF of the percentage throughput improvement for $DF = 1$ scenario [$P_z = 512$ B, $Pr = 2.5$ pps].

Figure B-1.4: PDF of the percentage throughput improvement for $DF = 1.5$ scenario [$P_z = 512$ B, $Pr = 2.5$ pps].
Figure B-1.5: PDF of the percentage throughput improvement for $DF = 2$ scenario [$P_z = 512$ B, $Pr = 2.5$ pps].

Figure B-1.6: PDF of the percentage throughput improvement for $DF = 2.5$ scenario [$P_z = 512$ B, $Pr = 2.5$ pps].
Figure B-1.7: PDF of the percentage throughput improvement for $DF = 3$ scenario [$P_z = 512$ B, $Pr = 2.5$ pps].

Figure B-1.8: PDF of the percentage throughput improvement for $DF = 4$ scenario [$P_z = 512$ B, $Pr = 2.5$ pps].
Figure B-1.9: PDF of the percentage throughput improvement for \(DF = 5\) scenario \([P_z = 512 \text{ B}, P_r = 2.5 \text{ pps}]\).

Figure B-1.10: PDF of the percentage throughput improvement for \(DF = 6\) scenario \([P_z = 512 \text{ B}, P_r = 2.5 \text{ pps}]\).
Figure B-1.11: PDF of the percentage throughput improvement for $DF = 8$ scenario [$P_z = 512$ B, $Pr = 2.5$ pps].

Figure B-1.12: PDF of the percentage throughput improvement for $DF = 10$ scenario [$P_z = 512$ B, $Pr = 2.5$ pps].
Figure B-1.13: CCDF of the percentage delay increment for $DF = 1, 1.5, 2, 2.5, \text{ and } 3$ scenario [$P_z = 512 \text{ B}, Pr = 2.5 \text{ pps}$].

Figure B-1.14: CCDF of the percentage delay increment for $DF = 4, 5, 6, 8, \text{ and } 10$ scenarios [$P_z = 512 \text{ B}, Pr = 2.5 \text{ pps}$].
Figure B-1.15: PDF of the percentage delay increment for $DF = 1$ scenario [$P_z = 512$ B, $Pr = 2.5$ pps].

Figure B-1.16: PDF of the percentage delay increment for $DF = 1.5$ scenario [$P_z = 512$ B, $Pr = 2.5$ pps].
Figure B-1.17: PDF of the percentage delay increment for $DF = 2$ scenario [$P_z = 512$ B, $Pr = 2.5$ pps].

Figure B-1.18: PDF of the percentage delay increment for $DF = 2.5$ scenario [$P_z = 512$ B, $Pr = 2.5$ pps].
Figure B-1.19: PDF of the percentage delay increment for $DF = 3$ scenario [$P_z = 512$ B, $P_r = 2.5$ pps].

Figure B-1.20: PDF of the percentage delay increment for $DF = 4$ scenario [$P_z = 512$ B, $P_r = 2.5$ pps].
Figure B-1.21: PDF of the percentage delay increment for $DF = 5$ scenario [$P_z = 512$ B, $P_r = 2.5$ pps].

Figure B-1.22: PDF of the percentage delay increment for $DF = 6$ scenario [$P_z = 512$ B, $P_r = 2.5$ pps].
Figure B-1.23: PDF of the percentage delay increment for $DF = 8$ scenario [$P_z = 512$ B, $Pr = 2.5$ pps].

Figure B-1.24: PDF of the percentage delay increment for $DF = 10$ scenario [$P_z = 512$ B, $Pr = 2.5$ pps].
Appendix B-2

Figure B-2.1: CCDF of the percentage throughput improvement for $DF = 1, 1.5, 2, 2.5, \text{ and } 3$ scenarios [$P_z = 512 \text{ B}, P_r = 10 \text{ pps}$].

Figure B-2.2: CCDF of the percentage throughput improvement for $DF = 4, 5, 6, 8, \text{ and } 10$ scenarios [$P_z = 512 \text{ B}, P_r = 10 \text{ pps}$].
Figure B-2.3: PDF of the percentage throughput improvement for $DF = 1$ scenario [$P_z = 512$ B, $Pr = 10$ pps].

Figure B-2.4: PDF of the percentage throughput improvement for $DF = 1.5$ scenario [$P_z = 512$ B, $Pr = 10$ pps].
Figure B-2.5: PDF of the percentage throughput improvement for $DF = 2$ scenario [$P_z = 512$ B, $Pr = 10$ pps].

Figure B-2.6: PDF of the percentage throughput improvement for $DF = 2.5$ scenario [$P_z = 512$ B, $Pr = 10$ pps].
Figure B-2.7: PDF of the percentage throughput improvement for $DF = 3$ scenario [$P_Z = 512$ B, $Pr = 10$ pps].

Figure B-2.8: PDF of the percentage throughput improvement for $DF = 4$ scenario [$P_Z = 512$ B, $Pr = 10$ pps].
Figure B-2.9: PDF of the percentage throughput improvement for \( DF = 5 \) scenario \([P_z = 512 \text{ B}, Pr = 10 \text{ pps}]\).

Figure B-2.10: PDF of the percentage throughput improvement for \( DF = 6 \) scenario \([P_z = 512 \text{ B}, Pr = 10 \text{ pps}]\).
Figure B-2.11: PDF of the percentage throughput improvement for $DF = 8$ scenario [$P_z = 512$ B, $Pr = 10$ pps].

Figure B-2.12: PDF of the percentage throughput improvement for $DF = 10$ scenario [$P_z = 512$ B, $Pr = 10$ pps].
Figure B-2.13: CCDF of the percentage delay increment for $DF = 1, 1.5, 2, 2.5, \text{ and } 3 \text{ scenarios } [Pr = 512, Pz = 10]$. 

Figure B-2.14: CCDF of the percentage delay increment for $DF = 4, 5, 6, 8, \text{ and } 10 \text{ scenarios } [Pz = 512 \text{ B}, Pr = 10 \text{ pps}].$
Figure B-2.15: PDF of the percentage delay increment for $DF = 1$ scenario [$P_x = 512$ B, $Pr = 10$ pps].

Figure B-2.16: PDF of the percentage delay increment for $DF = 1.5$ scenario [$P_x = 512$ B, $Pr = 10$ pps].
Figure B-2.17: PDF of the percentage delay increment for $DF = 2$ scenario [$P_z = 512$ B, $Pr = 10$ pps].

Figure B-2.18: PDF of the percentage delay increment for $DF = 2.5$ scenario [$P_z = 512$ B, $Pr = 10$ pps].
Figure B-2.19: PDF of the percentage delay increment for $DF = 3$ scenario [$P_z = 512$ B, $P_r = 10$ pps].

Figure B-2.20: PDF of the percentage delay increment for $DF = 4$ scenario [$P_z = 512$ B, $P_r = 10$ pps].
Figure B-2.21: PDF of the percentage delay increment for $DF = 5$ scenario [$P_z = 512$ B, $P_r = 10$ pps].

Figure B-2.22: PDF of the percentage delay increment for $DF = 6$ scenario [$P_z = 512$ B, $P_r = 10$ pps].
Figure B-2.23: PDF of the percentage delay increment for $DF = 8$ scenario [$P_z = 512$ B, $P_r = 10$ pps].

Figure B-2.24: PDF of the percentage delay increment for $DF = 10$ scenario [$P_z = 512$ B, $P_r = 10$ pps].
Figure B-x1: Probability of percentage throughput improvement as a function of node density factor \( [P_z = 512 \text{ B}, P_r = 2.5 \text{ and } 10 \text{ pps}] \).

Figure B-x2: Probability of percentage delay increment as a function of node density factor \( [P_z = 512 \text{ B}, P_r = 2.5 \text{ and } 10 \text{ pps}] \).
Appendix C-1

Figure C.1.1: CCDF of the percentage throughput improvement for $DF = 1, 1.5, 2, 2.5, \text{ and } 3$ scenarios [$P_z = 256 \text{ B}, \ P_r = 5 \text{ pps}$].

Figure C-1.2: CCDF of the percentage throughput improvement for $DF = 4, 5, 6, 8, \text{ and } 10$ scenarios [$P_z = 256 \text{ B}, \ P_r = 5 \text{ pps}$].
Figure C-1.3: PDF of the percentage throughput improvement for $DF = 1$ scenario [$P_z = 256$ B, $Pr = 5$ pps].

Figure C-1.4: PDF of the percentage throughput improvement for $DF = 1.5$ scenario [$P_z = 256$ B, $Pr = 5$ pps].
Figure C-1.5: PDF of the percentage throughput improvement for $DF = 2$ scenario [$P_z = 256$ B, $Pr = 5$ pps].

Figure C-1.6: PDF of the percentage throughput improvement for $DF = 2.5$ scenario [$P_z = 256$ B, $Pr = 5$ pps].
Figure C-1.7: PDF of the percentage throughput improvement for $DF = 3$ scenario [$Pz = 256$ B, $Pr = 5$ pps].

Figure C-1.8: PDF of the percentage throughput improvement for $DF = 4$ scenario [$Pz = 256$ B, $Pr = 5$ pps].
Figure C-1.9: PDF of the percentage throughput improvement for $DF = 5$ scenario [$P_z = 256$ B, $Pr = 5$ pps].

Figure C-1.10: PDF of the percentage throughput improvement for $DF = 6$ scenario [$P_z = 256$ B, $Pr = 5$ pps].
Figure C-1.11: PDF of the percentage throughput improvement for $DF = 8$ scenario [$P_z = 256$ B, $Pr = 5$ pps].

Figure C-1.12: PDF of the percentage throughput improvement for $DF = 10$ scenario [$P_z = 256$ B, $Pr = 5$ pps].
Figure C.1.13: CCDF of the percentage delay increment for $DF = 1, 1.5, 2, 2.5,$ and $3$ scenarios [$P_z = 256$ B, $Pr = 5$ pps].

Figure C.1.14: CCDF of the percentage delay increment for $DF = 4, 5, 6, 8,$ and $10$ scenarios [$P_z = 256$ B, $Pr = 5$ pps].
Figure C-1.15: PDF of the percentage delay increment for \( DF = 1 \) scenario \([Pz = 256 \text{ B}, Pr = 5 \text{ pps}].\)

Figure C-1.16: PDF of the percentage delay increment for \( DF = 1.5 \) scenario \([Pz = 256 \text{ B}, Pr = 5 \text{ pps}].\)
Figure C-1.17: PDF of the percentage delay increment for $DF = 2$ scenario [$P_z = 256$ B, $Pr = 5$ pps].

Figure C-1.18: PDF of the percentage delay increment for $DF = 2.5$ scenario [$P_z = 256$ B, $Pr = 5$ pps].
Figure C-1.19: PDF of the percentage delay increment for $DF = 3$ scenario [$P_z = 256$ B, $Pr = 5$ pps].

Figure C-1.20: PDF of the percentage delay increment for $DF = 4$ scenario [$P_z = 256$ B, $Pr = 5$ pps].
Figure C-1.21: PDF of the percentage delay increment for $DF = 5$ scenario [$Pz = 256$ B, $Pr = 5$ pps].

Figure C-1.22: PDF of the percentage delay increment for $DF = 6$ scenario [$Pz = 256$ B, $Pr = 5$ pps].
Figure C-1.23: PDF of the percentage delay increment for $DF = 8$ scenario $[P_z = 256\ B, \ Pr = 5\ pps]$.

Figure C-1.24: PDF of the percentage delay increment for $DF = 10$ scenario $[P_z = 256\ B, \ Pr = 5\ pps]$. 
Appendix C-2

Figure C.2.1: CCDF of the percentage throughput improvement for $DF = 1, 1.5, 2, 2.5, \text{ and } 3$ scenarios [$P_z = 256 \text{ B}, P_r = 10 \text{ pps}$].

Figure C.2.2: CCDF of the percentage throughput improvement for $DF = 4, 5, 6, 8, \text{ and } 10$ scenarios [$P_z = 256 \text{ B}, P_r = 10 \text{ pps}$].
Figure C-2.3: PDF of the percentage throughput improvement for $DF = 1$ scenario [$P_z = 256$ B, $Pr = 10$ pps].

Figure C-2.4: PDF of the percentage throughput improvement for $DF = 1.5$ scenario [$P_z = 256$ B, $Pr = 10$ pps].
Figure C-2.5: PDF of the percentage throughput improvement for $DF = 2$ scenario [$P_z = 256$ B, $Pr = 10$ pps].

Figure C-2.6: PDF of the percentage throughput improvement for $DF = 2.5$ scenario [$P_z = 256$ B, $Pr = 10$ pps].
Figure C-2.7: PDF of the percentage throughput improvement for $DF = 3$ scenario [$P_z = 256$ B, $Pr = 10$ pps].

Figure C-2.8: PDF of the percentage throughput improvement for $DF = 4$ scenario [$P_z = 256$ B, $Pr = 10$ pps].
Figure C-2.9: PDF of the percentage throughput improvement for $DF = 5$ scenario [$P_z = 256$ B, $Pr = 10$ pps].

Figure C-2.10: PDF of the percentage throughput improvement for $DF = 6$ scenario [$P_z = 256$ B, $Pr = 10$ pps].
Figure C-2.11: PDF of the percentage throughput improvement for $DF = 8$ scenario [$P_z = 256$ B, $Pr = 10$ pps].

Figure C-2.12: PDF of the percentage throughput improvement for $DF = 10$ scenario [$P_z = 256$ B, $Pr = 10$ pps].
Figure C.2.13: CCDF of the percentage delay increment for $DF = 1, 1.5, 2, 2.5, \text{ and } 3$ scenarios [$P_z = 256 \text{ B, } Pr = 10 \text{ pps}$].

Figure C-2.14: CCDF of the percentage delay increment for $DF = 4, 5, 6, 8, \text{ and } 10$ scenarios [$P_z = 256 \text{ B, } Pr = 10 \text{ pps}$].
Figure C-2.15: PDF of the percentage delay increment for $DF = 1$ scenario [$P_z = 256$ B, $Pr = 10$ pps].

Figure C-2.16: PDF of the percentage delay increment for $DF = 1.5$ scenario [$P_z = 256$ B, $Pr = 10$ pps].
Figure C-2.17: PDF of the percentage delay increment for $DF = 2$ scenario [$P_z = 256$ B, $Pr = 10$ pps].

Figure C-2.18: PDF of the percentage delay increment for $DF = 2.5$ scenario [$P_z = 256$ B, $Pr = 10$ pps].
Figure C-2.19: PDF of the percentage delay increment for $DF = 3$ scenario [$P_z = 256$ B, $P_r = 10$ pps].

Figure C-2.20: PDF of the percentage delay increment for $DF = 4$ scenario [$P_z = 256$ B, $P_r = 10$ pps].
Figure C-2.21: PDF of the percentage delay increment for $DF = 5$ scenario [$P_z = 256$ B, $Pr = 10$ pps].

Figure C-2.22: PDF of the percentage delay increment for $DF = 6$ scenario [$P_z = 256$ B, $Pr = 10$ pps].
Figure C-2.23: PDF of the percentage delay increment for $DF = 8$ scenario $[P_z = 256 \text{ B}, Pr = 10 \text{ pps}]$.

Figure C-2.24: PDF of the percentage delay increment for $DF = 10$ scenario $[P_z = 256 \text{ B}, Pr = 10 \text{ pps}]$. 
Appendix C-3

Figure C-3.1: CCDF of the percentage throughput improvement for $DF = 1, 1.5, 2.5, \text{ and } 3$ scenarios [$P_z = 256 \text{ B}, P_r = 20 \text{ pps}$].

Figure C-3.2: CCDF of the percentage throughput improvement for $DF = 4, 5, 6, 8, \text{ and } 10$ scenarios [$P_z = 256 \text{ B}, P_r = 20 \text{ pps}$].
Figure C-3.3: PDF of the percentage throughput improvement for $DF = 1$ scenario [$P_z = 256 \, \text{B}, \, Pr = 20 \, \text{pps}$].

Figure C-3.4: PDF of the percentage throughput improvement for $DF = 1.5$ scenario [$P_z = 256 \, \text{B}, \, Pr = 20 \, \text{pps}$].
Figure C-3.5: PDF of the percentage throughput improvement for \(DF = 2\) scenario \([P_z = 256\ B, Pr = 20\ pps]\).

Figure C-3.6: PDF of the percentage throughput improvement for \(DF = 2.5\) scenario \([P_z = 256\ B, Pr = 20\ pps]\).
Figure C-3.7: PDF of the percentage throughput improvement for $DF = 3$ scenario [$P_z = 256$ B, $Pr = 20$ pps].

Figure C-3.8: PDF of the percentage throughput improvement for $DF = 4$ scenario [$P_z = 256$ B, $Pr = 20$ pps].
Figure C-3.9: PDF of the percentage throughput improvement for $DF = 5$ scenario [$P_z = 256$ B, $Pr = 20$ pps].

Figure C-3.10: PDF of the percentage throughput improvement for $DF = 6$ scenario [$P_z = 256$ B, $Pr = 20$ pps].
Figure C-3.11: PDF of the percentage throughput improvement for $DF = 8$ scenario [$P_z = 256$ B, $Pr = 20$ pps].

Figure C-3.12: PDF of the percentage throughput improvement for $DF = 10$ scenario [$P_z = 256$ B, $Pr = 20$ pps].
Figure C-3.13: CCDF of the percentage delay increment for $DF = 1, 1.5, 2, 2.5, \text{ and } 3$ scenarios [$Pz = 256 \text{ B, } Pr = 20 \text{ pps}$].

Figure C-3.14: CCDF of the percentage delay increment for $DF = 4, 5, 6, 8, \text{ and } 10$ scenarios [$Pz = 256 \text{ B, } Pr = 20 \text{ pps}$].
Figure C-3.15: PDF of the percentage delay increment for $DF = 1$ scenario [$Pz = 256$ B, $Pr = 20$ pps].

Figure C-3.16: PDF of the percentage delay increment for $DF = 1.5$ scenario [$Pz = 256$ B, $Pr = 20$ pps].
Figure C-3.17: PDF of the percentage delay increment for $DF = 2$ scenario [$P_z = 256$ B, $Pr = 20$ pps].

Figure C-3.18: PDF of the percentage delay increment for $DF = 2.5$ scenario [$P_z = 256$ B, $Pr = 20$ pps].
Figure C-3.19: PDF of the percentage delay increment for $DF = 3$ scenario [$P_z = 256$ B, $P_r = 20$ pps].

Figure C-3.20: PDF of the percentage delay increment for $DF = 4$ scenario [$P_z = 256$ B, $P_r = 20$ pps].
Figure C-3.21: PDF of the percentage delay increment for $DF = 5$ scenario [$P_z = 256$ B, $Pr = 20$ pps].

Figure C-3.22: PDF of the percentage delay increment for $DF = 6$ scenario [$P_z = 256$ B, $Pr = 20$ pps].
Figure C-3.23: PDF of the percentage delay increment for $DF = 8$ scenario [$P_z = 256$ B, $P_r = 20$ pps].

Figure C-3.24: PDF of the percentage delay increment for $DF = 10$ scenario [$P_z = 256$ B, $P_r = 20$ pps].
Figure C.x1: Probability of percentage throughput improvement as a function of node density factor $[P_z = 256 \text{ B}, Pr = 5, 10, \text{ and } 20 \text{ pps}]$.

Figure C.x2: Probability of percentage delay increment as a function of node density factor $[P_z = 256 \text{ B}, Pr = 5, 10, \text{ and } 20 \text{ pps}]$. 
Appendix D

Figure D.1: CCDF of the percentage throughput improvement for $DF = 1, 1.5, 2, 2.5,$ and 3 scenarios [$P_z = 128, 512, \text{ and } 1500 \text{ B}, Pr = 5 \text{ pps}$].
Figure D.2: CCDF of the percentage throughput improvement for \( DF = 4, 5, 6, 8, \) and 10 scenarios \([P_z = 128, 512, \text{ and } 1500 \text{ B, } P_r = 5 \text{ pps}].\)

Figure D.3: PDF of the percentage throughput improvement for \( DF = 1 \) scenario \([P_z = 128, 512, \text{ and } 1500 \text{ B, } P_r = 5 \text{ pps}].\)
Figure D.4: PDF of the percentage throughput improvement for $DF = 1.5$ scenario [$P_z = 128, 512, \text{and} 1500 \text{ B}, P_r = 5 \text{ pps}$].

Figure D.5: PDF of the percentage throughput improvement for $DF = 2$ scenario [$P_z = 128, 512, \text{and} 1500 \text{ B}, P_r = 5 \text{ pps}$].
Figure D.6: PDF of the percentage throughput improvement for $DF = 2.5$ scenario [$P_z = 128, 512, \text{ and } 1500 \text{ B, } Pr = 5 \text{ pps}$].

Figure D.7: PDF of the percentage throughput improvement for $DF = 3$ scenario [$P_z = 128, 512, \text{ and } 1500 \text{ B, } Pr = 5 \text{ pps}$].
Figure D.8: PDF of the percentage throughput improvement for $DF = 4$ scenario [$P_z = 128, 512, \text{ and } 1500 \text{ B}, P_r = 5 \text{ pps}$].

Figure D.9: PDF of the percentage throughput improvement for $DF = 5$ scenario [$P_z = 128, 512, \text{ and } 1500 \text{ B}, P_r = 5 \text{ pps}$].
Figure D.10: PDF of the percentage throughput improvement for $DF = 6$ scenario [$P_z = 128, 512, \text{ and } 1500 \text{ B}, P_r = 5 \text{ pps}$].

Figure D.11: PDF of the percentage throughput improvement for $DF = 8$ scenario [$P_z = 128, 512, \text{ and } 1500 \text{ B}, P_r = 5 \text{ pps}$].
Figure D.12: PDF of the percentage throughput improvement for $DF = 10$ scenario [$P_z = 128, 512, \text{ and } 1500 \text{ B}, Pr = 5 \text{ pps}$].

Figure D.13: CCDF of the percentage delay increment for $DF = 1, 1.5, 2, 2.5, \text{ and } 3$ scenarios [$P_z = 128, 512, \text{ and } 1500 \text{ B}, Pr = 5 \text{ pps}$].
Figure D.14: CCDF of the percentage delay increment for $DF = 4, 5, 6, 8,$ and 10 scenarios [$P_z = 128, 512,$ and 1500 B, $Pr = 5$ pps].

Figure D.15: PDF of the percentage delay increment for $DF = 1$ scenario [$P_z = 128, 512,$ and 1500 B, $Pr = 5$ pps].
Figure D.16: PDF of the percentage delay increment for $DF = 1.5$ scenario [$Pz = 128, 512, \text{ and } 1500 \text{ B}, Pr = 5 \text{ pps}$].

Figure D.17: PDF of the percentage delay increment for $DF = 2$ scenario [$Pz = 128, 512, \text{ and } 1500 \text{ B}, Pr = 5 \text{ pps}$].
Figure D.18: PDF of the percentage delay increment for $DF = 2.5$ scenario [$P_z = 128, 512, \text{ and } 1500 \text{ B}, Pr = 5 \text{ pps}$].

Figure D.19: PDF of the percentage delay increment for $DF = 3$ scenario [$P_z = 128, 512, \text{ and } 1500 \text{ B}, Pr = 5 \text{ pps}$].
Figure D.20: PDF of the percentage delay increment for $DF = 4$ scenario [$P_z = 128, 512, \text{ and } 1500 \text{ B}, Pr = 5 \text{ pps}$].

Figure D.21: PDF of the percentage delay increment for $DF = 5$ scenario [$P_z = 128, 512, \text{ and } 1500 \text{ B}, Pr = 5 \text{ pps}$].
Figure D.22: PDF of the percentage delay increment for $DF = 6$ scenario [$P_z = 128, 512, \text{ and } 1500 \text{ B, } Pr = 5 \text{ pps}$].

Figure D.23: PDF of the percentage delay increment for $DF = 8$ scenario [$P_z = 128, 512, \text{ and } 1500 \text{ B, } Pr = 5 \text{ pps}$].
Figure D.24: PDF of the percentage delay increment for \(DF = 10\) scenario \([Pz = 128, 512, \text{ and } 1500 \text{ B}, Pr = 5 \text{ pps}]\).

Figure D.x1: Probability of percentage throughput improvement as a function of node density factor \([Pz = 128, 512, \text{ and } 1500 \text{ B}, Pr = 5 \text{ pps}]\).
Figure D.x2: Probability of percentage delay increment as a function of node density factor \([P_z = 128, 512,\]
and 1500 B, \(P_f = 5\) pps].
Appendix E

Figure E.1: CCDF of the percentage throughput improvement for $DF = 1, 1.5, 2, 2.5, \text{ and } 3$ scenarios ($P_z = 256 \text{ bytes}, P_r = 10 \text{ pps}$), ($P_z = 512 \text{ B}, P_r = 5 \text{ pps}$).

Figure E.2: CCDF of the percentage throughput improvement for $DF = 4, 5, 6, 8, \text{ and } 10$ scenarios ($P_z = 256 \text{ bytes}, P_r = 10 \text{ pps}$), ($P_z = 512 \text{ B}, P_r = 5 \text{ pps}$).
Figure E.3: PDF of the percentage throughput improvement for $DF = 1$ scenario [(Pz = 256 bytes, Pr = 10 pps), (Pz = 512 B, Pr = 5 pps)].

Figure E.4: PDF of the percentage throughput improvement for $DF = 1.5$ scenario [(Pz = 256 bytes, Pr = 10 pps), (Pz = 512 B, Pr = 5 pps)].
Figure E.5: PDF of the percentage throughput improvement for $DF = 2$ scenario [(Pz = 256 bytes, Pr = 10 pps), (Pz = 512 B, Pr = 5 pps)].

Figure E.6: PDF of the percentage throughput improvement for $DF = 2.5$ scenario [(Pz = 256 bytes, Pr = 10 pps), (Pz = 512 B, Pr = 5 pps)].
Figure E.7: PDF of the percentage throughput improvement for $DF = 3$ scenario [(Pz = 256 bytes, Pr = 10 pps), (Pz = 512 B, Pr = 5 pps)].

Figure E.8: PDF of the percentage throughput improvement for $DF = 4$ scenario [(Pz = 256 bytes, Pr = 10 pps), (Pz = 512 B, Pr = 5 pps)].
Figure E.9: PDF of the percentage throughput improvement for $DF = 5$ scenario \[((P_z = 256 \text{ bytes, } Pr = 10 \text{ pps}), \ (P_z = 512 \text{ B, } Pr = 5 \text{ pps}))\].

Figure E.10: PDF of the percentage throughput improvement for $DF = 6$ scenario \[((P_z = 256 \text{ bytes, } Pr = 10 \text{ pps}), \ (P_z = 512 \text{ B, } Pr = 5 \text{ pps}))\].
Figure E.11: PDF of the percentage throughput improvement for $DF = 8$ scenario [(Pz = 256 bytes, Pr = 10 pps), (Pz = 512 B, Pr = 5 pps)].

Figure E.12: PDF of the percentage throughput improvement for $DF = 10$ scenario [(Pz = 256 bytes, Pr = 10 pps), (Pz = 512 B, Pr = 5 pps)].
Figure E.13: CCDF of the percentage delay increment for $DF = 1, 1.5, 2, 2.5, \text{ and } 3$ scenarios [(Pz = 256 bytes, Pr = 10 pps), (Pz = 512 B, Pr = 5 pps)].

Figure E.14: CCDF of the percentage delay increment for $DF = 4, 5, 6, 8, \text{ and } 10$ scenarios [(Pz = 256 bytes, Pr = 10 pps), (Pz = 512 B, Pr = 5 pps)].
Figure E.15: PDF of the percentage delay increment for $DF = 1$ scenario [($P_z = 256$ bytes, $P_r = 10$ pps), ($P_z = 512$ B, $P_r = 5$ pps)].

Figure E.16: PDF of the percentage delay increment for $DF = 1.5$ scenario [($P_z = 256$ bytes, $P_r = 10$ pps), ($P_z = 512$ B, $P_r = 5$ pps)].
Figure E.17: PDF of the percentage delay increment for $DF = 2$ scenario \([(Pz = 256 \text{ bytes, } Pr = 10 \text{ pps}), (Pz = 512 \text{ B, } Pr = 5 \text{ pps})]\).

Figure E.18: PDF of the percentage delay increment for $DF = 2.5$ scenario \([(Pz = 256 \text{ bytes, } Pr = 10 \text{ pps}), (Pz = 512 \text{ B, } Pr = 5 \text{ pps})]\).
Figure E.19: PDF of the percentage delay increment for $DF = 3$ scenario [(Pz = 256 bytes, Pr = 10 pps), (Pz = 512 B, Pr = 5 pps)].

Figure E.20: PDF of the percentage delay increment for $DF = 4$ scenario [(Pz = 256 bytes, Pr = 10 pps), (Pz = 512 B, Pr = 5 pps)].
Figure E.21: PDF of the percentage delay increment for $DF = 5$ scenario \([P_z = 256 \text{ bytes}, Pr = 10 \text{ pps}), (P_z = 512 \text{ B}, Pr = 5 \text{ pps})]\).

Figure E.22: PDF of the percentage delay increment for $DF = 6$ scenario \([P_z = 256 \text{ bytes}, Pr = 10 \text{ pps}), (P_z = 512 \text{ B}, Pr = 5 \text{ pps})]\).
Figure E.23: PDF of the percentage delay increment for $DF = 8$ scenario $[(P_z = 256 \text{ bytes}, P_r = 10 \text{ pps}), (P_z = 512 \text{ B}, P_r = 5 \text{ pps})]$.

Figure E.24: PDF of the percentage delay increment for $DF = 10$ scenario $[(P_z = 256 \text{ bytes}, P_r = 10 \text{ pps}), (P_z = 512 \text{ B}, P_r = 5 \text{ pps})]$.
Figure E.x1: Probability of percentage throughput improvement as a function of node density factor [(\(P_z = 256\) bytes, \(Pr = 10\) pps), \((P_z = 512\) B, \(Pr = 5\) pps)].

Figure E.x2: Probability of percentage delay increment as a function of node density factor [(\(P_z = 256\) bytes, \(Pr = 10\) pps), \((P_z = 512\) B, \(Pr = 5\) pps)].
Figure F.1: CCDF of the percentage throughput improvement for $DF = 2$ scenarios.
Figure F.2: CCDF of the percentage throughput improvement for $DF = 4$ scenarios.

Figure F.3: CCDF of the percentage throughput improvement for $DF = 6$ scenarios.

Figure F.4: PDF of the percentage throughput improvement for $DF = 2$ scenarios.
Figure F.5: PDF of the percentage throughput improvement for $DF = 4$ scenarios.

Figure F.6: PDF of the percentage throughput improvement for $DF = 6$ scenarios.
Figure F.7: CCDF of the percentage delay increment for DF = 2 scenarios.

Figure F.8: CCDF of the percentage delay increment for DF = 4 scenarios.
Figure F.9: CCDF of the percentage delay increment for $DF = 6$ scenarios.

Figure F.10: PDF of the percentage delay increment for $DF = 2$ scenario.
Figure F.11: PDF of the percentage delay increment for $DF = 4$ scenario.

Figure F.12: PDF of the percentage delay increment for $DF = 6$ scenario.
Appendix G

Figure G.1: CCDF of the percentage throughput improvement for $DF = 2$ scenarios.

Figure G.2: PDF of the percentage throughput improvement for network of centered gateway scenario of $DF = 2$. 
Figure G.3: PDF of the percentage throughput improvement for network of two gateways scenario of $DF = 2$.

Figure G.4: PDF of the percentage throughput improvement for network of three gateways scenario of $DF = 2$. 
Figure G.5: PDF of the percentage throughput improvement for network of four gateways scenario of $DF = 2$.

Figure G.6: CCDF of the percentage delay increment for $DF = 2$ scenarios.
Figure G.7: PDF of the percentage delay increment for network of centered gateway scenario of $DF = 2$.

Figure G.8: PDF of the percentage delay increment for network of two gateways scenario of $DF = 2$. 
Figure G.9: PDF of the percentage delay increment for network of three gateways scenario of $DF = 2$.

Figure G.10: PDF of the percentage delay increment for network of four gateways scenario of $DF = 2$. 
Appendix H

Figure H.1: CCDF of the percentage throughput improvement for $DF = 2$ scenarios of different $Hc$ limits.

Figure H.2: PDF of the percentage throughput improvement for $DF = 2$ scenarios of $Hc = 7$. 
Figure H.3: PDF of the percentage throughput improvement for $DF = 2$ scenarios of $Hc = 6$.

Figure H.4: PDF of the percentage throughput improvement for $DF = 2$ scenarios of $Hc = 5$. 
Figure H.5: CCDF of the percentage delay increment for $DF=2$ scenarios of different $H_c$ limits.

Figure H.6: PDF of the percentage delay increment for $DF=2$ scenarios of $H_c=7$. 
Figure H.7: PDF of the percentage delay increment for $DF = 2$ scenarios of $Hc = 6$.

Figure H.8: PDF of the percentage delay increment for $DF = 2$ scenarios of $Hc = 5$. 

353
Appendix I

Figure I.1: CCDF of the percentage throughput improvement for $DF = 2$ scenario.

Figure I.2: CCDF of the percentage throughput improvement for $DF = 4$ scenario.
Figure I.3: CCDF of the percentage throughput improvement for $DF = 6$ scenario.

Figure I.4: PDF of the percentage throughput improvement for $DF = 2$ scenario.
Figure I.5: PDF of the percentage throughput improvement for $DF = 4$ scenario.

Figure I.6: PDF of the percentage throughput improvement for $DF = 6$ scenario.
Figure I.7: CCDF of the percentage global delay increment for $DF = 2$ scenarios.

Figure I.8: CCDF of the percentage global delay increment for $DF = 4$ scenarios.
Figure I.9: CCDF of the percentage global delay increment for $DF = 6$ scenarios.

Figure I.10: PDF of the percentage delay increment for $DF = 2$ scenario.
Figure I.11: PDF of the percentage delay increment for $DF = 4$ scenario.

Figure I.12: PDF of the percentage delay increment for $DF = 6$ scenario.
Appendix J

Figure J.1: CCDF of the percentage throughput improvement for $DF = 2$ scenarios.

Figure J.2: CCDF of the percentage throughput improvement for $DF = 4$ scenarios.
Figure J.3: CCDF of the percentage throughput improvement for $DF = 6$ scenarios.

Figure J.4: CCDF of the percentage global delay increment for $DF = 2$ scenarios.
Figure J.5: CCDF of the percentage global delay increment for $DF = 4$ scenarios.

Figure J.6: CCDF of the percentage global delay increment for $DF = 6$ scenarios.
Appendix K

Figure K.1: CCDF of the percentage throughput improvement for $DF = 2$ scenarios.

Figure K.2: CCDF of the percentage throughput improvement for $DF = 4$ scenarios.
Figure K.3: CCDF of the percentage throughput improvement for $DF = 6$ scenarios.

Figure K.4: CCDF of the percentage global delay increment for $DF = 2$ scenarios.
Figure K.5: CCDF of the percentage global delay increment for $DF = 4$ scenarios.

Figure K.6: CCDF of the percentage delay increment for $DF = 6$ scenarios.
Appendix K

Figure L.1: CCDF of the percentage throughput improvement for $DF = 2$ scenarios when the calculation of the $AEF$ metric based on 2 second interval time.
Figure L.2: CCDF of the percentage delay increment for $DF = 2$ scenarios when the calculation of the $AEF$ metric based on 2 second interval time.

Figure L.3: CCDF of the percentage throughput improvement for $DF = 2$ scenarios when the calculation of the $AEF$ metric based on 5 second interval time.
Figure L.4: CCDF of the percentage delay increment for $DF = 2$ scenarios when the calculation of the $AEF$ metric based on 5 second interval time.

Figure L.5: CCDF of the percentage throughput improvement for $DF = 2$ scenarios when the calculation of the $AEF$ metric based on 0.5 second interval time.
Figure L.6: CCDF of the percentage delay increment for $DF = 2$ scenarios when the calculation of the $AEF$ metric based on 0.5 second interval time.