Spectrum-Aware Routing In Cognitive Radio MANETs

by

Shahin, Shariat

Submitted for the Degree of Doctor of Philosophy

Faculty of Engineering & Physical Sciences

Institute for Communication Systems (ICS)
5G Innovation Centre (5GIC)

University of Surrey
Guildford, Surrey GU2 7XH, UK

Supervisors:
Prof Rahim Tafazolli
Dr Seiamk Vahid

© Shahin Shariat, 2017
Dedicated to my mom’s kind heart and to my dad’s warm hands…
Statement of Originality

This thesis and the work to which it refers are the results of my own efforts. Any ideas, data, images or text resulting from the work of others (whether published or unpublished) are fully identified as such within the work and attributed to their originator in the text, bibliography or in footnotes. This thesis has not been submitted in whole or in part for any other academic degree or professional qualification. I agree that the University has the right to submit my work to the plagiarism detection service Turnitin UK for originality checks. Whether or not drafts have been so-assessed, the University reserves the right to require an electronic version of the final document (as submitted) for assessment as above.

Shahin Shariat
December, 2017
Cognitive Radio (CR) provides a promising means to the more efficient use of available spectrum. Routing in multi-hop wireless networks remains challenging and introduction of CR technology has created additional demands on routing within Cognitive Radio Mobile Ad-hoc Networks (CR-MANETs). To address these challenges, spectrum-aware routing protocols aiming at dynamic utilization of the so-called spectrum opportunities have been developed recently to improve end-to-end performance of the network for example in terms of Delay, Packet Loss and Throughput. One of the bottlenecks in the performance of ad hoc networks has been the lack of a load balancing mechanism. With the addition of potential routing opportunities introduced by CR technology, a load balanced routing protocol which can utilize SOPs into the load balancing mechanism is a missing puzzle in the problem of routing in CR-technology. Quantum game theory provides a framework to utilize entangled particles with the aim of affecting decision-making process of distant players. Hence, this theory has the potential to be used as a framework to target the load balancing problem in ad hoc networks.

First, a novel spectrum-aware routing protocol based on OLSR as the basis of implementation is proposed in this research. The proposed algorithm utilizes ETX as the link quality estimation metric and provides the best weigh end-to-end paths based on generalization of Dijkstra’s algorithm to multigraphs. The results demonstrate that the proposed algorithm outperforms the existing baseline OLSR routing algorithm. Due to the instability in the end-to-end delay performance of the proposed algorithm, backpressure algorithm is identified as a potential solution to stabilize queues in the network and target the shortcoming of the proposed algorithm. Hence, a novel spectrum-aware routing algorithm based on backpressure load balancing mechanism is proposed and compared against the baseline OLSR and the proposed spectrum-aware OLSR algorithm. The OLSR backpressure spectrum-aware (OLSR-BSA) routing algorithm not only optimizes route computation based on the predefined cost metric but also incorporates the queue gradients of backpressure algorithm to perform load balancing. The results proof that the backpressure algorithm can efficiently utilize the SOPs in the load balancing optimization problem and results a performance and stability gain both in terms of end-to-end delay and packet delivery ratio. The instability resulted by inaccuracy of queue information in the proposed OLSR-BSA algorithm motivated our research to explore the problem of load balancing from a completely new perspective of quantum game theory. We have formulated the problem of load balancing in ad hoc networks using quantum game theory and proposed a novel routing algorithm so called Quantum Load Balanced OLSR (QLB-OLSR). The simulation results demonstrate a significant load balancing stability gain against the baseline OLSR routing algorithm.

**Key words:** Spectrum, Routing, Cognitive Radio, Spectrum-aware routing, Backpressure Algorithm, Quantum Game Theory, Quantum Load balancing

Email: s.shariat@surrey.ac.uk
Acknowledgements

I would like to express my highest gratitude to my supervisors without which this journey would have been impossible to complete. I would like to firstly thank my principal supervisor, Prof Rahim Tafazolli, for his endless support throughout this journey. I also would like to thank my co-supervisor Dr Seiamak Vahid for his calm and patient support throughout my research.

I was blessed to have a caring and supportive brother, Dr Mehrdad Shariat who has always been my role model for success; he never gave up on me and I cannot express my feeling when I told him, “I finally finished it bro”. I am also very grateful to have met so many talented individuals in the 5GIC which made my research experience invaluable.
# Contents

Statement of Originality........................................................................................................... iv

Summary ..................................................................................................................................... v

Acknowledgements .................................................................................................................. vi

List of Figures ............................................................................................................................. xi

List of Tables ................................................................................................................................. xiv

List of Abbreviations .................................................................................................................... xv

Chapter 1 ......................................................................................................................................... 1

Introduction ............................................................................................................................... 1

1.1 Overview ............................................................................................................................... 1

1.2 Motivations, Novelty and Challenges .................................................................................. 2

1.3 Contributions ......................................................................................................................... 4

1.4 Thesis Structure ..................................................................................................................... 5

Chapter 2 ......................................................................................................................................... 6

2 Background Study and Related Research............................................................................... 6

2.1 Cognitive Radio ..................................................................................................................... 6

2.2 Correlation of Spectrum with routing ................................................................................... 7

2.2.1 Fundamentals of Cognitive Radio ..................................................................................... 8

2.3 Wireless Ad hoc Networks .................................................................................................. 9

2.3.1 Excessive Noise and Interference .................................................................................... 10

2.3.2 Dynamic topology ............................................................................................................ 10

2.3.3 Security ............................................................................................................................. 11

2.3.4 Load Balancing................................................................................................................... 11

2.4 Cognitive Radio and Mobile Ad-hoc Networks (MANETs).................................................. 12

2.5 Dynamic Spectrum Utilization in CR-based Multi-Hop Ad-hoc Networks.......................... 13

2.5.1 DSA Spectrum Management, Categorizations and Challenges ....................................... 14

2.6 Spectrum Aware Routing ..................................................................................................... 17

2.7 Challenges in Spectrum-aware Routing ................................................................................. 18

2.7.1 Control Channel.................................................................................................................... 18
2.7.2 Coupled and decoupled route selection and spectrum management .................. 19
2.7.3 Sensing Techniques and the impact on Routing ........................................... 20
2.7.4 Distribution of SOP information .................................................................... 21
2.7.5 Spectrum-aware signalling mechanism ......................................................... 21
2.7.6 Switching Delay ............................................................................................ 22

2.8 Survey on Spectrum-aware Routing Protocols .................................................. 22

2.8.1 A spectrum-aware routing for multi-hop single transceiver Cognitive Radio networks (MSCRN) .................................................................................. 22
2.8.2 SEARCH ....................................................................................................... 23
2.8.3 SPEAR ......................................................................................................... 24
2.8.4 ASAR ............................................................................................................ 24
2.8.5 SAMER ........................................................................................................ 25
2.8.6 SARP ............................................................................................................ 26
2.8.7 IPSAG ......................................................................................................... 26
2.8.8 Channel assignment and bandwidth allocation algorithm for multi-channel wireless mesh networks ......................................................................................... 27
2.8.9 SAOR ............................................................................................................ 28
2.8.10 Coupled/decoupled route and spectrum selection in DSA networks............... 28
2.8.11 A layered graph model to target routing in DSA .......................................... 29
2.8.12 On-demand spectrum-aware routing ......................................................... 29

2.9 Taxonomy of spectrum-aware routing protocols ................................................ 30

2.10 Background study, load balancing in ad hoc networks .................................... 32

2.11 Back-pressure Routing .................................................................................... 34

2.11.1 Backpressure Queue Structure ................................................................... 35
2.11.2 Back-pressure Mathematical Model ............................................................ 35
2.11.3 Back-pressure Routing in Networks with Dynamic Channels/links .............. 36

2.12 Quantum Game Theory and Load Balancing ................................................... 36

Chapter 3 .................................................................................................................. 38

3 Spectrum-aware Routing ....................................................................................... 38

3.1 Introduction ...................................................................................................... 38
3.2 Direction and Vision ......................................................................................... 39
3.3 Performance Comparison of Conventional MANET Routing Protocols .......... 40

3.3.1 Speed Variation ............................................................................................ 41
3.3.2 Relay Node Density Variation ....................................................................... 44
3.3.3 Conclusion on Benchmarking of MANET Routing Protocols ....................... 46
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.2</td>
<td>Quantum entanglement, from concept to application</td>
<td>112</td>
</tr>
<tr>
<td>5.3</td>
<td>Quantum Game and Entanglement</td>
<td>114</td>
</tr>
<tr>
<td>5.3.1</td>
<td>Rotation and quantum strategies</td>
<td>114</td>
</tr>
<tr>
<td>5.3.2</td>
<td>Quantum advice and entangled opinions</td>
<td>116</td>
</tr>
<tr>
<td>5.3.3</td>
<td>Doublet topology</td>
<td>116</td>
</tr>
<tr>
<td>5.3.4</td>
<td>Triplet topology</td>
<td>120</td>
</tr>
<tr>
<td>5.3.5</td>
<td>Quantum advice</td>
<td>123</td>
</tr>
<tr>
<td>5.4</td>
<td>Ad hoc Load Balancing Problem Definition</td>
<td>123</td>
</tr>
<tr>
<td>5.5</td>
<td>OLSR Routing Protocol as the Baseline of Implementation</td>
<td>124</td>
</tr>
<tr>
<td>5.5.1</td>
<td>OLSR Basic Operation</td>
<td>125</td>
</tr>
<tr>
<td>5.6</td>
<td>Implementation of QLB in OLSR</td>
<td>126</td>
</tr>
<tr>
<td>5.6.1</td>
<td>Topology Identification</td>
<td>126</td>
</tr>
<tr>
<td>5.6.2</td>
<td>Traffic Based Load Balancing</td>
<td>126</td>
</tr>
<tr>
<td>5.6.3</td>
<td>Instantaneous Load Balancing</td>
<td>127</td>
</tr>
<tr>
<td>5.6.4</td>
<td>QLB Algorithm</td>
<td>127</td>
</tr>
<tr>
<td>5.7</td>
<td>QLB-OLSR Simulations and results</td>
<td>129</td>
</tr>
<tr>
<td>5.7.1</td>
<td>Simulation Setup and Assumptions</td>
<td>129</td>
</tr>
<tr>
<td>5.7.2</td>
<td>Simulation Parameters</td>
<td>130</td>
</tr>
<tr>
<td>5.8</td>
<td>Simulation Results</td>
<td>132</td>
</tr>
<tr>
<td>5.8.1</td>
<td>2-Relay Simulation Study</td>
<td>132</td>
</tr>
<tr>
<td>5.8.2</td>
<td>3-Relay Simulation Study</td>
<td>135</td>
</tr>
<tr>
<td>5.9</td>
<td>Summary</td>
<td>137</td>
</tr>
<tr>
<td>6</td>
<td>Conclusion and Future Work</td>
<td>139</td>
</tr>
<tr>
<td>6.1</td>
<td>Conclusion</td>
<td>139</td>
</tr>
<tr>
<td>6.2</td>
<td>Future Work</td>
<td>140</td>
</tr>
<tr>
<td></td>
<td>Bibliography</td>
<td>142</td>
</tr>
</tbody>
</table>
List of Figures

Figure 2-1: Cognitive Radio and the necessity of Cross Layering ................................................. 7
Figure 2-2: Characteristics of MANETs ..................................................................................... 12
Figure 2-3: Taxonomy of Spectrum Management Architectures .................................................... 14
Figure 2-4: Spectrum Management Architectures ........................................................................ 15
Figure 2-5: Categorization of Channel to Interface Assignments ................................................ 16
Figure 2-6: An example for coupled and decoupled design of DSA systems ................................ 19
Figure 2-7: Categories of Load-balanced Routing ...................................................................... 32
Figure 3-1: End-to-End Delay VS Speed (OLSR, AODV, DYMO and DSR) .................................. 42
Figure 3-2: Packet Delivery Ratio VS Speed (OLSR, AODV, DYMO and DSR) ........................... 43
Figure 3-3: Normalized Routing Overhead VS Speed (OLSR, AODV, DYMO and DSR) .......... 43
Figure 3-4: End-to-End Delay VS Relay Node Density (OLSR, AODV, DYMO and DSR) ......... 45
Figure 3-5: Packet Delivery Ratio VS Relay Node Density (OLSR, AODV, DYMO and DSR) .... 45
Figure 3-6: Normalized Routing overhead VS Relay Node Density (OLSR, AODV, DYMO and DSR) .................................................................................................................. 46
Figure 3-7: MPR Optimization in OLSR .................................................................................... 47
Figure 3-8: End-to-End Delay versus Speed (OLSR, OLSR-ETX, OLSR-MD and OLSR-ML) ...... 50
Figure 3-9: Packet Delivery Ratio versus Speed (OLSR, OLSR-ETX, OLSR-MD and OLSR-ML) ... 51
Figure 3-10: Normalized Routing Overhead versus Speed (OLSR, OLSR-ETX, OLSR-MD and OLSR-ML) .................................................................................................................. 51
Figure 3-11: End-to-End Delay versus Load (OLSR, OLSR-ETX, OLSR-MD and OLSR-ML) ...... 52
Figure 3-12: Packet Delivery Ratio Versus Load (OLSR, OLSR-ETX, OLSR-MD and OLSR-ML) . 52
Figure 3-13: Normalized Routing Overhead Versus Load (OLSR, OLSR-ETX, OLSR-MD and OLSR-ML) .................................................................................................................. 53
Figure 3-14: Channel Switching Triggers Implemented in OMNET++ ........................................... 55
Figure 3-15: Example simulation scenario to validate channel switching .................................... 56
Figure 3-16: End-to-End Delay vs. Simulation Time, before and After Channel Switching ......... 57
Figure 3-17: Flowchart of GDA Integrated in Route Discovery Phase of OLSR .............................. 60
Figure 3-18: An Example Scenario, Applying GDA Graph Transformation ................................. 60
Figure 3-19: Spectrum-aware routing scenario .......................................................................... 62
Figure 3-20: End-to-End ETX vs. Simulation Time (s) (Without Channel Switching) .................... 64
Figure 3-21: End-to-End ETX versus Simulation Time (s) (With Channel Switching) ............... 64
Figure 3-22: End-to-End Delay (s) vs. Simulation Time (s) ......................................................... 65
Figure 3-23: MAC Delay (s) vs. Simulation Time (s) (Without Channel Switching) ..................... 66
Figure 3-24: MAC Delay (s) versus Simulation Time (s) (With Channel Switching) ................. 66
List of Figures

Figure 3-25: End-to-End Delay VS Speed (OLSR-ETX and OLSR-SpectrumAware) ............................................ 69
Figure 3-26: PDR VS Speed (OLSR-ETX and OLSR-SpectrumAware) ................................................................. 70
Figure 3-27: Routing Overhead (Norm.) VS Speed (OLSR-ETX and OLSR-SpectrumAware) .......................... 70
Figure 3-28: End-to-End Delay vs RND (OLSR-ETX and OLSR-SpectrumAware) ........................................... 71
Figure 3-29: PDR vs RND (OLSR-ETX and OLSR-SpectrumAware) ................................................................. 72
Figure 3-30: Routing Overhead (Norm.) vs RND (OLSR-ETX and OLSR-SpectrumAware) ........................... 72
Figure 3-31: End-to-End Delay vs Load (OLSR-ETX and OLSR-SpectrumAware) ........................................... 73
Figure 3-32: PDR vs Load (OLSR-ETX and OLSR-SpectrumAware) ................................................................. 73
Figure 3-33: Routing Overhead (Norm.) vs Load (OLSR-ETX and OLSR-SpectrumAware) .......................... 74
Figure 4-1: Sample of Routing Table in Spectrum-aware OLSR ........................................................................... 83
Figure 4-2: Sample Topology for Multi-Path Scenario ....................................................................................... 86
Figure 4-3: Multi-path Route Extraction and Computation Algorithm ............................................................. 87
Figure 4-4: OLSR’s HELLO Message with the Queue Information ................................................................. 88
Figure 4-5: OSI Model .................................................................................................................................... 89
Figure 4-6: Example of a Multi-Channel Network ............................................................................................ 91
Figure 4-7: End-to-End Delay (s) vs Speed (m/s) OLSR-BSA, OLSR-SA and OLSR-ETX .......................... 96
Figure 4-8: Packet Delivery Ratio vs Speed (m/s) OLSR-BSA, OLSR-SA and OLSR-ETX ....................... 97
Figure 4-9: Normalized Routing Overhead vs Speed (m/s) OLSR-BSA, OLSR-SA and OLSR-ETX .... 97
Figure 4-10: End-to-End Delay (s) vs RND OLSR-BSA, OLSR-SA and OLSR-ETX ............................... 99
Figure 4-11: Packet Delivery Ratio vs RND OLSR-BSA, OLSR-SA and OLSR-ETX .................................. 100
Figure 4-12: Normalized Routing Overhead vs RND OLSR-BSA, OLSR-SA and OLSR-ETX .......... 100
Figure 4-13: End-to-End Delay (s) vs Load (KB/s) OLSR-BSA, OLSR-SA and OLSR-ETX .................. 102
Figure 4-14: Packet Delivery Ratio vs Load (KB/s) OLSR-BSA, OLSR-SA and OLSR-ETX ............ 102
Figure 4-15: Normalized Routing Overhead vs Load (KB/s) OLSR-BSA, OLSR-SA and OLSR-ETX .... 103
Figure 4-16: End-to-End Delay (s) vs Speed (m/s), OLSR-BSA, ROSA and DIBAPS .......................... 104
Figure 4-17: Packet Delivery Ratio vs Speed (m/s), OLSR-BSA, ROSA and DIBAPS .......................... 105
Figure 4-18: End-to-End Delay (s) vs Load (KB/s), OLSR-BSA, ROSA and DIBAPS .......................... 106
Figure 4-19: Packet Delivery Ratio vs Load (KB/s), OLSR-BSA, ROSA and DIBAPS ........................ 106
Figure 4-20: End-to-End Delay (s) vs RND, OLSR-BSA, ROSA and DIBAPS ................................. 107
Figure 4-21: Packet Delivery Ratio vs RND, OLSR-BSA, ROSA and DIBAPS .................................. 107
Figure 4-22: Average Queue Length vs Speed (m/s), OLSR-BSA, ROSA and DIBAPS ................... 108
Figure 4-23: Average Queue Length vs Load (KB/s), OLSR-BSA, ROSA and DIBAPS .................. 109
Figure 4-24: Average Queue Length vs RND, OLSR-BSA, ROSA and DIBAPS ............................. 109
Figure 5-1: Doublet topology ......................................................................................................................... 117
Figure 5-2: Triplet topology ........................................................................................................................... 120
Figure 5-3: Ad hoc Topology Scenario ......................................................................................................... 124
Figure 5-4: Quantum Load Balancing (QLB) Algorithm ................................................................. 128
Figure 5-5: 3-Relay Simulation Setup.................................................................................................. 130
Figure 5-6: Throughput Unbalanced Factor, 2-Relay (box-plot) ....................................................... 133
Figure 5-7: Throughput Unbalanced Factor, 2-Relay (time-plot) ....................................................... 133
Figure 5-8: End to End Delay, 2-Relay ............................................................................................... 134
Figure 5-9: Jitter, 2-Relay .................................................................................................................... 135
Figure 5-10: Throughput Unbalanced Factor, 3-Relay (box-plot) ...................................................... 135
Figure 5-11: Throughput Unbalanced Factor, 3-Relay (time-plot) ...................................................... 136
Figure 5-12: End to End Delay, 3-Relay ............................................................................................... 136
Figure 5-13: Jitter, 3-Relay .................................................................................................................... 137
List of Tables

Table 2-1: A Taxonomy of Spectrum-aware routing protocols ......................................................... 31
Table 3-1: Benchmarking Simulation Parameters................................................................................. 40
Table 3-2: Spectrum-aware Simulation scenario parameters................................................................. 63
Table 3-3: Simulation Parameter, spectrum-aware OLSR ................................................................. 68
Table 4-1: Example of Routing Table .................................................................................................. 92
Table 5-1: Probabilities of Path Selection by Sender Nodes – Triplet Topology ................................. 122
Table 5-2: Simulation Parameters ....................................................................................................... 131
# List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AODV</td>
<td>Ad hoc On-Demand Distance Vector</td>
</tr>
<tr>
<td>BSA</td>
<td>Backpressure Spectrum Aware</td>
</tr>
<tr>
<td>CR</td>
<td>Cognitive Radio</td>
</tr>
<tr>
<td>CSLAR</td>
<td>Content Sensitive Load Aware Routing</td>
</tr>
<tr>
<td>CSMA - CA</td>
<td>Carrier Sense Multiple Access – Collision Avoidance</td>
</tr>
<tr>
<td>DCF</td>
<td>Distributed Coordination Function</td>
</tr>
<tr>
<td>DLAR</td>
<td>Dynamic Load-Aware Routing</td>
</tr>
<tr>
<td>DSA</td>
<td>Dynamic Spectrum Access</td>
</tr>
<tr>
<td>DLAR</td>
<td>Dynamic Load-Aware Routing</td>
</tr>
<tr>
<td>DSR</td>
<td>Dynamic Source Routing</td>
</tr>
<tr>
<td>DYMO</td>
<td>Dynamic Manet On-demand</td>
</tr>
<tr>
<td>E-t-E</td>
<td>End to End</td>
</tr>
<tr>
<td>ETX</td>
<td>Expected Transmission Count</td>
</tr>
<tr>
<td>FCC</td>
<td>Federal Communication Commission</td>
</tr>
<tr>
<td>GPS</td>
<td>Global Positioning System</td>
</tr>
<tr>
<td>IMPSF</td>
<td>Intelligent Multi-Path Selection Function</td>
</tr>
<tr>
<td>ISM</td>
<td>Industrial Scientific and Medical</td>
</tr>
<tr>
<td>LAOR</td>
<td>Load-Aware On-Demand Routing</td>
</tr>
<tr>
<td>LWR</td>
<td>Load Aware Routing</td>
</tr>
<tr>
<td>MAC</td>
<td>Media Access Control</td>
</tr>
<tr>
<td>MANET</td>
<td>Mobile Ad hoc Network</td>
</tr>
<tr>
<td>MD</td>
<td>Minimum Delay</td>
</tr>
<tr>
<td>MISF</td>
<td>Multi-Interface devices</td>
</tr>
<tr>
<td>ML</td>
<td>Minimum Loss</td>
</tr>
<tr>
<td>MPR</td>
<td>Multi Point Relay</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------------------</td>
</tr>
<tr>
<td>NIC</td>
<td>Network Interface Card</td>
</tr>
<tr>
<td>OLSR</td>
<td>Optimised Link State Routing</td>
</tr>
<tr>
<td>OLT</td>
<td>Opportunistic Link Transmission</td>
</tr>
<tr>
<td>OSI</td>
<td>Open Systems Interconnection</td>
</tr>
<tr>
<td>PDR</td>
<td>Packet Delivery Ratio</td>
</tr>
<tr>
<td>PU</td>
<td>Primary User</td>
</tr>
<tr>
<td>QoS</td>
<td>Quality of Service</td>
</tr>
<tr>
<td>RAT</td>
<td>Radio Access Technology</td>
</tr>
<tr>
<td>RERR</td>
<td>Route Error</td>
</tr>
<tr>
<td>RND</td>
<td>Relay Node Density</td>
</tr>
<tr>
<td>RREP</td>
<td>Route Reply</td>
</tr>
<tr>
<td>RREQ</td>
<td>Route REQuest</td>
</tr>
<tr>
<td>SA</td>
<td>Spectrum Aware</td>
</tr>
<tr>
<td>SDR</td>
<td>Software Defined Radio</td>
</tr>
<tr>
<td>SOP</td>
<td>Spectrum Opportunity</td>
</tr>
<tr>
<td>SSC</td>
<td>Share Spectrum Company</td>
</tr>
<tr>
<td>SU</td>
<td>Secondary User</td>
</tr>
<tr>
<td>TDMA</td>
<td>Time Division Multiple Access</td>
</tr>
<tr>
<td>U-NII</td>
<td>Unlicensed National Information Infrastructure</td>
</tr>
<tr>
<td>VBR</td>
<td>Variable Bit Rate</td>
</tr>
<tr>
<td>ZRP</td>
<td>Zone Routing Protocol</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

1.1 Overview

Wireless ad hoc networks have been one of the popular topics of research for many years. Due to the lack of infrastructure, ad hoc networks have been subject to many limitations such as power constraints, low capacity, instability and etc. However, independency of ad hoc networks to a fixed infrastructure has made them ideal for many applications such as emergency services and tactical/military operations. Data communication in such networks relies on effective cooperation of the nodes in creating self-reliant end-to-end connections. Routing in such networks is defined as the process of delivering data packets from source to destination nodes via intermediate hops. Hence, routing protocols play a vital role in the quality and reliability of data delivery in such networks. The low capacity and lack of stability in ad hoc networks has been one of the main barriers in reliability of them in realistic scenarios. Mobile Ad hoc Networks (MANETs) are one of the main categories of ad hoc networks which support mobility.

Cognitive Radio (CR) was introduced as a concept to provide a promising solution to the problem of spectrum scarcity by enabling unlicensed users to sense and intelligently access the unoccupied spectrum bands. In the recent years, multi-hop wireless ad hoc networks which operate in the unlicensed spectrum bands have been the subject of interest by the research community. However, the limited capacity of multi-hop wireless networks has always slowed the research community from exploring the full potential of these networks. With the introduction of CR, a new framework was introduced with the potential of targeting the spectrum scarcity and consequently a solution to the limited capacity of multi-hop wireless networks. The introduction of CR brought new interesting areas of research centred on dynamic spectrum management and mobility.

By introduction of CR technology and dynamic spectrum management, a new category of routing protocols was introduced with the capability of utilizing the full potential of this technology in targeting some of the limitations in wireless ad hoc networks. One of the areas which was introduced under this branch was Spectrum-aware routing. The name spectrum-aware routing is given to a category of routing protocols which are designed to intelligently utilize the Spectrum Opportunities (SOPs) introduced by the concepts of CR technology. Addition of SOPs to the routing problem in the classical ad hoc networks resulted more complexity in the original problem which could not be targeted by the classical route computation techniques. Hence, the conventional network topologies that was modelled by classical graph has now transformed into a multi-graph with added route computation complexities. Some of the
main challenges associated with spectrum-aware routing are, reliable signalling, deafness problem, coupled and decoupled approaches, routing metrics, control channel and load balancing. A thorough discussion on all of these challenges have been given in Chapter 2.

One of the bottlenecks in the performance of ad hoc networks is the lack of any load balancing mechanism. Due to the self-organizing nature of ad hoc networks, a load balancing mechanism which results maximization in efficiency of such networks is vitally important. With addition of potential routing opportunities introduced by CR technology, a load balanced routing protocol which is capable factoring in utilization of SOPs into the load balancing mechanism can lead to significant performance gains. One of the most well-known load balancing algorithms in multi-hop networks is backpressure algorithm which was originally designed based on a TDMA (Time Division Multiple Access) MAC (Medium Access Control) layer structure. This algorithm performs load balancing in multi-hop networks based on congestion gradients and is proven to be throughput optimal. Most researches involving the conventional ad hoc networks and CR-MANETs are based on the assumption of IEEE802.11 as the MAC layer protocol. Due to the CSMA (Carrier Sense Multiple Access) structure of this protocol, applying backpressure’s queue gradients is known to be a challenging area of research. Under the most efficient implementations, due to the CSMA structure of IEEE802.11, backpressure algorithm does not necessarily support the original algorithm’s throughput optimality. One of the major problems in applying backpressure algorithm in a fully functional routing protocol targeting CR-MANETs is synchronization of queue updates so that nodes are supplied with accurate information. The out of date information and inaccuracy of queue related information can lead to instability in backpressure-based algorithms which affects the end-to-end performance of the network.

Quantum game theory provides a framework to utilize entangled particles with the aim of affecting decision-making process of distant players without transmission of any information. This is enabled by the properties of entangled particles which create a communication-less instantaneous channel that can be used to influence the strategies made by the players in a quantum game. One of the properties of a quantum system is the larger accessible space of states which can be utilized to maximize a pre-defined utility function. Hence, quantum game theory has the potential of targeting the load balancing problem in CR-MANETs from a completely different perspective to that of backpressure algorithm. The potential of quantum game theory in targeting load balancing in ad hoc networks lies on the nearly perfect synchronization capabilities of entangled particles.

### 1.2 Motivations, Novelty and Challenges

Most of the existing spectrum-aware routing protocols that are proposed to target CR-MANETs perform their signalling and route computation on a reactive basis which exhibits unsatisfactory performance given the dynamic structure of such networks. As reactive routing protocols perform their
signalling only when there is request for data transmission, they can result high and unstable end-to-end delay performance which is not in line with the QoS requirement of the applications running by end users. The dynamic structure of MANETs and the complexity resulted by addition of SOPs to this problem requires a proactive routing protocol which performs route computation and optimization on a proactive basis. Under a proactive routing scenario where nodes are updated with the full network topology, the routing algorithm can utilize all the SOPs in guaranteeing an end-to-end QoS performance which is not possible using a reactive routing protocol. On the other hand, integration of a link quality estimation metric and analysing its performance in a proactive routing protocol to prioritize utilization of some SOPs over others to improve the QoS supported by end-to-end paths is another motivation of the first contribution in this thesis. With the assumption of a proactive routing protocol, the route computation problem can only be modelled as multigraphs which to the best of our knowledge, the solutions given to this problem in the literature do not have a graph theoretical backbone in support of their algorithms. The majority of the well-known spectrum-aware routing protocols (in the literature) target the SOP allocation problem on a micro level which results instability on the macro level QoS support. One of the motivations in the first contribution of this research was to target the performance of the spectrum-aware routing protocols on an end-to-end basis and optimize the problem in order to provide routes with higher stability.

As it was detailed before, lack of load balancing capability in majority of routing protocols targeting CR-MANETs is one of the main reasons for instability of computed routes and unsatisfactory end-to-end performance in them. This was the motivation for the second contribution of this thesis. In order to maximize the end-to-end network performance using the spectrum opportunities resulted by CR technology, a spectrum-aware routing algorithm must distribute the network traffic over all the available routes while taking full potential of SOPs into account. The idea of traffic distribution using backpressure algorithm which results maximization in utilization of SOPs in CR-MANETs is the novelty in the second contribution of this work. As it was detailed in Section 1.1, due to the TDMA MAC layer assumption in the original idea of backpressure algorithm, implementation of it under the CSMA structure of IEEE 802.11 is a challenging area of research. On the other hand, spectrum-aware routing algorithms utilize a link quality estimation metric to prioritize routing paths against each other. However, the routes computed based on the backpressure algorithm might be in conflict with the ones computed based on the spectrum-aware quality metric. Hence, another challenge identified in this area of research is to provide a solution to this routing conflict problem.

One of the main problems in load balancing performed by backpressure algorithm in CR-MANETs is the inaccuracy of queue and topology related information. This inaccuracy is resulted by the fact that in a proactive routing protocol the queue related data is distributed proactively on a periodic basis and given the dynamic structure of the CR-MANETs, at times this information could be out of date. The faulty queue information leads to instability in the backpressure algorithm which results unsatisfactory
performance in the computed routes. The motivation in the third contribution of this thesis was to target load balancing from a new perspective which does not rely on the synchronization and accuracy of the queue updates in the network. Towards this aim, quantum game theory was reported to provide a good framework to target load balancing on a fundamental level. Quantum game theory and utilization of entangled particles creates the mean requirement for manipulating distant players of a game with the aim of maximizing a predefined utility function. Utilizing the synchronization properties of quantum entanglement to perform load balancing in ad hoc networks is the novelty in the third contribution of this thesis. Modelling the problem of load balancing using quantum game theory requires the system to be in violation of the bell inequalities (refer to Section 2.12) which is considered as one of the main challenges targeted by this work. Another challenge in this work is to model load balancing as a quantum game and design strategies which players can take to maximize the predefined utility function.

1.3 Contributions

There are three major contributions provided in this thesis which are categorized as below.

i. As the first contribution in this thesis, we have proposed a novel spectrum-aware routing protocol with OLSR (Optimised Link State Routing) as the baseline of our implementation so called OLSR-SA (OLSR- Spectrum Aware). The proposed routing protocol is capable of signalling the channel related spectrum opportunities to the nodes across the network. Furthermore, ETX (Expected Transmission count) is used as the link quality estimation metric to evaluate suitability of SOPs to be used in the end-to-end routing paths. The generalization of Dijkstra’s algorithm is reported as a solution to best weight route computations in multigraphs. Furthermore, this algorithm is implemented and integrated with the baseline OLSR and its functionality is successfully confirmed. The performance gain resulted by the first contribution of this work has been confirmed through extensive simulation studies.

ii. The second contribution of this thesis is built upon the first contribution and in order to target the inefficiencies resulted by the shortcomings of it. Towards this aim, a novel load balancing mechanism based on backpressure algorithm is proposed. As the algorithm requires queue related information to perform load balancing based on traffic gradients, a backpressure signalling mechanism was proposed. The backpressure algorithm is unified with the dynamic spectrum structure of CR-MANETs by incorporating SOPs into the load balancing decision making process; this results stability in terms of end-to-end delays and packet delivery ratio. The final OLSR backpressure spectrum-aware (OLSR-BSA) routing algorithm not only optimizes route computation based on the predefined cost metric but also incorporates the queue gradients of backpressure algorithm to perform load balancing. By extensive
simulation study we have confirmed that the algorithm outperforms the baseline OLSR and OLSR-SA in terms of end-to-end delay and packet delivery ratio.

iii. The third contribution in this thesis is based on the interdisciplinary idea of quantum game theory. It was identified that load balancing in backpressure algorithm requires up to date and accurate queue information which given the proactive structure of our routing protocol, achievement of this accuracy is infeasible. On the other hand, lack of synchronization in the queue information that is distributed in the network results instability in the computed routes by the OLSR-BSA algorithm. In the third contribution of this thesis, we show that, synchronization of entangled particles in a quantum game can be used to affect the decision-making process of distant players, without transmission of any information. This enables us to formulate the problem of load balancing in ad hoc networks using the novel concept of quantum game theory. In this work, we have formulated the utility function of the quantum games tailored to the problem of load balancing in ad hoc networks. The Quantum Load Balanced OLSR algorithm (QLB-OLSR) was implemented in the simulation environment and it is shown that the proposed algorithm provides stability in throughput, end-to-end delay and jitter performance compared to the baseline OLSR algorithm.

1.4 Thesis Structure

The organization for the rest of this thesis is as follows. In Chapter 2, a literature review on the main subjects of this thesis is provided. Chapter 3 summarizes the first contribution of this thesis which is the proposed spectrum-aware routing algorithm. Chapter 4 concentrates on the second contribution of this thesis which is centred on load balancing based on backpressure algorithm. Next, Chapter 5 provides the last contribution of this work which is load balancing based on the idea of quantum game theory. Finally, Chapter 6 summarizes this thesis and provides the future work direction based on the contributions listed.
Chapter 2

2 Background Study and Related Research

2.1 Cognitive Radio

CR was introduced in [1] as a concept to provide a promising solution to the scarce spectrum resource problem by enabling unlicensed users to sense and intelligently access the unoccupied spectrum bands [2]. Spectrum is a limited resource and with growth of the current communication technologies and inefficiency in allocation of spectrum bands, this valuable resource has remained underutilized to this date. Based on the report by FCC (Federal Communications Commission) Spectrum Efficiency Working Group, the licenced frequency bands are often inefficiently utilized, resulting in SOP with availability fluctuating in temporal, special or code domains. Other measurements-based studies e.g.[3] point out that for example, based on empirical data gathered during period 2004-2005 by the SSC (Shared Spectrum Company) on average only 5.2% of the spectrum band between 30 MHz and 3 GHz is in use at any time, based on data gathered from six different locations in USA.

Inefficiency in the utilization of valuable and limited spectrum bands drew the attention of the research community to re-think the spectrum allocation paradigm. Concepts such as spectrum management, Spectrum mobility and spectrum sharing have been studied and evaluated [4]. CR networks picture a situation where two groups of network users coexist, so called secondary and primary users. PU (Primary User) is the licenced user of each spectrum band and must be able to operate in that band without any unwanted interference from SU (Secondary User). SUs consider the underutilization of the spectrum by PUs as an opportunity and utilize the available SOPs with the condition that no harmful interference is caused on PUs. As a result, SUs can improve their performance by switching among SOPs (although switching delays are a known issue) and in more advanced scenarios, SUs may relay PU traffic resulting is substantial increases in overall system throughput/capacity.

In CR, management of SOPs can be achieved within the so-called Software Defined Radio (SDRs). With the aid of SDRs a CR system can sense a wide range of spectrum bands and dynamically switch among them based on different RATs (Radio Access Technologies). The performance of a CR system is maximized when the system can interact with all layers of OSI (Open Systems Interconnection) reference model. The work of [3] states that CR has implications on all layers of the OSI model. Hence
many of the SOP related decisions should be made based on a cross-layer paradigm which is further illustrated in Figure 2-1.

![Diagram of Cognitive Radio and the necessity of Cross Layering](image)

**Figure 2-1: Cognitive Radio and the necessity of Cross Layering**

A typical communication system comprises of distinctive logical layers acting as encapsulated modules which pass on messages on a layer by layer basis. With the introduction of CR technology, the conventionally strict OSI layering approach has been under some debate in the research community. Different works have argued that a cross-layer design is necessary to realize the idea of CR technology and end up with an optimal solution to end-to-end communication [5, 6]. Furthermore the example-based proof in [7] argues that a cross-layer design is necessary to target dynamic spectrum systems. The work of [8] also considers that there is a joint correlation between dynamic frequency assignment and routing (Network layer) in CR wireless networks which is another argument proving that cross layering is vital for CR networks. The reason behind validity of cross layering in a CR environment is that without having direct feedback from other layers it is almost impossible to find an optimal solution to the problem of when, where and how the spectrum should be utilized and to subsequent selection of best paths at the routing layer, given the variability in resource/spectrum availability to SUs.

### 2.2 Correlation of Spectrum with routing

As shown in Figure 2-1, the concept of Cognitive Radio does not add a horizontal layer to the OSI reference model but rather a vertical parallel layer. The main reason behind it is that Cognitive Radio
comes from the word ‘cognition’ which is learning through experience and cannot be achieved without sufficient direct information from other OSI layers. By sensing the environment CR nodes can detect the variations of the communication resource availability in specific locations and by learning of the changes, to respond and serve the needs of the user in the best and most efficient way.

One of the early major contributions toward enabling CR technology was through modification to the lower OSI layers (MAC and Physical) [9, 10]; the modification was aimed at changing the fixed spectrum allocation structure toward a dynamic structure. While this was one of the major steps toward creation of Dynamic Spectrum Access (DSA) [2] framework, it was not solely sufficient. Lack of a cooperative mechanism toward management of spectrum in the nodes of a communication network was still a major technical problem. Availability of the spectrum as a communication resource is sensed by the Physical layer and partially managed by the MAC layer (e.g. IEEE 802.11). Cooperation among MAC and Physical layers is an accepted paradigm which takes place in many communication protocols to provide a hop by hop connection among nodes; by this argument, MAC and Physical layers are not capable of providing an end-to-end connectivity in a multi-hop communication network. Since CR networks mostly focus on multi-hop scenarios, without cooperation, while efficiency in utilization of spectrum can be achieved in the vicinity of any individual node, the optimal end to end performance cannot be guaranteed. Layer three of the OSI reference model which is the network layer is responsible for providing end-to-end connectivity with the aid of routing protocols. Network layer is where connections of all nodes are managed and basically the network topology is created. Routing prior to DSA systems relied on the fact that there is a frequency band/channel available for every individual node/link at a time and this is managed by lower layer/MAC protocols. CR technology necessitated the introduction of a new concept which could enable nodes to switch among the spectrum opportunities and hence created a more stable communication framework whilst minimizing interference to primary users; this direction opened up two options, either changing the whole structure of the OSI layers [11] (for such dynamic-spectrum environment) or moving toward a cross layering solution among network and lower layers [4]. Since introduction of any communication framework or protocol necessitates adaptation to the currently deployed systems, the research community has paid more attention to cooperation between network and Physical/MAC layer (aka. cross layering). This was the birth place of a new category of routing protocols called, Spectrum-aware routing protocols. A spectrum-aware routing protocol must be aware of the spectrum opportunities (through cross layering or other means) and performs channel switching on a hop-by-hop basis for forwarding the data to a given destination whilst maintaining path stability and ensuring minimal delays.

### 2.2.1 Fundamentals of Cognitive Radio

It is elaborated that the main functionalities of Cognitive Radios can be grouped into the following.
Spectrum Sensing: One of the primary requirements of Cognitive Radios is to detect the presence of Primary Users (Licensed Users) through sensing techniques.

Spectrum Management: Management of SOPs based on the user’s QoS requirement is another main goal of CRs. SOPs are to be analysed based on a mechanism and groups into classes of service.

Spectrum Mobility: SOP switching or seamless transition among spectrum bands is a vital task in CRs. Switching takes place either when PU activity is detected or when the current SOP does not satisfy the minimum QoS requirement of the SU.

Spectrum Sharing: There exist multiple alternatives for arranging the secondary access. This is well analysed on the work presented by A. M. Wyglinski [12]:

CR technology was introduced on the basis that channel is a dynamic resource which can be shared efficiently amongst communication users. The works in [13] [14] [15] provided a strong foundation to the concept of cognitive channel. The three spectrum sharing models analysed in the work of [12] are Underlay, Interweave and Overlay Spectrum sharing models [15]. In the underlay spectrum sharing, both primary and secondary users can coexist without any necessary PU or SU detection mechanism [16] [17]. In the Interweave spectrum sharing model the assumption is that the PU usage of the spectrum bands can be monitored and modelled for the SUs cognitive usage. The research community has argued that this can be modelled using an ON-OFF model as elaborated in [18] [14]. Interweave spectrum sharing model relies on sharing the spectrum on the basis of time-occupancy level. On the other hand, the overlay spectrum sharing model relies on full integration of SUs with PUs spectrum and technology architecture [13] [19]. Hence, the overlay model requires tremendous change in the currently deployed systems in order to allow SU usage of the spectrum.

With analysing the above categorizations of the main CR functionalities, we can see that Spectrum Management, Spectrum Mobility and Spectrum Sharing can all be implemented under the layer three functionalities which is Network Layer. Though cross layering, the SOP information can be pulled up to network layer to be managed, switched or shared among SUs. The reason why research pushes these functionalities into the network layer is mainly due to the routing protocols that are based in the network layer. The way a routing protocol manages the connectivity, if the SOP information is pulled up to this layer, we can achieve the best end to end paths based on the performance metrics of our choice.

2.3 Wireless Ad hoc Networks

Wireless ad hoc networks are a category of decentralized infrastructure-less networks that operate based on a self-organizing paradigm. When a group of nodes, equipped with wireless interfaces dynamically connect with one another in an infrastructure-less manner [20], an ad hoc network is
formed. Ad hoc networks do not rely on a fixed infrastructure, which makes them ideal for many applications such as emergency services and Tactical/military operations. In wireless ad hoc networks, nodes communicate using wireless interfaces via the communication medium (by means of spectrum/channels). One of the well-known communication standards that supports infrastructure-less mode of operation is IEEE 802.11. Due to the nature of wireless communication and limitations of currently deployed MAC (Medium Access Control) protocols [21, 22], the challenges involved with ad hoc networks are significantly more than infrastructure-based wireless networks. Due to lack of infrastructure in ad hoc networks, nodes within the network are responsible for forwarding data packets from source to destinations via other ad hoc nodes. The process of delivering data packets from source to destination via other intermediate nodes is so called routing. Routing is known to be one of the main challenges in ad hoc networks. The reason being, while discovered routes in ad hoc networks need to maintain certain QoS criteria by efficient utilization of network resources but they are not allowed to exhaust network resources in this process. There are various limitations associated with ad hoc networks which are all listed in this section. To be able to direct the focus of the literature review provided in this thesis we have assumed the MAC/PHY layer to be IEEE 802.11. Hence all the limitations associated with ad hoc networks are based on the assumption that the MAC/PHY layer is using this protocol.

2.3.1 Excessive Noise and Interference

Interference and noise are two main factors affecting the quality of routes created in ad hoc networks; this result lower data rate mainly in the ad hoc networks that utilize IEEE 802.11 DCF (Distributed Coordination Function) as the communication protocol for their wireless interface. IEEE 802.11 is one of the well-known protocols which has been studied extensively in various researches involving multi-hop ad hoc networks. IEEE 802.11 assumes that each channel can only be accessed by one node and if two or more nodes try to access the channel at the same time, collision takes place. The sensing mechanism designed in IEEE 802.11 in order to avoid collision is CSMA/CA [23]. In CSMA/CA when a collision takes place, the colliding nodes back-off from their transmission based on a random time and re-attempt their transmission after a specified time. The main source of low data rate and capacity in ad hoc networks using IEEE 802.11 is this structure of CSMA/CA. Each collision results a subsequent back-off delay resulting higher end-to-end delay and lower throughput in the network. The dynamic structure of ad hoc networks utilizing this technology results excessive collision.

2.3.2 Dynamic topology

Another problem of ad hoc networks is their dynamic structure. MANETs are a type of ad hoc networks with support for mobility. Mobility results in excessive topology changes which results
excessive unpredictable collision and is considered as another performance barrier. MANETs are further explained in Section 2.4.

2.3.3 Security

Security is another important challenge in ad hoc networks. The infrastructure-less design of ad hoc networks makes it prone to security issues. There has been various surveys performed in this area which lists some of the most important security threats in ad hoc networks [24, 25] and more specifically routing in ad hoc networks [26]. Security in ad hoc networks is to ensure that malicious nodes are not able to starve resources in the network due to personal interests. Furthermore, data communication between the ad hoc nodes should stay encrypted and only readable by the intended destination. Like any other communication network security plays very important role in ad hoc networks and many researches have studied it in detail.

2.3.4 Load Balancing

Another factor that affects the performance of ad hoc networks is the lack of any load balancing mechanism. The decentralized structure of ad hoc networks makes load balancing a complicated problem. Most of the routing protocols designed for ad hoc networks do not take load balancing into account which results uneven and unstable QoS performance across the network. Conventionally, routing protocols designed for ad hoc networks utilize a cost/quality metric to assess the performance of routes in the network. Most of the ad hoc routing protocols such as AODV (Ad hoc On-Demand Distance Vector), OLSR, DSDV (Destination-Sequenced Distance-Vector Routing) and DSR (Dynamic Source Routing protocol) use hop count as the quality metric [27]. Hop count has shown many shortcomings in the past studies which is the reason why shortest paths (resulted from minimization of hop count) in ad hoc networks would not necessarily guarantee routes with the best QoS [28]. One of the major problems with hop count or many other metrics listed in [28] is that they encourage overutilization of a single route without considering congestion and buffer overflow problems; this results some segments of the network being heavily loaded which creates bottlenecks and as a result the overall QoS performance of the network is affected. Unbalanced load distribution results congestion of heavily loaded nodes, buffer overflow and finally increased end-to-end delay in the network [29]. Overly utilized paths cause exhaustion of network resources such as power, bandwidth and memory. Hence load balancing is one of the major issues in ad hoc networks and is the topic of some of the contributions in this thesis. Load balancing is further expanded in Section 2.10.
2.4 Cognitive Radio and Mobile Ad-hoc Networks (MANETs)

As elaborated before, ad hoc networks are a subset of wireless networks which follow an infrastructure-less paradigm. They have no central control or management mechanism and communications occur based on a self-configuring distributed design. The same limitations that are applicable to other wireless networks also apply to ad hoc networks such as bandwidth limitations, power control, coverage and etc. As no infrastructure exists in ad-hoc networks, they rely on decentralized resource access and hop by hop relaying of the data for end-to-end connectivity. Throughput of ad-hoc networks is considerably less than other wireless networks due to the requirement of hop-by-hop mechanism in relaying of the data. Consequently, their stability in terms of throughput and end-to-end delay decreases as the number of end-to-end hop along the path increases. The performance of IEEE 802.11 in ad-hoc mode is well analysed in the work of [30]. They have elaborated that the end to end performance of ad-hoc network cannot exceed a certain limit even under the best communicational conditions. MANETs follow the same structure of ad-hoc networks with the difference that nodes support mobility. Mobility in MANETs puts ad-hoc networks under worse conditions compared to the conventional ad-hoc networks. As shown in Figure 2-2, limitations of MANETs can be summarized as, highly dynamic topology, low throughput, low security, high data loss rate, higher end-to-end delay, limited power. Nevertheless, due to the infrastructure-less design, ad-hoc networks are one of the most favourable ones when it comes to self-configuring networks such as CRN (Cognitive Radio Network).

![Figure 2-2: Characteristics of MANETs](image)

One of the most commonly used MAC/PHY standards is the IEEE 802.11 standard [31]. This standard can both function under infrastructure and infrastructure-less (ad-hoc) modes. The newer generation of IEEE 802.11 wireless LAN standard is capable of bandwidth aggregation with utilizing multiple non-overlapping frequencies [32]. This capability of IEEE 802.11 is only available in the infrastructure mode operation and is not designed for infrastructure-less case operating in ad-hoc mode. The research community has taken vast interest in changing this structure so that multi-hop ad-hoc mode can also take advantage of all the 11 channels available to the 802.11 standard. Most of the research in
the area of cognitive radio considers it under an ad-hoc network structure. Hence IEEE 802.11 in ad-hoc mode provides a very promising testbed to analyse capability of this idea and vast researches in the area of cognitive radio base their simulation studies on 802.11 standard. Toward this goal, there are different architectures designed to support multi-channel 802.11 in ad-hoc mode. The approaches provided in [32] and [33] targets channel allocation by defining multi-channel through multiple NICs (Network Interface Cards). With the idea of multiple NICs per node, one of the fundamental issues raised, is to decide that which channel (aka NIC) should route the data. One of the most logical ways to target this problem is to involve network layer in this decision-making process and manage the usage of each channel and interface from third layer of the OSI model. This cross-layering approach can only be achieved through a new routing protocol that is aware of the SOPs to manage the traffic through less occupied (in terms of channel usage) links/paths and provides maximum connectivity in an infrastructure-less ad-hoc network.

2.5 Dynamic Spectrum Utilization in CR-based Multi-Hop Ad-hoc Networks

As explained in the previous sections, CR brought a new area into focus which pushed the boundaries of research on ad hoc networks even further. Suddenly focus of research community has turned into designing an appropriate frequency selection mechanism that allows unlicensed SUs to efficiently utilize available SOPs while avoiding interference with PUs. Currently the licensed spectrum bands are only accessible by the users licensed by their service provider to use them. To the best of our knowledge there has not been implemented any globally used standard to opportunistically share the spectrum bands among the licensed PUs and unlicensed SUs. The idea is to impose a dynamic paradigm for spectrum access and with that trend, increase the spectrum efficiency by saturating each spectrum band. The two common unlicensed categories of spectrum bands are ISM (Industrial Scientific and Medical) and U-NII (Unlicensed National Information Infrastructure). 802.11a, b, g and n are the standards made by IEEE (Institute of Electrical and Electronics Engineers) which operate on these bands. Since the number of non-overlapping spectrum bands in these set of standards (802.11a, b, g and n) is limited, as size of the network is increased and populated with users, the performance is highly degraded. Design and implementation of a routing protocol tailored for CR-MANETS needs presumption about the underlying MAC/PHY layer protocol. Due to the interactions required between the network and MAC/PHY layers, it is almost impossible to design a routing protocol independent of them. It will be demonstrated in Section 2.8 that many of the current researches in the area of spectrum management in CR consider IEEE 802.11 as the base of MAC/PHY layers.
2.5.1 DSA Spectrum Management, Categorizations and Challenges

There are diverse ways that DSA categorises the spectrum bands and their management in CRNs. Figure 2-3 shows a high-level taxonomy of spectrum management in DSA which is the focus of this section.

![Spectrum Management Taxonomy]

**Figure 2-3: Taxonomy of Spectrum Management Architectures**

### 2.5.1.1 Dynamic Spectrum Access Models

The work of [34] categorises the DSA into the three categories of Dynamic Exclusive Use Model, Open Sharing Model and Hierarchical Access Model. Open Sharing Model is the scheme implied on the unlicensed range of spectrum bands. Two main categories of unlicensed spectrum bands are ISM and U-NII. The 802.11a, b, g, n and ac, are the standards by IEEE which operate on these bands [13].

Hierarchical Access Model has two sub-categories of Spectrum Overlay and Spectrum Underlay. In the Hierarchical Access Model, sharing users of the spectrum in the network are divided into two types of PUs and SUs. As elaborated in previous sections PUs are the licensed band of the spectrum in the network and SUs are the opportunistic users that try to use the spectrum while primaries are not sensed in the area. SUs should try to impose the least interference on PUs at all time. According to the work of [10] this scheme is called underlay. Based on this scheme SUs do not need to sense the existence of primaries, instead by keeping low interference level during the whole period of transmission, they can achieve a short-range high data rate with extremely low transmission power [10]. On the other hand, spectrum overlay does not impose any restriction on the transmission power of primary users, but instead it controls the time and location of transmission. With exploitation of sensing techniques, secondary users sense the existence of primaries and transmission only takes place where PUs are not detected in the vicinity.
Most of the Spectrum-aware routing protocols currently under research which will be explained in Section 2.8 use the spectrum bands categorized under Open Sharing Model but follow the architecture of Spectrum Overlay (categorized under Hierarchical Access Model).

### 2.5.1.2 Spectrum Management Architecture

There are generally three approaches in which SOPs can be managed as shown in Figure 2-4; these are Centralized, Distributed and Hybrid. The spectrum management entity has the duty of collecting geographically indexed SOP information through nodes (or by other means) in addition to disseminating this information to any number of nodes in the network upon request.

![Spectrum Management Architectures](image)

The distributed model lacks any central managing database hence, nodes cooperate in the distribution of SOP information and compete over utilization of available ones. The hybrid model is a mixture of the former approaches meaning that nodes not only cooperate in discovery of SOPs but also send this information to the central database entity. Nodes in hybrid model can acquire SOP information both based on local and global knowledge. Of course, the local knowledge of the topology and spectrum availability is not as accurate as the spectrum broker and there is a trade-off involved. Looking from the routing protocol point of view, accessing a central spectrum broker by all the nodes across the network causes heavy signalling load which is a waste of bandwidth. On the other hand, relying on the local knowledge may not be sufficiently accurate to find efficient routes. In conclusion, a spectrum aware routing protocol should find a balance between the three approaches and manage the path selection efficiently.

### 2.5.1.3 Multi-Channel, Multi-interface solution

Multi-channel networks with either single or multiple transceivers (interfaces) are an active topic of research. The work of [35] argues that with the low cost of the 802.11 interfaces in the market, it is more logical to equip nodes with multiple interfaces rather than having one interface and increase the number of switching among different channels. One of the changes that devices should undertake toward evolution of dynamic spectrum paradigm is having multiple interfaces. Extensive research has targeted
the multi-channel multi-interface model [36-40]. With the introduction of dynamic spectrum systems, management of channels (SOPs) has been one of the main challenges. It is arguable that whether one interface is enough under dynamic spectrum systems or since we have a dynamic spectrum structure more than one is necessary. One of the solutions given to the problem of channel management is to use multiple interfaces and assign each channel to an interface based on an appropriate algorithm. Furthermore, interface assignment can be divided into two categories of static or dynamic; Static meaning that the interface is assigned to a channel for the whole period of time and dynamic meaning that different channels are assigned to an interface at different periods of time. As shown in Figure 2-5, each of these categories is further subdivided into centralized and distributed.

Centralized allocation scheme, meaning that a central entity has a global knowledge of the network and it is that which decides about the best mapping and scheduling among channels and interfaces. Distributed allocation scheme interpreted as channel to interface mapping is handled locally and without the provision of global information; it is worth noting that static-distributed interface assignment can be achieved based on a statistical pre-set function but dynamic-distributed interface assignment is based on cooperation among network and lower layers of all devices to find the best match for channel and interface assignment.

The research community considers a few assumptions to simplify analyses on multi-channel multi-interface devices. For example, most of the algorithms proposed for multi-interface devices assume that availability of channel is independent of location and time so basically all channels are available at all periods of time. With this assumption, the interface assignment can only focus on efficiency of channel allocation and does not consider the main factor which is availability. While this can be a valid assumption for 802.11 sets of standards, it does not stand true for all DSA network environments. There are other works in this area which try to match a multi-channel environment with the currently deployed single interface devices. In a DSA environment where there are a large number of SOPs and their availability is time and location dependent, this assumption necessitates channel switching within the interface very frequently to avoid conflicts among different channels which results poor performance or
in situations where the mobility in the network is high it is almost impossible. Another method is to consider channel switching at packet granularity without taking into account the cost accompanied by channel switching delay.

One of the factors that many researches in this area have not considered is the practical limitation of the wireless interfaces, which is the fact that two interfaces in two different nodes should be capable of simultaneously switching to a mutual channel to be able to communicate with one another. Hence due to the hardware limitations, a device cannot send data on one channel and listen for incoming data on all other channels simultaneously. The work presented in [41] refers to this phenomenon as deafness problem. Even though, assumptions such as the ones explained above make the system simpler to analyse but it needs to be addressed in the real-world application of CR technology.

### 2.6 Spectrum Aware Routing

In previous sections we explained the connection between network layer and more specifically routing protocols of the network layer to MAC and PHY layers. It was elaborated that without a higher layer view of the connectivity across the network, end-to-end route quality cannot be guaranteed. Network layer is where connections of all nodes in the network are managed and basically the network topology is created/maintained. Routing prior to DSA systems relied on the fact there is only one channel/frequency band available for all nodes/links to use at any given time and this is reliably managed by lower layer protocols. In DSA systems creation of wireless communication link is still the responsibility of lower layers with the difference that there might be multiple choices of wireless channels available at different times and geographical locations. Routing protocols are capable of monitoring the whole network topology and nodes through network layer signalling and are in fact the best candidates for channel management. Toward this aim, a routing protocol should disseminate information on channel allocation among the nodes within the network and continuously update the status of each channel within the network. Hence spectrum availability information is added as a new factor into the decision-making process of the routing algorithm and plays a very important role in creation & maintenance of the network topology in CR-MANETs. Consequently, the shortest paths in the network may not always result in the best paths since reliability of the routes is a new important factor. A new routing protocol should be capable of updating the current state of each and every SOP through the network and utilize them based on different QoS requirement of routing streams. Furthermore, a new protocol should be capable of utilizing different DSA techniques to allow the use of best available channel/channel-sets for routing the data.

It is worth noting that there is a major difference between the ideas of spectrum-aware and QoS-aware routing protocols applied to MANETs. The major difference arises from the fact that QoS-aware routing protocols [42] assume that each and every link in the network operates on one channel but
Chapter 2: Background Study and Related Research

Spectrum-aware routing protocols are capable of utilizing multiple channels (SOPs) along the route between any source destination pair. While the term, “a spectrum-aware routing protocol can also be QoS-aware” makes complete sense but it does not make any sense the other way around.

2.7 Challenges in Spectrum-aware Routing

Maximizing efficient utilization of SOPs and minimization of the interference and delay are the main goals in designing any spectrum-aware routing protocol. With introduction of the trend toward spectrum-aware routing, many new challenges opened up to the research community. Balance distribution of SOPs among nodes in the network is one of the main concerns in spectrum-aware routing. The connectivity of a DSA system depends on SOPs and the result of an unfair distribution of them is disconnection of the network and high rate of data loss. The decision on whether a node should broadcast the spectrum-aware routing data on all of its channels or specifically decide to send the data on one channel toward destination is another important factor. The approaches used in evaluation of availability and quality of available channels is a very important issue. The area of spectrum-aware routing opened up discussion on these areas to target these challenges. In this section, we have tried to analyse the main challenges in the area of spectrum-aware routing and provide possible solutions to them.

2.7.1 Control Channel

One of the problems encountered in the area of routing within a DSA systems is to provide a reliable control channel. While data can be routed through SOPs with the aid of an opportunistic spectrum allocation mechanism and routing, it is not reliable to use SOPs for exchange of control information based on two reasons:

1. At the beginning of topology creation existence of a reliable spectrum band (aka. channel) is essential for exchange of routing information. The opportunistic nature of SOPs makes their reliability as a signalling medium questionable. As a routing protocol relies on signalling for topology discovery and if under any circumstances the signalling channel undergoes interference, given the dynamic structure of CR-MANETs, the data loss would be unbearable.
2. At any time during the data transmission, a PU can appear among SUs; consequently, link breakage might happen frequently. Hence if signalling is dependent on the SOPs, there might be periods that no signalling data can go through the network.

Choice of a common control channel to exchange spectrum related data at beginning of topology creation in a spectrum aware routing protocol is of high importance. Signalling which carries the topology and SOP related data should be performed uninterruptedly independent of channel availability.
In conclusion, it is highly recommended that design of a new spectrum-aware routing protocol be based on a fixed control channel.

### 2.7.2 Coupled and decoupled route selection and spectrum management

An important challenge in the area of DSA is the question of whether the spectrum selection should be integrated with the route selection or they are two entities operating on two different layers of OSI model.

The work of [43] looks at the benefits and costs of both perspectives of coupled and decoupled design. Decoupled design focuses on hop-by-hop relaying case of routing which spectrum selection and route discovery are decoupled into two distinct entities without any interaction among them. Basically, a node that uses this approach considers only the local information of spectrum and applies that into the route discovery without any knowledge of the next hops in the forwarding process. Logically, hop by hop relaying of the data in DSA systems is not sufficient to provide an end-to-end performance gain; this has been demonstrated in the example given in Figure 2-6. Let us assume Node-S (source) is willing to relay its data to Node-D (destination). Each link along the path might be on different channels hence having different costs. Let us assume two cases, first when the data is relayed over \{2, 3, 4, 5\} and second where the data is relayed over \{9, 8\}. The first scenario is the result of hop-by-hop relaying of the data based on local SOP information which is categorized as decoupled design.

![Figure 2-6: An example for coupled and decoupled design of DSA systems](image)

As shown in Figure 2-6, since the first hop 1-2 has lower cost compared to 1-9, it is chosen for relaying the data regardless of other nodes along the path. Consequently, the data is relayed over a route with higher end-to-end cost. On the other hand, in the second case, as spectrum decision making process is integrated in network layer (coupled design) and given that network layer has information from all
nodes of the network, it can guarantee end-to-end performance; this is achieved via the information provided at network layer and used toward minimization of end-to-end cost regardless of the cost of first hop (1-2 or 1-9).

Obviously, routes found by decoupled approach suffer from unreliability, as hop by hop routing does not guarantee an end-to-end benefit. In general, the task of scheduling (aka spectrum management) is the responsibility of MAC layer. Due to the nature of radio links, transmission on a channel can be heard by all the proximate devices operating on that channel resulting interference. Based on this, since a decoupled design doesn’t have any global knowledge of channel usage and does not broadcast any information on the local channel availability, it may suffer from high interference.

On the other hand, a coupled design relies on the collaboration of network and lower layers (MAC and PHY) to integrate the spectrum management into route selection. In this approach, if a source node needs to reach a destination, not only it should consider the available paths to destination, but also the spectrum heterogeneity along that path to destination. It is through this method that a minimum level of QoS can be guaranteed. Of course, the coupled design comes at the cost of extra singling overhead (generated by network layer) and a fluctuation on the spectrum availability has a direct impact on route assignment. The research community has shown interest in the coupled design in the recent past. Toward this aim, many spectrum aware routing protocols (Section 2.8) are designed based on collaborations of MAC and network layers (cross-layering) to achieve reliability in CR networks.

### 2.7.3 Sensing Techniques and the impact on Routing

Sensing is one of the most important factors in coexistence of PUs and SUs. SUs must continuously sense the signal power level of PUs, otherwise when the interference level of SUs goes above the threshold accepted by the PUs radio technology, the signal is rendered unusable and this interference is unacceptable to primary user of the radio technology. Different communications barriers such as the hidden node problem or shadowing effect can make this issue even worse. Another unreliability issue is demonstrated in the work of [44] in which, the noise power uncertainty can highly degrade the accuracy in detection of signals in low SNR.

Based on the work of [4] secondary users can compete in two distinct ways over free spectrum. Now when we consider sensing in a CR environment, there are two scenarios which can be considered:

- **Licensed band operation**: The SUs of the network should continuously sense the existence of PUs and in the absence of transmissions in a particular spectrum band; then SUs are allowed to utilize that band. Furthermore, if suddenly in the middle of transmission primary signal was sensed nearby, the secondary users should stop the transmission and hand over to another free spectrum band.
• **Unlicensed band operation:** In the complete absence of primary users in a certain geographical area and time, secondary users are allocated with the same priority over the spectrum bands and should compete among themselves for their transmission based on a particular mechanism and standards.

In both of the above cases, sensing plays a very important role in terms of accuracy and delay. Depending on the speed which the available spectrum is sensed, the routing protocol can start calculating the best routes to destinations. However, most researches concerning the spectrum-aware routing consider perfect sensing (Section 2.8) as one of the initial assumptions. While for now, this assumption makes simulations over such networks feasible, it does not hold true in a real communication environments where sensing a spectrum band is associated with a finite delay which has an impact on the performance of a network layer routing.

### 2.7.4 Distribution of SOP information

Sharing of spectrum-availability information in infrastructure-based networks such as CWMN is not as challenging as in an infrastructure-less network such as CRAHNs. Additionally, mobility of the nodes adds to the dynamic nature of resource availability in such networks. In a centralized approach, i.e. a network supported by an infrastructure, sharing the spectrum availability information is managed by a central entity (spectrum broker). Hence nodes would continuously update the spectrum broker’s database and the information can be requested by any other node within the network on demand.

The case is completely different when it comes to decentralized i.e. an infrastructure-less networks. Due to the dynamic nature of the network, there is no permanent link between all the nodes and the spectrum broker; hence it is not feasible to have a central entity within the network and update it regularly. Instead this information can be updated in a cooperative way. A spectrum-aware routing protocol can provide the framework for this cooperation among CR nodes. Such a routing protocol not only uses the spectrum availability information sensed by individual nodes within the network, but also provides a means for transferring this information to other nodes either in a proactive or reactive manner.

### 2.7.5 Spectrum-aware signalling mechanism

Cooperative distribution of channel availability information through all nodes of the network requires a strong signalling mechanism to be designed. Most of existing spectrum-aware routing protocols (explained in Section2.8) use a broadcasting mechanism to update & broadcast the network topology based on spectrum diversity [41]. Since every node in a DSA environment has multiple SOPs available to them, most of the routing protocols designed for such networks follow an inefficient broadcasting mechanism. The reason behind this being that each node tries to update the information of channel availabilities (SOPs) through all the available channels so that the channel availability information
collected at the receiver side is sufficient to add that specific channel and its quality to the SOP list. While this mechanism assures that the SOP information is fully distributed throughout the network, it is not actually an efficient signalling mechanism. The result is massive signalling overhead in the network which could become a source of interference and degrades the performance of the network. The first solution to this problem is to choose a sub-set of the SOPs available to the node and use them to distribute the topology of the network. The other solution which is not in line with the structure of CR networks is to dedicate a channel to signalling purposes and do not consider that channel as a SOP for data traffic communication. The first solution is preferred over the second one due to the fact that by sending signalling information over a sub-set of channels we can actually sense their quality.

### 2.7.6 Switching Delay

Channel switching is necessary in CR networks firstly to avoid unwanted interference with PUs and secondly to achieve better end-to-end performance. Channel switching results in non-negligible delays which are proportional to the step size difference between the frequency of current channel and the one which a node is switching to. This delay has a direct impact on the operation of a spectrum-aware routing protocol. The end-to-end delay and the achievable bandwidth are directly affected by switching delay. Channel switching delay is discussed in [45] where the overall route delay is affected by switching delay; consequently a routing metric is derived with the aim of minimizing cumulative delay. Another work where the channel switching delay is considered is in [46] where in the simulations, this delay is assumed to be fixed.

Generally, any delay in the network is considered as a cost. Switching delay is not excluded from this argument. Whether the cost of switching delay is worth the benefit of gaining better routing performance and the frequency of this switching are open research challenges and partially targeted in the work of [47] but need more in depth analysis.

### 2.8 Survey on Spectrum-aware Routing Protocols

In this section, different state of the art spectrum-aware routing protocols are summarized and short discussion on the benefits and weaknesses of each protocol is provided.

#### 2.8.1 A spectrum-aware routing for multi-hop single transceiver Cognitive Radio networks (MSCRN)

This work [41] focuses on multi-hop single transceiver Cognitive Radio networks (MSCRN). They propose a spectrum aware on-demand routing which does not rely on availability of a single control channel; this is because in cognitive radio availability of a control channel cannot be guaranteed. Their
implementation of routing protocol is based on AODV [48] and similarly uses RREQ (Route Request) packet to initiate route discovery process through the network. It broadcasts the RREQ packets through all the available channels of a node and consecutively the next node does the same until the destination is reached. The destination follows the reverse order to reach the source. Their work has also targeted the deafness problem which results from multiple channel switching requests at one node while a switching node can only handle one switch at a time. To avoid failed packets on a channel and too much contention in a switching node they have used LEAVE/JOIN messages to inform joining and leaving of a channel to the other members of that channel. The drawback of their method is firstly the major signaling overhead that is caused by broadcasting RREQ message through the whole network. Another issue that has not been considered in this work is if there are multiple routing paths to the destination, which path and based on what knowledge are they used on the request reply message. Mobility which is one of the important factors of routing in cognitive radio network is not considered. Another issue that has not been well addressed in this work is the effect of delays caused by multiple switching, leaving/joining a channel, hardware switch and back-off process on performance of the routing protocol from QoS aspects.

2.8.2 SEARCH

Unlike other geographical routing protocols that are unaware of PU activity region and operate on a single channel, SEARCH [49] uses the greedy geographic routing [50] on each individual channel to gain access to the destination. On the return path from destination to source all the information from different channels are utilized to find the most efficient path to the source. This work points out a very important fact to consider about channel switching. For example, when a WLAN device is transmitting in one of the 802.11 channels, the spectral leakage power causes interference on other channels. This result different coverage ranges at different channels. SEARCH is capable of predicting the future location of the nodes by having their current position, direction and speed information. By doing so, transmission at PU active regions is avoided and an alternative path is created to get around those regions. Furthermore, this prediction helps in the precision of the paths to the destination especially in cases that the destination has mobility too. SEARCH routing process is divided into (i) Route setup phase and (ii) Route enhancement phase. At route setup phase RREQ messages are transmitted through all the paths and channels that are not affected by the PU-active regions. All the intermediate nodes add their context info to the message and forward it until the destination is reached. Finally, the information received by the destination from all different paths is integrated by joint channel-path optimization algorithm to find the most efficient route. As from the name appears the Route enhancement phase is for the purpose of maintenance. In case a PU appears in a region where SUs are transmitting then the PU avoidance phase should get activated and nodes inform their vicinity about PU-active region and create an alternative path to the destination. Since SEARCH relies on geographic location and estimation
of the direction of movement, the accuracy of this information has a very high impact on the performance of the routing protocol.

2.8.3 **SPEAR**

SPEectrum-Aware Routing Protocol (SPEAR) [51] integrates the spectrum discovery into the route discovery process to mitigate spectrum heterogeneity problem. Channel selection is done on a per-flow basis which is claimed to minimize interference and achieve nearly optimal throughput. Among the two methods of managing SOPs, SPEAR uses the distributed architecture meaning that spectrum is searched and allocated locally at each node without the central knowledge about other node’s channel availability; this result intermittent node along the path of a route to switch between SOPs for the purpose of forwarding the data. SPEAR assumes each device has one dedicated control radio and one data radio. In a high level view, it uses the same signaling model as in AODV [48]. It initiates the route discovery by broadcasting a discovery message to be forwarded through the whole network and is eventually received by the destination. Route discovery message piggybacks the information about channel availability of nodes while traversing the network. Hence, the destination and all the nodes on the way of this broadcast are updated with channel availabilities of each other. Upon reception of the message, destination decides on the most optimal path for replying to source. All the nodes on the return route to source parse the message, extract channel information from it and forward the data. Next, by informing this channel usage to their surrounding one-hop neighbors, interference is avoided for the duration that the transmission is taking place. Intersection of flows and the impact on throughput is considered in this work with respect to TDMA-style channel scheduling.

2.8.4 **ASAR**

Ant-based Spectrum Aware Routing for Cognitive Radio Networks [52] provides a learning-based routing algorithm which utilizes two types of agents (called ants in this work which is based on other ant-based protocols [53-55]) to find spectrum-aware paths to the destination. Each node maintains a table holding information such as pheromone concentration, local heuristic information and statistical utilization history. Further into detail, this table holds the number of common channels to reach neighbors of the current node and history of path quality based on statistical probing. When a destination needs to be reached, unless the spectrum-aware path to destination exists, the source node needs to generate F-ant and broadcast it to all its neighbors on the common control channel. During this broadcast, all the nodes along the way check the channel availability and put their address in the F-ant. In case that prior F-ant has created multiple spectrum-aware paths to destination, one of them is chosen based on a probabilistic formula given in this work. Upon reception of the F-ant by destination, the routing information is extracted and B-ant is generated and sent back to source on the data channel. B-
ant is responsible to collect and update path quality and pheromone concentration along the path to source. Since the B-ant is sent over the data channel, the statistical data that it collects can exactly reflect the condition of the channel that a regular data may suffer from. The information that A-ant and B-ant provide to source, destination and nodes along their way, is the key factor in choosing the best path from source to destination. While F-ants and B-ants provide a full knowledge of the network and quality of the available channels, we expect it to come at the cost of the signaling load. Although it is claimed that signaling load is controlled proportional to traffic but there is no measurement of it given in this work. It is assumed that F-ants are sent over a common control channel but maintaining a common control channel in a dynamic CR environment is not feasible unless the infrastructure is maintaining it. The learning based algorithm of this work is a promising solution to the proactive learning nature of the CR.

2.8.5 SAMER

Spectrum Aware Routing in Cognitive Radio Mesh Networks (SAMER) [7] leverages a new routing metric to find the paths with higher spectrum quality and availability. While data is forwarded from source to destination, it adapts to the dynamic spectrum conditions and the route with highest spectrum availability is chosen as the best path for routing the data. Each node creates a local spectrum matrix holding SOPs available to both PUs and SUs. According to this work, an optimal spectrum aware path is defined based on three factors: 1) Hop count 2) End-to-End throughput 3) Spectrum utilization. This work also proves that in a dynamic spectrum system, routes which minimize spectrum utilization (optimal routes) is reproducible as minimum cost (shortest) paths in terms of positive weights. SAMER follows two level routing mechanisms to find a balance between long term route optimality and shortest opportunistic spectrum gain. Based on the number of neighbors each node calculates a cost value, each node that forwards the packet adds up to the cost value and forwards the packet to the next neighbor with highest SOP available. SOP is computed using the PSA (Path Spectrum Availability) metric designed by this work which targets both spectrum quality and availability. The loss probability formula proposed by [56] is one of the main elements for computing in this work. It is considered that each node has a full topology map which is possible for a wireless mesh network but an infeasible assumption for power constrained distributed MANETs. It is assumed that the cost function is pre-calculated for all the destination periodically which creates short periods of time where the cost is not up to date which is considered as future work. While maximum cost defines the upper bound limit for the cost function and is a trade-off between the short term and long-term performance, but there is no study provided on the optimal value of maximum cost. The link utilization based on PSA metric is a very promising factor considering the low standard deviation among each sample of the utilization. A steady throughput despite the variation in the node density is another strong point of this work.
2.8.6 SARP

SARP (Spectrum-aware Routing Protocol) [35] consists of two important modules, one is responsible for intelligent selection of MISF (Multi-Interface devices), the other responsible for intelligent selection of multi-paths. It is assumed that devices have multiple interfaces and each of them listen on different channels. The main goal of this routing protocol is to increase channel diversity hence reduce the frequency of channel switching within each interface of a node. The MISF (intelligent Multi-Interface Selection Function) considers delay of the RREQ packets as the main metric to evaluate channel conditions and load level on each channel of a node. Queuing delay of RREQ packets is the sum of transmission delays of all queued packets. So, it is considered as a sufficient metric to evaluate load and channel conditions of the link. IMPSF (Intelligent Multi-Path Selection Function) uses throughput increment as a metric for path selection. The throughput increment is defined as predicted throughput after a new application joins minus current throughput. It is assumed that the current throughput and the throughput after an application joins the network can be measured by means of application data rate, channel capacity and packet loss rate. Channel capacity is estimated by distance of sender and receiver provided by GPS (Global Positioning System) position information and the type of channels. Information such as data rate, queue length, channel types and channel capacity is achieved through cross layering between MAC, Physical and Network layers. The normal RREQ packet is broadcasted through all the interfaces and eventually reaches the destination. The first interface which has received the RREQ packet is only responsible for relaying the RREP packet from destination back to source. RREQ packet along its hop by hop relaying from source to destination provides the mean to calculate throughput increment and registers it in the routing table. RREP on its way from destination to source uses the information provided by RREQ to choose the path with highest throughput increment. The strong point about this work is the improvement in throughput by using multiple interfaces. But this improvement is achieved through the cost of routing overhead. The routing overhead that this routing protocol induces on the network is high and the justification is the number of broadcasts that is loaded on all the interfaces of a node to deliver RREQ packets. On the other hand, we can see rapid fluctuation in the standard deviation of throughput which is resulted from uneven distribution of the traffic in the network. Applications that are QoS constraint can highly suffer from these fluctuations.

2.8.7 IPSAG

IPSAG (IP Spectrum Aware Geographic routing) [57] targets routing in Multi-hop cognitive radio networks. Since CR coexists with the currently deployed technologies (operating on standard spectrum bands) hence (based on this work) it is necessary to differentiate routing to two different categories, 1) Targets routing inside of pre-existing systems (such as WIMAX, WIFI and WLAN) 2) Targets routing outside of pre-existing systems (which is an opportunistic environment in terms of SOPs); While a CR
user is located inside a pre-existing system the normal routing mechanism of that specific system is applied. Otherwise, IPSAG’s routing algorithm will be used to relay the data toward or from the pre-existing systems. In this work, a common control channel is dedicated to signaling purposes. The routing metric proposed takes into account the mutual SOPs between every two nodes, SOPs quality and locations of destinations node as input for calculating the metric. In order to create the geographical topology of the network, IPSAG broadcasts node ID and geographical location on the dedicated control channel. Furthermore, it defines global and local network knowledge through defining Global table and Local table. Global table holds the position and ID of the whole network topology and local table stores SOP availability and quality of the neighbors only. Each node takes the geographical location and SOP quality as factors to forward the data to destination. It is mentioned that route discovery is integrated with data forwarding, meaning that both data and signaling information are carried out in the same packet which is unlike the similar routing protocols (e.g. SEARCH [58]) in the same context. The function of routing being creation of routes for data transmission, the argument of data and signaling in the same packet does not seem feasible in technical view. It is also mentioned that a common control channel is used for signaling purposes but in the end the conclusion is that signaling and data are in the same packet which this two have contradiction with each other. Another assumption of this work is that the entire topology of the network can be obtained in real-time consequently CRs should have perfect sensing capability which is not possible in reality.

2.8.8 Channel assignment and bandwidth allocation algorithm for multi-channel wireless mesh networks

The work of [32] proposes a multi-channel multi-hop wireless ad-hoc network architectures which uses multiple 802.11 NICs operating on different channels to achieve spectrum awareness. The focus of this work is wireless mesh networks that serves as the backbone for relaying nodes traffic from wireless access point to wired network. It is claimed that by using multi-channel routing algorithm, the good-put performance of the network can be improved by a factor of 8. One of the aims of this work is to provide a routing algorithm that balances the load on the network and maximizes the good-put. The combined channel assignment and routing algorithm first calculates an estimate of the expected link load considering the given radio channel to each link, if the real load of routing over that channel is less than the expected load then the channel assignment process iterates; this process continues until the best available channels are assigned to each interface. The main aim of this work is not a complete routing algorithm but integration of routing and channel assignment. The two challenges not considered in this work are 1) Multi-channel multi-interface MANETs 2) Distributed channel assignment using signaling at network layer.
2.8.9 SAOR

SAOR (Spectrum Aware Opportunistic Routing) [59] favors CRN under wireless fading channels. This work implements a spectrum map for local sensing information. Furthermore, derives a routing metric known as OLT (Opportunistic Link Transmission) which is based on delay and connectivity to neighboring nodes. According to this work, constructing the entire spectrum map with all routing paths is more practical than other approaches. Each CR source collects link information for the entire network. Based on the derived OLT metric, creates a set of forwarding candidate list and broadcasts random network coded packets from a single batch with useful information. Destination CR receives these coded packets and responds with ACK transmission. The signaling mechanism for distribution of information (such as spectrum map) is not considered in this work. Hence signaling load which is an important factor in CRN is not evaluated in the analysis. The evaluation of SOAR in terms of end-to-end delay is a linear function which shows rather unusual behavior considering a fading radio environment and changes in CR.

2.8.10 Coupled/decoupled route and spectrum selection in DSA networks

The work of [43] analyzes the spectrum selection and routing from two different aspects of 1) When they are integrated together as a single module 2) the case when they are decoupled and operate independently in distinct OSI layers. This work argues that although multi-radio interfaces result more capability (in terms of simultaneous transmissions) but it can heavily exhaust energy which is an important factor in ad-hoc and sensor networks. Hence the assumption of this work is that each device is equipped with a single half-duplex NIC (unlike works of [32, 60]) which is in line with the design of IEEE 802.11 network devices on the market. This work models the problem of spectrum heterogeneity and routing as a conflict graph G. Each single-hop link maps to a vertex in the graph and an edge exists between two vertices if the corresponding links cannot be active at the same time. Furthermore, links that are in close proximity should utilize different channels as this can cause interference. The result of this method is a recursive algorithm creating a conflict free (in terms of channel and time scheduling) graph. Whenever multiple paths between a source and destination exist, an algorithm chooses the best route and channel combination to maximize throughput. This work concludes that a coupled design improves the end to end performance at many levels. Based on experimental results, it is shown that throughput increases as the channel diversity is increased which is promising. Since it is assumed that a central entity holds the knowledge of the topology and spectrum diversity, distributed collaborative design of this work is still considered as a major challenge.
2.8.11 A layered graph model to target routing in DSA

The work of [37] provides another routing approach targeting DSA networks based on a layered graph model. The interface to channel mapping and route calculation are highly dependent on one another; hence this work believes that integrating channel assignment into the route calculation process can result optimal paths. The layered graph algorithm provided in this work is designed with objectives of 1) Targeting networks with heterogeneous channels 2) Computing routes with the aim of maximizing network connectivity 3) Performing interface and channel assignment jointly 4) Diversify channel selection on a routing computation to maximize capacity and minimize interference. There are defined four types of graph edges 1) Access edge to connect a node to its sub-nodes 2) Horizontal edges to connect sub-nodes within the same layer which represents available channels among nodes 3) Vertical edges to connect sub-nodes which indicated forwarding capability to different channels of the same node 4) Internal edges to connect a sub-node to its auxiliary node. By mapping the network topology to graph connections, the problem of route calculation and interface assignment is managed through shortest path algorithm. The cost for vertical traversal of topology graph is set to a negative value; this is to increase the frequency of switches among channels hence reducing the probability of channel usage conflict in every node. In the end, the improvement of the proposed algorithm is benchmarked against sequential interface assignment algorithm. This work uses shortest path algorithms based on the assumption that channels have identical quality; this assumption does not stand true since depending on the interference received from the environment, channels do not have the same quality and capacity which is the reason why metrics such as ETX [56] has been implemented to progressively monitor the quality of them.

2.8.12 On-demand spectrum-aware routing

The work of [47] proposes an approach to reactively initiate route computation and frequency band selection. In addition to this a novel scheduling scheme for intersecting flows in individual nodes has been proposed. This work assumes that there is a separate spectrum agile transceiver which forms a control channel. Furthermore, a cumulative delay metric is derived considering the switching and back-off delays along the path with the main aim of minimization of this delay along the routes. Spectrum aware On-demand Routing Protocol (SORP) implemented by this work inherits the basic functionality of AODV routing protocol. The delay analysis, points out the switching delay which was not well established in previous works on spectrum-aware routing. One irregularity observed in the simulation results provided by this work is the steep downward trend of cumulative delay until spectrum sparsity reaches 50MHz and the upward trend afterward. Since switching delay has a direct mathematical relationship with spectrum sparsity, the expected behavior is a constant upward trend in the performance of SORP in terms of cumulative delay. Based on the simulation results the K-hop scheme has a higher cumulative delay than Switch-aware scheme; which concludes that frequent switching, while reduces
the probability of interference with other flows, but has a very negative impact on cumulative delay of routes within the network.

2.9 Taxonomy of spectrum-aware routing protocols

In this section, we provide a taxonomy of the spectrum-aware routing protocols which were discussed and analyzed in Section 2.8. We have already explained most of the categories of the taxonomy in previous sections but two of the new concepts are, 1) Channel Stability capable, 2) Redundant route caching. A routing protocol has channel stability capability if it is capable of measuring stability of a channel through historic or statistical information of the channel based on previous time frames. Furthermore, a routing protocol that performs route caching is capable of recording more than one route for each destination as backup routes.
## Chapter 2: Background Study and Related Research

### Table 2-1: A Taxonomy of Spectrum-aware routing protocols

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Spectrum Management Architecture</th>
<th>Single/Multi-Transceiver Design</th>
<th>Route Discovery Mechanism</th>
<th>Existence of Common Control Channel</th>
<th>Channel Stability Capable</th>
<th>Redundant Routes Caching</th>
<th>Location Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>M. Huisheng et al. [41]</td>
<td>Distributed</td>
<td>Single</td>
<td>Re-Active</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>K. R. Chowdhury et al. [49]</td>
<td>Distributed</td>
<td>Single</td>
<td>Pro-active</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>A. Sampath et al. [51]</td>
<td>Distributed</td>
<td>Multiple</td>
<td>Re-active</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>L. Bowen et al. [52]</td>
<td>Distributed</td>
<td>Single</td>
<td>Hybrid</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>I. Pefkianakis et al. [7]</td>
<td>Distributed</td>
<td>Multiple</td>
<td>Proactive</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>J. Suyang et al. [35]</td>
<td>Distributed</td>
<td>Multiple</td>
<td>Hybrid</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Ba et al. [57]</td>
<td>Distributed</td>
<td>Single</td>
<td>Reactive</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>A. Raniwala et al. [32]</td>
<td>Centralized</td>
<td>Multiple</td>
<td>Proactive</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>L. Shih-Chun et al. [59]</td>
<td>Distributed</td>
<td>Single</td>
<td>Reactive</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>W. Qiwei et al. [43]</td>
<td>Centralized/Distributed</td>
<td>Single</td>
<td>Reactive</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>X. Chunsheng et al. [37]</td>
<td>Centralized/Distributed</td>
<td>Multiple</td>
<td>Proactive</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>C. Geng et al. [47]</td>
<td>Distributed</td>
<td>Multiple</td>
<td>Reactive</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>
2.10 Background study, load balancing in ad hoc networks

Over the past, load balancing has been the focus of many researches involving ad hoc networks. As load balancing is a network-wide optimization and improvement mechanism, the solutions involving this area has mainly been implemented in the network layer of the OSI model. Given that network layer is responsible for managing the network-wide connectivity information, it is ideal for implementation of any load balancing algorithm. Routing protocols, as network layer agents, are responsible for computation of the network connectivity graph (taking into account the cost/benefit metrics) which would make them the most suitable candidates to hold the load balancing strategies. By taking load balancing strategies in the routes requested by the nodes in the network, a minimum acceptable QoS can be guaranteed in the entire network. In ad hoc networks, routing protocols operate in a distributed manner within each node without any direct interactions among them. Given that design of routing protocols are based on a distributed paradigm, load balancing algorithms have to operate under the same network assumptions. As a result, achievement of an overall network gain would be extremely challenging. As shown in Figure 2-7, the work of [29], has categorized routing protocols into the three categories of Delay-based, Traffic-based and Hybrid-based in terms of their load balancing approaches. In the delay-based protocols, minimization of link delay has been identified to have direct impact on the network-wide load balancing.

\[\text{Load Balanced Routing}\]

- Delay Based
- Traffic Based
- Hybrid

*Figure 2-7: Categories of Load-balanced Routing*

The work of [61] proposes a new routing algorithm based on AODV so called LAOR (Load-Aware On-Demand Routing); this work claims that by identifying low delay routes, we can achieve balanced load conditions in the network, however this work has not provided any results supporting the argument that the achieved gains in end-to-end delay and packet delivery ratio is the result of this balanced load condition or vice versa. The works proposed in [62], [63] and [64] use the traffic-based load balancing to achieve even load distribution in the network. As load is mainly generated by the application layer traffic, load balancing can be well achieved by distribution of traffic in the network. The work of [62] highlights that other traffic-aware load balancing algorithms use buffer size at each node to estimate the traffic load passing through that node. However due to the fact that packets buffered at each node can have different sizes, the buffer size alone cannot be used as a good estimator of load. On the other hand, as packets buffered at each node would possibly have different destinations, by analyzing the buffer as
a whole, we cannot achieve a conclusive evaluation on the load level on each specific node. Hence the author argues that it is more effective to use the actual packet sizes in the queue and their generation rates to estimate the imposed load in the network. The work presented in [64], proposes a new on-demand routing algorithm so called LBAR (Load-Balanced Ad hoc Routing) which attempts to achieve load balancing by circumventing congested segments of the network. It is shown that the proposed algorithm achieves higher PDR (Packet Delivery Ratio) when compared to AODV and DSR (Dynamic Source Routing). According to the study performed by [29], Hybrid-based routing protocols achieve load balancing by combination of delay-based and traffic based techniques. CSLAR (Content Sensitive Load Aware Routing) [65] is one example of Hybrid-based routing protocols. The work presented in [66], introduces a new routing protocol so called DLAR (Dynamic Load-Aware Routing). DLAR is a re-active routing protocol that uses the buffer size of intermediate nodes as a performance metric for route discovery and computation phase. The main aim of DLAR is to choose routes with minimum cumulative buffer size. It is shown that DLAR achieves better PDR and End-to-End Delay compared to the baseline protocol DSR. Another work in this area is LWR (Load Aware Routing) [67] in which the author makes the argument that sometimes a detour of routes over the idle nodes can have significant positive impact on performance of routing in ad hoc networks. The main idea behind LWR is that the heavily loaded nodes should opt-out of taking part in routing of data in the network by dropping route requests. In this way, the heavily loaded segments of the network go under a relaxation process which should gradually balance the load distribution in the network. LB-AODV implements a load balanced version of AODV routing protocol [68]. Load balancing is performed by a distributed grouping mechanism which performs a logical division of mobile nodes into distinct groups. It is claimed that the grouping mechanism reduces the number of unnecessary retransmissions of routing messages and by distribution of source nodes among the groups the packet relaying is limited to each source node within its group. The grouping idea used in LB-AODV is based on the ZRP (Zone Routing Protocol) presented in [69]. Another approach in load balancing is to geometrically target the problem. The work presented in [70], uses a geometrical model to perform load balancing in the network. More specifically it considers a special case where nodes are located in a narrow strip with the width of at most 86% of communication range. A location based load balanced routing protocol is presented in [71], which is so called LB²R. It uses a dual GH (Grid Header) routing scheme which balances the routing load among two GHs. It is claimed that this method improves queueing delay and congestion on heavily loaded nodes. However, the results do not show an improvement in terms of load balancing.

One of the approaches to achieve load balancing in ad hoc networks is by multi-path routing protocols. Multi-path routing protocols store multiple alternative paths for every source-destination pairs. The main idea is that utilization of the alternative routes for forwarding the data can result distribution of the network load. While creation of multiple routing paths between any source destination pair in the network has been proven to be beneficial in the context of wired networks, this
scheme has been under debate for wireless networks. The analytical model presented in the work of [72] argues that, unless the alternative paths provided by multipath routing consists of a very large set, the gain achieved by load distribution is almost the same as single path shortest hop routing. As ad hoc networks are not particularly designed for large dense networks, having a large set of alternative paths is an infeasible assumption and it can be concluded that multipath routing in a realistic ad hoc scenario would not result effective load distribution.

2.11 Back-pressure Routing

Back-pressure routing utilizes congestion gradients to dynamically schedule and route user’s data across multi-hop networks. The original idea of this algorithm was introduced by Tassiulas and Ephremides in 1992 which is based on Lyapunov drift theory [73]. The algorithm is designed to operate under a slotted time manner in multi-hop communication systems. At every time slot, the next hop route is chosen based on the differential backlog of the queues among the neighbouring nodes. The name back-pressure was initially given to this algorithm in the work of [74], which is where the algorithm was initially applied to networks with dynamic mobility factors as well as ad hoc networks. Back-pressure is inspired by the way water flows in a network of pipes based on actual water pressure gradients. Back-pressure routing is proven to be throughput optimal and is capable of functioning in a network with dynamic links (channels). The fundamental idea behind back-pressure routing is that links in the network are considered as valuable resources which their usage should be maximized. By monitoring differential backlog of neighbouring nodes, back-pressure scatters the network traffic from congested areas of the network to lightly loaded parts; this results load balancing in the network and is proven to create throughput optimality. The downside to the optimal throughput is higher end-to-end delay and looping effects. As it was mentioned before, conventional routing protocols use hop count as the quality metric and their aim is to minimize the length of routing paths. As back-pressure algorithm is designed to benefit from longer length paths resulted by traffic distribution, it is said to be in conflict with the conventional routing protocols using shortest path paradigm. Back-pressure routing performs efficiently under high network load conditions where network queues are full and stable but fails when the network is lightly loaded. It is a known fact that under low load conditions packets circle around the network resulting in excessive and unnecessary delays. Furthermore, stabilizing network queues using traffic gradient and utilization of all available paths results optimal throughput at the cost of extra energy consumption in the network nodes. Hence another downside to back-pressure routing is the energy consumption.
2.11.1 Backpressure Queue Structure

In multi-hop communications networks queues are created when the rate of generation or arrival of data packets at a node are greater than the rate at which the node can send or forward those packets. Conventionally nodes in ad hoc networks operate based on a single queue structure regardless of the packet size or intended destination. However, one of the assumptions that is necessary for the functionality of back-pressure routing is for nodes to have distinct queues per the number of destinations in the network. The reason this is important is that back-pressure calculates the differential backlog of queues on a per destination basis.

2.11.2 Back-pressure Mathematical Model

Considering a multi-hop network with M static nodes and the network operating in slotted time \( t \in \{0, 1, 2, \ldots \} \). At each time slot, the back-pressure algorithm not only schedules the next transmission but also moves the packets in the right direction in order to route them to destination. As it was explained in Section 2.11.1, all nodes store and maintain their queue backlog based on the destination of the packets. Now let us consider \( Q_m^d(t) \) to be the queued packets at time \( t \) currently at node \( m \), that have destination \( d \). Depending on the type of communication system that is being modelled the unit of \( Q_m^d(t) \) can be different. If our system uses CBR traffic, as packet sizes are the same, then the queues can simply be the integer number of packets. On the other hand, if the system uses VBR (Variable Bit Rate) traffic, the actual packet size in bits/bytes needs to be taken into account. The network scheduler has the responsibility of optimizing the scheduling-routing process such that the Eq. 2.1 satisfies at every time slot \( t \) for all \( m \in \{1,2,\ldots,M\} \) and \( d \in \{1,2,\ldots,M\} \) with the condition that \( m \neq d \) [75].

\[
Q_m^d(t + 1) \leq \text{Max} \left( Q_m^d(t) - \sum_{y=1}^{M} \delta_{my}^d(t), 0 \right) + \sum_{x=1}^{M} \delta_{xn}^d(t) + I_m^d(t) \quad \text{Eq. 2.1}
\]

\( I_m^d(t) \) is the new randomly generated data destined to node \( d \) that arrives in node \( m \) at time slot \( t \). Additionally, \( \delta_{xy}^d(t) \) is the transmission rate which is pre-allocated to link \((x, y)\) for the data traffic destined to node \( d \) at time slot \( t \). The amount of queued data, \( Q_m^d(t) \) might not necessarily be greater than the permitted transmission rate for the link \( \delta_{my}^d(t) \), which simply means that the queue is almost empty. In these situations, the Max operator in Eq.1 avoid getting a negative value out of \( (Q_m^d(t) - \sum_{y=1}^{M} \delta_{my}^d(t)) \). Furthermore the \( Q_m^d = 0 \) for all \( t \) and all \( d \in \{1,2,\ldots,M\} \); this is simply because no queue contains data that has itself as destination.
2.11.3 Back-pressure Routing in Networks with Dynamic Channels/links

There are numerous examples of networks with dynamic channel/link availability such as MANTEs or even CR-MANETs. Back-pressure algorithm can be easily applied to such networks since the network optimization inequality presented in Eq. 2.1 takes the channel transmission rate $\delta_{my}(t)$, into account, which is a measure of link quality. Hence in dynamic network scenarios where at certain time intervals the quality of some links is compromised, the back-pressure algorithm only pushes the amount of traffic which the network links are capable of handling.

2.12 Quantum Game Theory and Load Balancing

This section is provided to briefly cover the background knowledge required for the third contribution in this thesis provided in Chapter 5. Due to the interdisciplinary nature of quantum game theory, covering this topic in detail is out of the scope of this work, and hence it is recommended for the interested reader to use the references provided in this section to cover the theory in further detail.

Game theory is the theory of strategies. With the aid of this theory, players of the game are suggested a strategy that maximizes their total payoff. Quantum game theory provides a framework to utilize entangled particles with the aim of affecting decision-making process of distant players without transmission of any information. In a quantum game, players can use properties of entangled particles to have instantaneous influence on the strategies of other players to increase their pre-defined utility function [76-78].

In some occasions, the strategy chosen by each player is unknown to the other one. In such cases, we should consider a game theory with incomplete information. The games with incomplete information are studied in literature under the category of Bayesian game theory [79, 80]. Brunner and Linden [81] discovered a connection between quantum mechanics and Bayesian games. They show that the quantum utility function in Bayesian games can be written in an inequality form that is significantly similar to Bell inequalities. If behavioral states of a system are identified to be in violation of the Bell inequalities then the system is known to be quantum. In a quantum system, the accessible space of states is larger than the classical system (non-quantum), which then can be utilized to maximize the utility function of that system. In other words, Brunner and Linden have shown that, if players exploit quantum states and quantum operators as strategies in a Bayesian game, the players can maximize their utility function.

At Bayesian games (also known as incomplete information games), players do not have full knowledge about other players types and strategies. Despite this, the players can benefit from an advisor, who updates them with the information of their types and strategies. Hence, our interest is concentrated on the games in which an advisor can provide entangled states for players. As a result,
players can use the entangled particles to acquire relative knowledge about one another that affects their strategies. The strategies are fundamentally designed to maximize the utility function, hence abiding by them would result in a gain in utility function. A mathematical formulation of quantum games and strategies can be found in [76, 82]. Since rotations play a key role in quantum game theory, the focus of Section 5.3.1 is on these operators. So, we formulate rotation operators and demonstrate their capability in rotating the spin of quantum particles. These operators are widely used as quantum strategies at quantum game theory.

Utilization of entangled particles results better decisions to be made by the players, which can be utilized to achieve a gain in a predefined utility. In the case of network traffic management, sender nodes can benefit from strategies that help them choose routes that are capable of fair distribution of the network load. Therefore, quantum game theory offers a framework for load balancing in the network. In Chapter 5 of this thesis we have demonstrated that by utilizing entanglement in our game setup and designing a utility function we can achieve load balancing.
Chapter 3

3 Spectrum-aware Routing

3.1 Introduction

In this chapter, we have summarized the first contribution of this thesis which is a proactive Spectrum-aware Routing based on OLSR. In Chapter 2 we covered the fundamentals of cognitive radio and their potential in providing stability, increased capacity and higher levels of QoS provisioning in CR-MANETs. The concepts of spectrum sharing, spectrum mobility and management can be utilized in various entities in CR-MANETs to result performance gains. In Section 2.3, the problems involving wireless ad hoc networks was summarized such as excessive noise, interference, dynamic topology, security and load balancing. It was covered in Chapter 2 that the dynamic spectrum environment of CR-MANETs requires a spectrum management entity to be able to efficiently maximize utilization of SOPs them. Furthermore, it was identified that the routing protocols as agents in the network layer are the best candidates in hosting a globally solid spectrum management mechanism in the network. Connections of all nodes in the network are managed by the network layer and it is where the network topology is created and maintained. Routing protocols prior to introduction of DSA systems relied on availability of only a single channel/frequency at any given time which is reliably managed by lower layer protocols (at MAC and PHY layers). In DSA systems, creation of wireless links is still the responsibility of lower layers with the difference that there might be multiple choices of wireless channels (aka SOPs) subject to availability in time and geographical location. Routing protocols can monitor the whole network topology and nodes through network layer signalling and are in fact the best candidates for channel management. As it was detailed in Section 2.6, spectrum-aware routing protocols are defined as a category of protocols which utilize the spectrum opportunities resulted by the idea of cognitive radio and DSA to increase the network capacity and provide a higher level of QoS provisioning. In this chapter, we have initially analysed the performance of 4 conventional MANET routing protocols via simulation study. Next, we have used OLSR as the base of our spectrum-aware implementation and analysed the challenges involved in the implementation process. Finally, we have modelled the multi-channel topology structure of CR-MANETs based on the concept of multigraphs and report a solution for computation of shortest weighted paths in this setup. Lastly the performance of the proposed Spectrum-aware OLSR algorithm is analysed against the baseline OLSR and a considerable performance gain in terms of packet delivery ratio is reported.
3.2 Direction and Vision

The aim of this chapter is to study various Spectrum-aware routing techniques and to propose a novel routing protocol utilizing appropriate metrics to address the identified challenges. Toward this aim, the research was divided into the following.

1. Benchmarking Analysis: Analysing the performance of conventional MANET routing protocols through simulations. The reason for doing this analysis was first to familiarize with the current routing protocols that are designed for MANETs and analyse their behaviour under various simulation scenarios. Furthermore, to choose one as the baseline implementation framework for the proposed Spectrum-aware routing protocol. The analysis is provided in Section 3.3.

2. Routing Metrics: Another important aim was to compare few of the well-known routing metrics and compare their performance against each other in the baseline OLSR routing protocol. The reason behind this goal was firstly to develop a better understanding of the performance OLSR with various routing metrics which are specifically designed to target MANETs. Secondly, with modification to one of the metrics adapt it to our proposal of spectrum-aware routing protocol. The results of this analysis are provided in Section 3.5.

3. A multi-channel protocol: Another challenge was to find a MAC/PHY layer protocol that can be adapted to the multichannel structure of CR-MANETs. As the area of CR is relatively new and to the best of our knowledge the multi-channel idea has not yet been fully standardized; the other reasonable option was to use a well-designed standard such as IEEE 802.11 (ad-hoc mode) and implement a multichannel structure based on it in the simulation environment. This is explained in detail in Section 3.7.

4. Spectrum-aware Signalling: It is the signalling mechanism which defines a routing protocol. Hence defining a signalling mechanism which is responsible for updating topology and SOP information among the nodes within the network was essential. The detail of the signalling mechanism and its implementation is provided in Section 3.8.

5. Multi-graph and routing computation: The other check-point toward implementation of a fully functional spectrum-aware routing protocol was providing a solution to the challenging problem of multi-graph (which results from multiple SOPs) and embedding this solution in a route calculation algorithm. This is fully explained in Section 3.9.

6. Implementation of Spectrum-aware OLSR: At last all the achievements from above sections where integrated to implement the spectrum-aware OLSR. The design, implementation and simulation-based proof of concept are provided in Section 3.10.

It is worth noting that all the implementation and analysis provided in this section are based in OMNET++ even-based simulation platform [83, 84].
Chapter 3: Spectrum-aware Routing

3.3 Performance Comparison of Conventional MANET Routing Protocols

As it was explained in section 3.1, the first check point toward implementation of a spectrum-aware routing protocol was to perform benchmarking simulations on four of the most popular routing protocols designed for MANETs; which are OLSR [85], AODV [86], DYMO [87] and DSR [88]. The simulations were performed in OMNET++ event based simulator. These analyses were performed based on variations of node’s velocity and relay node density. The aim was to analyse performance of these protocols in terms of End-to-End Delay (E-t-E Delay), PDR and Normalized Routing Overhead. To provide reliable results and improve confidence levels, we set the transient interval (warm-up period) to 700s (total sim. Time = 2000s) and report averaged results over 50 simulation runs with different seed sets.

It is very important to consider that for all the statistical averaging and error bar analysis performed on the simulation results presented in this chapter a confidence level of 95% has been used.

The simulation scenario is a basic configuration of 6 source hosts and 6 destinations i.e. 12 Active Hosts, located randomly in the simulation area (a square of size 2000m*2000m). Sources generate and send CBR data traffic through the relays to the destinations. In the case of Speed variation analysis, number of relay hosts are fixed to 15 and in the other case of node relay number variation, speed is fixed to 3m/s. The rest of simulation set parameters are listed in Table 3-1.

Table 3-1: Benchmarking Simulation Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulation Time</td>
<td>2000s</td>
</tr>
<tr>
<td>Transient Interval</td>
<td>700s</td>
</tr>
<tr>
<td>Number of repeats</td>
<td>50</td>
</tr>
<tr>
<td>Simulation Area</td>
<td>2000m*2000m</td>
</tr>
<tr>
<td>Active Hosts (Default)</td>
<td>12</td>
</tr>
<tr>
<td>Relay Hosts (Default)</td>
<td>15</td>
</tr>
<tr>
<td>Mobility Model</td>
<td>Random Waypoint</td>
</tr>
<tr>
<td>Mobility Initial X and Y positions</td>
<td>Random</td>
</tr>
<tr>
<td>Mobility Speed (Default)</td>
<td>3m/s</td>
</tr>
<tr>
<td>Mobility Wait Time</td>
<td>Random [ 3s , 5s ]</td>
</tr>
<tr>
<td>Number of Flows</td>
<td>6</td>
</tr>
</tbody>
</table>
3.3.1 Speed Variation

In this section performance of 4 different MANET routing protocols is analysed over variations of velocity. We can see that both OLSR and AODV manage to achieve a very stable end-to-end delay regardless of variations in mobility applied to the network as shown in Figure 3-1. Furthermore, according to Figure 3-2 AODV and OLSR show a higher PDR compared to DYMO and DSR. Given the pro-activeness of OLSR routing protocol, this stability in terms of End-to-End delay and a relatively high PDR is justifiable. OLSR relies on the regular dissemination of HELLO and TC messages which completes the topology graph prior to any request for data transmission. As a result, routes are ready upon request for data transmission and PDR stays relatively high even at higher speeds. On the other hand, AODV which is a re-active routing protocol achieves a very good performance too; this can be justified by the fact that AODV efficiently and quickly responds to network changes. Unlike OLSR, AODV updates routing tables upon any request for data transmission. It performs on-demand signalling by sending RREQ (Route Request) and RREP (Route Reply) messages; not only the nodes along the path of RREQ messages (broadcasted by source node) update their routing tables but furthermore this
table is optimized upon return of RREP message (based on minimum hops) from destination. Another signalling feature of AODV is RERR (Route Error) messages. RERR messages inform other nodes in the network that a link that was available before is no longer available which can trigger a link discovery mechanism in the other nodes in the network. At higher speeds where links loss is more frequent, with generation of RERR (Route Error) messages AODV maintains up to date routes and reduces end-to-end delays. As a result of all these signalling mechanisms, AODV can manage a good end-to-end delay in low density network scenarios. DSR which is a routing protocol designed for wireless mesh networks performs similar to AODV and OLSR in the static case but its performance degrades as speed increases. This is mainly due to the fact that DSR is a source routing protocol. In DSR as RREQ message is being broadcasted in the network, each hop along the path amends the RREQ by adding its address at the end. Destination nodes send back the RREP message based on the addresses which were listed in the RREQ. As a result, under high mobility, the RREP messages follow routes that are no longer valid (due to the high frequency of changes resulted by mobility) back to source and cause high end-to-end delay and low packet delivery ratio as it can be seen in the results shown in Figure 3-1 and Figure 3-2. Generally, source routing protocols are not suitable for MANETs where topology changes are frequent. DYMO is a simplified version of ADOV which according to the results shown in Figure 3-1 and Figure 3-2 performs worse than AODV in terms of PDR and end-to-end delay. Although DYMO claims to be loop-free, we noticed many routing loops created in our static simulation scenario of DYMO and that is why the performance of this protocol in terms of end-to-end delay and packet delivery ratio is not acceptable. These routing loops happen less frequently under the dynamic scenario where the speed of nodes increases which is why the end-to-end delay is reduced and follows a stable value.

![End-to-End Delay](image)

*Figure 3-1: End-to-End Delay VS Speed (OLSR, AODV, DYMO and DSR)*
The signalling load reported in this result section is in fact a normalized the signalling load (in Bytes) which is calculated by division of signalling load by the number of received packets (in Bytes). As it can be seen in Figure 3-3, the normalized signalling load of AODV and OLSR are the lowest, justifying that they are the most efficient protocols in terms of signalling. As OLSR is a pro-active protocol, its signalling load is marginally higher that the re-active AODV. DSR has a low normalized signalling load in a static network scenario compared to DYMO, which is due to the fact that at low speed its PDR averages to 92% that is higher than all other protocols. In dynamic scenarios resulted by increase in speed, PDR falls dramatically hence the ratio of signalling load to the number of delivered packets increases rapidly leading to a higher normalized signalling load.
3.3.2 Relay Node Density Variation

In this section performance of 4 different routing protocols, OLSR, AODV, DYMO and DSR are analysed under variations of Relay Node Density (RND).

On an average basis, AODV and OLSR provide very low and reliable end-to-end delays versus variations of Relay Node Density (RND). If we have a close look at Figure 3-4, as RND is increased, AODV keeps a nearly constant end-to-end delay but OLSR’s end-to-end delay increases causing a gap between the slopes of AODV and OLSR in terms of end-to-end delay performance. The reason behind this is that as the RND increases, there are more nodes in the network that disseminate the signalling data pro-actively and this is a source of interference for other nodes in the network. On the other hand, as AODV is a re-active routing protocol, it only sends signalling messages whenever necessary and causes less signalling load on the network that results lower end-to-end delay. In terms of PDR, as shown in Figure 3-5, AODV performs better than all other protocols including OLSR. The good performance of AODV can be justified based on low signalling load and the fact that AODV has proven to be loop free. The low signalling load reduces the chance of collisions among the nodes in the network and the loop-freeness causes less delays and higher PDR. In DSR, the end-to-end delay follows an upward trend as RND is increased from 5 to 35, which is the same behaviour as AODV and OLSR. In section 3.3.1, we explained that DSR is a source routing and due to that, it does not perform well under mobility. This is the main reason that DSR performs worse than OLSR and AODV. The reason behind seeing a sharp downward trend in DSR’s end-to-end delay performance after 35 RND lies behind the number of delivered packets after this point. We can see that DSR’s PDR decreases as RND increases and that is also due to the source routing structure of DSR. After passing the 35 RND point, DSR is no longer capable of keeping stale routes due to raise in the number invalid (error) routes in the routing tables. After this point, DSR’s PDR drops sharply due to the lower number of delivered packets. But the packets that are being delivered benefit from short routes which explain lower end-to-end delay observed in this figure. In DYMO the case is completely different from other routing protocols. We can see that as the RND is increased, the end-to-end delay decreases. As it was explained in Section 3.3.1, DYMO follows a source routing structure like DSR with the difference that in DSR source routing is only applied to the RREQ messages but in DYMO it is applied to both RREQ and RREP messages. Based on a complex process, when there are lower number of relay nodes available in the network, under a mobile scenario DYMO keeps on failing to deliver the packet as the source routing not only creates unstable routes based on RREQ but also on the return path of RREP messages.
Figure 3-4: End-to-End Delay VS Relay Node Density (OLSR, AODV, DYMO and DSR)

Because of invalid routing entries in the routing table, the number of re-tries for sending data packets increases and the end-to-end delay becomes much higher than AODV, OLSR and DSR. As RND increases, there are more nodes available in the network, there are more alternative paths upon each re-try for sending the data, as a result the possibility of reaching the destination via a better path increases and the end-to-end delay decreases.

Figure 3-5: Packet Delivery Ratio VS Relay Node Density (OLSR, AODV, DYMO and DSR)

As shown in Figure 3-6, AODV has the best normalized signalling load, which is directly explained by its re-activeness and high packet delivery ratio. Following that is OLSR which due to its pro-activeness has a sharper slope compared to AODV. DSR performs better in low node densities but
sharply becomes the worst performing protocol in terms of routing overhead. For the case of DYMO as we explained earlier in this section, as the number of RND increases end-to-end delay decreases which is an indication of more stable routes in the routing tables that need less frequent updates; hence the signalling load flats out at high RNDs.

![Normalized Routing Overhead VS Relay Node Density](image)

*Figure 3-6: Normalized Routing overhead VS Relay Node Density (OLSR, AODV, DYMO and DSR)*

### 3.3.3 Conclusion on Benchmarking of MANET Routing Protocols

The performance analysis on 4 of the most popular MANET routing protocol shows some interesting results. It can be observed from both speed and RND variations, that OLSR and AODV are the best performing candidates in terms of all performance indicators selected. In some of the experiments, re-active AODV performs even better than the pro-active OLSR due to its efficient route discovery and maintenance mechanism. DYMO and DSR on the other hand are not well suited for MANETs, mainly due to mobility related issues. DSR does not respond well to the topology changes at high speeds. DSR is well suited for wireless mesh networks where there is no mobility involved. The idea behind implementation of DYMO has been to simplify AODV and based on the simulation results, the modification to the nearly perfect AODV not only does not yield any improvement but degrades the performance of the protocol from many aspects.

As one of the main aims of this benchmarking analysis was to choose one of the protocols as the baseline of our spectrum-aware routing protocol, we decided to select OLSR. Further reasoning for choosing OLSR as the base of our implementation is given in Section 3.4.
3.4 OLSR as the Baseline of Implementation and Research

Based on the simulation analysis given in Section 3.3, we could confirm that OLSR outperforms the other routing protocols in terms of stability and relatively affordable signalling overhead. The reason that OLSR is preferable over AODV as the base of our implementation is mainly due to its pro-active structure. As it was explained in Chapter 2, given the dynamic structure of DSA systems, high route discovery delay is neither affordable nor recommended in DSA based networks; as a result, a pro-active routing protocol, is more likely to perform better than re-active protocols in such scenarios. When the network size grows, OLSR supports better QoS constraints compared to a re-active routing protocols such as AODV [89]. While AODV performs well in single channel MANETs, based on our literature analysis it may not be the best candidate under the multi-channel (DSA) environment of CR-MANETs.

On the other hand, in a DSA system with the added complexity of dynamic channel allocation, the added signalling load is unavoidable; based on this, we conducted a thorough analysis of OLSR’s specification and the idea of MPRs (Multi Point Relays). The reason for implementation of MPRs in OLSR is to reduce the number of broadcasts performed in each cycle of the route discovery phase. As it is shown in Figure 3-7, utilization of MPRs results less broadcasts compared to the classical broadcasting performed in most routing algorithms. Nodes in the network selectively choose certain neighbours as their MPRs which are responsible for distribution of the signalling messages [85]. This method is known to substantially reduce the signalling overhead as compared to the normal flooding mechanism implemented in other routing protocols such as AODV.

The main idea was to merge the optimization provided by MPRs into our spectrum-aware protocol to minimize the added signalling load. Consequently, the less signalling load in the network leads to more data throughput.
The base implementation of OLSR uses hop-count as a routing metric. Shortest hop routing is simple to implement but is insufficient to maintain a minimum level of Quality of Service (QoS). The use of hop-count as a routing metric could result choosing shortest hops which might follow longer routes. Additionally, it could result traversing through congested links which have high interference levels. In conclusion, shortest path does not guarantee the best quality paths in any way. As a result and in order to overcome this limitation, other routing metrics such as ETX [90], MD (Minimum Delay) [91] and ML (Minimum Loss) [92] have been introduced. The focus of these new metrics is to optimize the quality of chosen routes rather than their length. Further analysis on OLSR and its operation with these metrics is given in Section 3.5.

3.5 OLSR with Different Metric-wise Configuration

In this section, 4 different metric-wise modifications of OLSR which are OLSR-HC (Hop-Count), OLSR-ETX, OLSR-ML and OLSR-MD are analysed in terms of, End-to-End delay, Packet Delivery Ratio and Normalized Routing Overhead. The two main scenarios where these analyses have been performed are Speed variations provided in Section 3.5.1 and Load Variations in Section 3.5.2. While the benchmarking results of Section 3.3 was performed based on the INETMANET 1.0 package of OMNET++ simulator, the results of this section are based on INETMANET 2.0. ETX [56], ML and MD are three different metrics from three different works which has been embedded in OLSR. We have setup a simulation scenario comprising of 12 active nodes (6 sources and 6 destinations) which are distributed randomly in the network. Furthermore, to analyse the performance of these protocols under the right conditions we have created 15 relay hosts which are also distributed randomly in the network. The Relay hosts provide multiple path opportunities for routing the data and deciding upon quality of these routes depend on the specific metric that OLSR uses. The scenario is defined in such way that Active hosts are supposed to relay their (Application Layer) traffic through either other Active hosts or Relay hosts through the network. To increase the confidence level of our results we have repeated each simulation run for 50 times (with different seed sets) and ignored the first 700s (Transient Interval) of each individual simulation. The rest of simulation parameters are the same as what was listed in Table 3-1. In Section 3.5.1, the performance of these metrics is analysed under variation of speed from 0 to 35 m/s and in Section 3.5.2 under different load conditions from 1 to 40 KB/s.

ETX as a routing metric was implemented to detect high-throughput paths on multi-hop wireless networks. It reduces the total number of re-transmissions required to deliver a packet successfully hence incorporating it in OLSR should maximize throughput of the network. ETX’s fundamental functionality is supported by sending probe packets over each link in both directions and measuring the number of missed probe packets, the stability of each link is measured. The formula to calculate ETX is given in Eq. 3.1, in which $d_f$ is the measured probability that a data packet successfully arrives at the recipient
and $d_r$ is the probability of successful delivery of the ACK packet. Hence, we can say that $d_f$ and $d_r$ are the probabilities of forward and reverse links. OLSRd project added ETX as a metric of choice which claims to improve the performance when compared to the conventional minimum-hop metric used in the base implementation of OLSR and many other protocols (AODV, DYMO, DSR and etc.).

$$ETX = \frac{1}{d_f \times d_r}$$

$$d_f = \text{Forward Delivery Ratio}$$

$$d_r = \text{Reverse or ACK Delivery Ratio}$$

Eq. 3.1

ML (Minimum Loss) focuses on utilizing links with minimum loss. As it was explained in Section 3.4, OLSR uses MPRs as a signalling optimization mechanism. OLSR is fundamentally designed around the idea of MPRs and how they are selected in the network. Since Source-destination routes are created based on MPRs, their selection has direct impact on network topology. There are different mechanisms to choose MPRs so that not only they would cover all two-hop neighbours of the node, but also minimize redundant link advertisement to two-hop neighbours. The works of [89, 93], gives us an overview of various MPR selection algorithms for OLSR. In OLSR’s modification based on ML metric, MPRs are chosen based on the links which have the minimum loss to the 2-hop neighbourhood of each node. On the other hand, OLSR-MD focuses on minimization of delay, hence chooses MPR’s from the links that have minimum delay to the current node.

### 3.5.1 Speed Variations

According to Figure 3-8 we can see that in terms of End-to-End delay, OLSR-MD, OLSR-ETX and OLSR-ML perform approximately better than the OLSR with minimum hop metric when the speed is lower than 20m/s. OLSR-MD performs much better than other protocols when compared to the base OLSR at speeds lower than 20m/s. As it was elaborated in Section 3.5, the Minimum Delay (MD) metric focuses on choosing routes which have the minimum delay and that is why the routes chosen by OLSR-MD result an overall reduction in end-to-end delay at the application-layer level. On the other hand, OLSR-ML also performs relatively well and very stable at different speeds. OLSR-ETX performs worse than the base-OLSR when no mobility exists and it performs better when the speed is raised from 5 to 20 m/s and then again suffers at speeds higher than 20 m/s. First, we should note that the confidence interval of all routing protocols is relatively high when no mobility is applied to the network. Hence, we have to be careful when we analyse the data in no-mobility case. All the three variations of OLSR have an upward trend in terms of end-to-end delay which is because all of them rely on link level probing mechanism to sense the quality of each individual link. As the speed increases, the fixed probing cannot keep up with the mobility of the network; as a result, the accuracy of the metrics on realization of links decreases and the end-to-end delay rises.
On the other hand, when we look at Figure 3-9, it can clearly be seen that choosing better routes in terms of end-to-end delay does not necessarily result lower PDR. If we have a close look at Figure 3-9, we would notice that PDR in OLSR-MD is higher compared to OLSR-ETX and OLSR-ML at all time. Furthermore OLSR-MD has a higher PDR compared to the base OLSR if the speed is lower than approximately 7 m/s. If we have a side by side look at Figure 3-8 and Figure 3-9, we can easily notice that the point where the performance of OLSR-MD degrades compared to the baseline OLSR (in terms of PDR) is where the end-to-end delay of OLSR-MD starts to increase rapidly; this shows that as the routing metric performs less accurately at higher speeds, the end-to-end delay starts increasing and as a result, low stability routes are chosen in the network and OLSR-MD’s figure in terms of PDR starts decreasing. Generally, end-to-end delay is measured over the successfully delivered packets, hence a lower PDR is an indication of the fact that the low-quality paths are not delivering much packets and the end-to-end delay is averaged over packets that traverse high quality paths.

As it can be seen in Figure 3-10, OLSR-MD applies a higher normalized routing overhead to the network compared to OLSR and other variations of it. OLSR, OLSR-ETX and OLSR-ML have relatively the same performance in terms of normalized routing overhead. As it is expected, by increasing the speed, the overhead of all protocols have an upward trend; which is due to higher re-routing triggers generated as a result of low PDR at higher speeds. The reason that OLSR-MD produces more overhead compared to other protocols is that it does not utilize OLSR’s HELLO messages for link sensing. OLSR-MD uses its own message transmission technique which triggers more updates through the network compared to other protocols. Although, this improves the accuracy of link sensing, it adds significantly higher load to the network compared to OLSR, OLSR-ML and OLSR-ETX.

![End-to-End Delay](image)

*Figure 3-8: End-to-End Delay versus Speed (OLSR, OLSR-ETX, OLSR-MD and OLSR-ML)*
3.5.2 Load Variation

The other category of performance analysis covered by this work is on OLSR, OLSR-ETX, OLSR-MD and OLSR-ML against variations of Application Layer load. We have analysed the performance of these protocols under different load conditions by gradually increasing application data rate at the source nodes. The simulations summarized here are based on variations of network load given that the speed is fixed to 3m/s. As it can be seen from the results shown in Figure 3-11, OLSR-MD still maintains a good end-to-end delay; this shows that the metric used by OLSR-MD maintains low end-to-end delay routes even at extremely high load scenarios. Another observation is that while in low load (from 1KB/s to
15KB/s) the end-to-end delay of all the protocols is approximately similar but under high load their performance is completely different. Under high load case, OLSR-ETX performance is the worst but OLSR-ML performs nearly as well as OLSR and OLSR-ETX. In terms of PDR, OLSR-MD can deliver more packets compared to all other protocols including OLSR itself. Based on the PDR analysis all protocols have a very mild downward trend which can be justified by the fact that a higher data rate results higher congestion and queue overflows which results higher packet loss. OLSR-ETX is the lowest performing protocol when analysed based on PDR.

**Figure 3-11: End-to-End Delay versus Load (OLSR, OLSR-ETX, OLSR-MD and OLSR-ML)**

**Figure 3-12: Packet Delivery Ratio Versus Load (OLSR, OLSR-ETX, OLSR-MD and OLSR-ML)**
Chapter 3: Spectrum-aware Routing

It can be seen in Figure 3-13 that as load increases, the normalized signalling load is decreased. That is due to the fact that signalling load is normalized to the number of delivered packets at each specific point. We can see that as the load increases, the normalized signalling load of OLSR-MD and the other three protocols converges to nearly the same level. Even though OLSR-MD results extremely high normalized overhead at low load condition, its overhead is relatively efficient at higher load conditions. This shows that OLSR-MD can be very efficient under highly loaded networks.

![Normalized Routing Overhead](image)

Figure 3-13: Normalized Routing Overhead Versus Load (OLSR, OLSR-ETX, OLSR-MD and OLSR-ML)

3.5.3 Conclusion on Performance of OLSR with Various Metrics (ETX, ML and MD)

Generally, it can be concluded that at high mobility scenarios, all metrics (ETX, ML and MD) struggle to efficiently sense the actual performance of the links due to the high frequency of topology changes. This results higher end-to-end delay and lower PDR at high speeds. Furthermore, based on the analysis given in this section, MD is a good metric when minimization of delay is the main aim of our optimization. It was observed that, the routes created by OLSR-MD at lower speeds are more stable than higher speeds. As a result, OLSR-MD results a consistently higher PDR with lower end-to-end delay at lower speeds compare to the baseline protocol OLSR. Of course, the good performance of OLSR-MD comes at the cost of high signalling load. While ML did not prove to be a better metric than MD and the baseline hop count but it shows a more stable performance compared to the other protocols in terms of end-to-end delay. The routes that are chosen based on ETX metric result better and more stable end-to-end delay at almost all speeds. ETX struggles to measure the channel quality at loads higher that 10KB/s which is due to the excessive interference affecting its probing mechanism. The final conclusion after
doing these analyses was that a routing metric which prioritizes links based on their delay history (e.g. MD) could potentially result in better end-to-end delay across the network. On the other hand, a routing metric which categorises links based on their loss history, could also result in good performance in terms of route stability but not necessarily a better end-to-end delay. The main conclusion that needs to be drawn from the results provided in this section is that hop-count used at the baseline OLSR cannot guarantee the best quality paths. As a result, the routes chosen by the baseline OLSR suffer from instability and lower end-to-end delay, both of which are very important QoS metrics.

3.6 Considerations of Spectrum-Aware Routing Metric

As it was elaborated in Section 2.3, IEEE 802.11 is the base of most CR-MANET implementations and analysis; however, it does not provide any support for multi-channel operation. On the other hand, the routing protocols analysed in Section 3.5 do not support any multi-channel mode of operation. Hence, all of the scenarios which were analysed in Section 3.5, are based on a single channel architecture. In Section 3.4, the reason why OLSR has the potential to be chosen as the base implementation of our spectrum-aware routing protocol was explained. As minimum hop-count (utilized in OLSR) is unreliable in evaluating the quality of individual links, we had to explore other metrics which provide a better quantitative view of quality of each link. Based on the fact that links in MANETs are simply wireless connection among two nodes over a specific channel, the terms link and channel are used interchangeably throughout the rest of this thesis. So, our conclusion on the performance of different routing metrics on the link level in Section 3.5, can be expanded to multi-channel scenarios; For example, the capability of OLSR-MD in minimization of delay in a single channel MANET scenario can be utilized to optimize end-to-end delay in a multi-channel DSA system too. Based on the analysis performed in Section 3.5 (as well as the definitions given), ETX is a trade-off of all the metrics and is suited for channel quality measurement which has been used as the spectrum-aware metric in this work. The algorithm proposed by this work utilizes ETX for channel quality (aka link cost) estimation which is built upon ETX probing technique; this ensures that SOPs are assessed and registered in the routing paths based on their quality which makes ETX the spectrum-aware metric of our choice.

3.7 Multi-channel, Multi-interface IEEE 802.11 Operating in Ad-hoc Mode

In Section 2.5.1 we covered that there are two approaches to target the dynamic channel structure of DSA systems, i.e. single-interface and multi-interface design. Based on a single interface design, nodes are required to perform channel switching in order to dynamically cover SOPs as a function of time and geographical location. It was covered under the literature review that a single interface design is prone
to deafness problem. Fundamentally, the limitation of a single interface design lies on the fact that when a data transmission takes place on a certain channel in the PHY layer, not only that interface is inaccessible for any other reception on the current channel (due to limitations of half-duplex PHY) but also the interface is fully blind to any nearby transmissions taking place on any other channels. In contrast, a multi-interface, multi-channel design could potentially eliminate the problems associated with the single interface design. Under a multi-interface design, each interface needs to be associated with a specific channel throughout the network lifetime. Hence, the transmission of data on one interface does not affect any simultaneous transmissions on another interface. As a result, upon creation of a channel opportunity (SOP), the interface is enabled and is utilized by the higher layer protocols and when the opportunity is removed, the interface stays idle until the next availability rises. Hence, throughout the rest of this chapter, we have the assumption of a multi-channel, multi-interface design for the design of our spectrum-aware routing protocol. Even though our design does not require any channel switching but we have implemented this capability in IEEE802.11 in order to analyse the effect of transition from a busy channel to a relatively free channel on the application layer data traffic. We have used this analysis as a ground basis for the rest of our spectrum-aware implementation.

Typical IEEE 802.11 networks allocate a single non-overlapping channel to all nodes which operate in ad-hoc mode and to the best of our knowledge there is no IEEE standardized protocol which targets multi-channel CR-MANETs. To enable progress toward the goals of this research we had to implement channel switching at PHY/MAC level for IEEE802.11 operating in ad-hoc mode. Our implementation was based on INETMANET 2.0 package of OMNET++ simulator. We had to define an interface which gives the routing protocol in network layer the capability to set channel switching triggers; as shown in Figure 3-14, these switching triggers had to be delivered to the MAC and PHY layers via a cross layering approach.

![Figure 3-14: Channel Switching Triggers Implemented in OMNET++](image)
Upon implementation of such channel switching mechanism we setup a simple simulation scenario to confirm its functionality. It must be noted that the channel switching capability was for the testing purposes and is not used in the final implementation of our spectrum-aware routing protocol.

As shown in Figure 3-15, the simulation scenario comprises of 4 nodes which are all in communication range of each other and run IEEE 802.11g interfaces. Node 1 sends UDP traffic to node 2 and node 3 sends a relatively higher load of traffic to node 4. All of the 4 nodes are tuned to channel 1 at the start of the simulation hence we have setup Node 3 and 4 to cause interference on node 1 and 2 on channel 1.

![Figure 3-15: Example simulation scenario to validate channel switching](image)

The purpose of this simulation scenario is to validate our channel switching mechanism implemented in IEEE 802.11 ad-hoc mode. Additionally, we had interest on the effect of channel switching on end-to-end performance of the data packets. We have triggered a channel switching at the 250s of the simulation time and monitored the end-to-end delay of UDP packets between node 1 and 2. As it can be seen from the result shown in Figure 3-16, from 0 to 250 seconds, the end-to-end delay suffers due to the interference caused by node 3 and 4 on channel 1 but after the channel switching is triggered (e.g. to a free channel 3), suddenly the end-to-end delay drops to the normal level. Furthermore, we have performed moving-average smoothing on the end-to-end delay to visualize the trend more clearly. This result confirms that our channel switching implementation functions properly. This simple scenario confirms that if the network layer takes control of the channels utilized in MACPHY layer then the end-to-end performance at the application layer can be improved.
3.8 Considerations of Signalling in Spectrum-aware Routing

One of the main challenges in the area of spectrum-aware routing in MANETs is signalling. As it was covered in Section 2.7.1, providing a reliable and stable signalling/control channel in such systems is very important. Considering the dynamic environment of the DSA systems, there should always be a reliable channel which can be used to distribute the signalling messages across the network. In order to provide a reliable solution to the problem of signalling, we have considered a fixed channel for signalling purposes which its usage for data transmission is restricted. In other words, our assumption is that one of the channels in IEEE 802.11 standard is strictly dedicated to signalling purposes. We have modified the HELLO and TC messages [85] in the base implementation of OLSR’s signalling mechanism to update the network with SOP information of each node in the network. As it was mentioned before, we use a multi-channel, multi-interface design for our DSA system. HELLO messages are responsible for link sensing and advertising the number of available channels/interfaces to the neighbouring nodes. Upon complete discovery of 1-hop neighbour’s channel information, the OLSR’s TC messages advertise these availabilities throughout the network. As opposed to some spectrum-aware routing protocols where the signalling is performed over all the available channels/links, we gather this information using HELLO messages and advertise it via TC messages throughout the network. It must be noted that
HELLO messages perform the link sensing over all their available interfaces and then the TC messages utilize the dedicated signalling channel to distribute the data that is processed via the HELLO messages. TC messages are responsible for majority of the signalling overhead in the network generated in the OLSR algorithm. This is due to the fact that TC messages are broadcasted throughout the network but HELLO messages are only sent to the 1-hop neighbourhood of the node. Utilization of the dedicated signalling channel assures that the broadcasts resulted by TC messages does not affect the data channels.

3.9 Solution to the Spectrum-aware Multi-Graph Problem

Graph theory provides very strong and consistent solutions to the problem of route computation in communication networks. The conventional graph oriented solutions to computation of shortest paths in the classical graphs theory does not provide a solution to multi layered graphs (aka multi-graphs). Multi-graphs or pseudo-graphs are defined as graphs which can have multiple parallel edges connecting any two adjacent vertices [94]. The classical Dijkstra’s algorithm is a shortest path route computation algorithm for conventional networks that has been around for many years. The conventional Dijkstra’s algorithm has not been optimized to compute the shortest weighted paths in multi-graphs. To the best of our knowledge the work introduced in [95] was the first time that the problem of finding shortest weighted paths using Dijkstra’s algorithm was generalized to multi-graphs. This work is the basis of our proposed, spectrum-aware routing algorithm in this chapter.

In Section 2.3 and 2.5 we explained that DSA, introduces possibility of multiple available SOPs among nodes of the network which results in more than one channel being available between any two nodes to communicate. Hence the networks in CR-MANETs follow this dynamic spectrum structure, which we have modelled it as multi-graphs; the reason being that such networks can have more than one link (aka. channels) available between any two nodes and this cannot be categorized under the classical graph theory. The work of [95], provides a promising solution to traversing multi-graphs via Generalization of Dijkstra’s algorithm (GDA). This work builds upon the classical Dijkstra’s algorithm and is based on extraction of minimum cost links among any two vertices. As a result, a multi-graph is converted into a normal graph and the classical Dijkstra’s algorithm for computation of shortest paths is then applicable to it.

As it was concluded in Section 3.4, OLSR was selected as the candidate for the base implementation of our spectrum-aware routing protocol. In Section 3.5, we concluded that different routing metrics can have different impacts on OLSR and analysed these impacts in detail. The missing puzzle from the main aim of this work, which is a spectrum-aware routing protocol, is integration of GDA in OLSR and harmonizing it based on a suitable routing metric. We have fully implemented GDA which was introduced in [95] into OLSR based on ETX as the weight metric. Although the implementation is based
on ETX, the other metrics such as MD or ML can alternatively be applied. We have provided the flowchart of our algorithm in Figure 3-17. This algorithm is triggered every time a new TC/HELLO message is received at any node in the network. Our assumption is that based on the spectrum-aware signalling mechanism explained in Section 3.8, nodes perform distributed proactive signalling to update multigraph topology of the network. According to our proposed spectrum-aware algorithm, initially the SOPs to 1-hop neighbours of the current node are filtered based on the cost metric (ETX in our algorithm). The information related to 1-hop neighbourhood of the node is mainly supplied by the HELLO messages. Next, the network topology from the current node’s perspective is analysed. Under the basic operation of OLSR, topology tuples are identified based on the (destination_address, last_address) pairs. Hence, every destination address in the network can be reached via a last address and these segments create the full topology of the network. Nodes in the network are identified by their IP address and the interfaces with their unique MAC address. Essentially, the association of a (destination_address, last_address) pair is based on channels also known as SOPs in the CR topic. Hence the algorithm uses these pairs to find all the alternative links between every two node pairs in the network and then finds the best cost among them using an iterative process. By this point in the proposed algorithm, the multi-layered topology graph is simplified to a single layer graph which then the shortest weighted path can be computed using the conventional Dijkstra’s algorithm. As this algorithm runs every time new network refresh messages (HELLO and TC) arrive, this simplification would have minimal impact on the accuracy of the computed routes. However, as OLSR which is the basis of our implementation, performs route computation on a proactive basis, this could essentially result outdated routes in between routing refresh signalling messages. This is addressed by reactive signalling messages which are sent by link layer when a hop by hop link breaks. Under such situation, a new network signalling would update outdated routing tables and our algorithm is used to re-compute the routing tables. Due to the reason explained here, the simplification performed by our algorithm can be seen as a shortcoming of this work.
### Chapter 3: Spectrum-aware Routing

#### Figure 3-17: Flowchart of GDA Integrated in Route Discovery Phase of OLSR

![Flowchart](image)

#### Figure 3-18: An Example Scenario, Applying GDA Graph Transformation

![Graph Transformation](image)
As a scenario, we have randomly created the multi-graph shown in Figure 3-18 (on the left) and applied our implementation of GDA transformation algorithm and the result was the graph on the right. We successfully confirmed, in many random multi-graph samples, that our algorithm actually chooses the minimal weight links between any node pair and is capable of simplifying a multi-graph to a classical graph. Finally, by applying Dijkstra’s classical shortest weight algorithm, the minimal weight routes between any two source/destination pairs can be found.

### 3.10 Performance Analysis on the Proposed Algorithm

#### 3.10.1 Initial Simulation Results (Proof of Concept)

In this section, we have used the channel switching implementation covered in Section 3.7 to perform an initial analysis on a simplified abstract version of our algorithm. The analysis provided in this section is based on a 2-interface design which is mainly provided as a proof of concept for the spectrum-aware routing idea. It must be noted that the algorithm that was covered in Section 3.9 is analysed in Section 3.10.2 of the result section.

Each node is assumed to have two individual IEEE 802.11 interfaces, one for the signalling and the other for the data traffic. As a result, at each time instance, a node has a pool of channels/SOPs available to perform the round-robin ETX based channel quality estimation which is in line with the proposed Graph Extraction algorithm discussed in Section 3.9. As it was explained in Section 2.5.1, there exist three general approaches of centralized, distributed and hybrid in the context of spectrum management in DSA systems. Our routing algorithm is based on a hybrid spectrum management architecture; the reason behind this being, existence of PUs are reported centrally from the spectrum-broker unit over the signalling channel but the decision on suitability of each individual channel is the responsibility of the routing protocol and is performed on a distributed manner.

We have setup a simulation scenario as shown in Figure 3-19. Based on this scenario, the source nodes route their UDP data packets to the destination nodes via two relays which are $R_1$ and $R_2$. Source, destination and relay nodes are all capable of channel switching based on our implementation in Section 3.7.
To simulate different channel conditions, we have setup 8 interfering nodes as shown in Figure 3-19, which are named $I_1$ through $I_8$. Only $R_1$ and $R_2$ are affected by the interfering nodes and due to distance, Source and Destination are not affected by this interference. We have set nodes $\{I_1, I_2, I_3, I_4\}$ (Group A) to send random UDP traffic ranging from $\{250\text{KBps to 350KBps}\}$ which is considered as high load for MANETs. Furthermore nodes $\{I_5, I_6\}$ (Group B) are set to generate lower random UDP traffic ranging from $\{80\text{KBps to 150KBps}\}$. Additionally, nodes $\{I_7, I_8\}$ (Group C) generate their traffic ranging from $\{30\text{KBps to 70KBps}\}$ which is considered as low load in our simulation scenario. Toward creating three different bursty channel conditions, we have set group A to operate on channel 1, group B on channel 2 and Group C on channel 3. At the start of the simulation source node, destination node and both relays are set to channel 1. We have set interfering nodes group A to start their transmission on channel 1 at 500s of the simulation time. Furthermore, group B and C create bursts of traffic on channel 2 and 3. The routing protocol has the responsibility of detecting the interference caused by Group A and switching to the next available (not occupied by PUs) channel based on a round-robin manner. The quality of each channel is measured by the ETX metric over an interval $T$ which in these simulations is set to 10 seconds. As a result, the mean ETX value of the end-to-end path from source to destination on each individual channel is measured over a time period of 10 seconds and if this value passes a threshold represented as $ETX_T$ (which sets minimum QoS), another channel switching is triggered. This process continues until a channel’s mean end-to-end ETX falls below the $ETX_T$ threshold. The rest of simulation parameters are listed in Table 3-2.

Figure 3-19: Spectrum-aware routing scenario
### Table 3-2: Spectrum-aware Simulation scenario parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulation Time</td>
<td>1000s</td>
</tr>
<tr>
<td>Number of repeats</td>
<td>10</td>
</tr>
<tr>
<td>Number of Interfering nodes</td>
<td>8</td>
</tr>
<tr>
<td>Number of Relays</td>
<td>2</td>
</tr>
<tr>
<td>Mobility</td>
<td>Static</td>
</tr>
<tr>
<td>Number of Flows</td>
<td>6</td>
</tr>
<tr>
<td>UDP Application</td>
<td>CBR</td>
</tr>
<tr>
<td>Application Data Rate</td>
<td>{I_1, I_2, I_3, I_4}, Group A</td>
</tr>
<tr>
<td>{250KBps to 350KBPs}</td>
<td>{250KBps to 350KBPs}</td>
</tr>
<tr>
<td>Application Data Rate</td>
<td>{I_5, I_6}, Group B</td>
</tr>
<tr>
<td>{80KBps to 150KBPs}</td>
<td>{80KBps to 150KBPs}</td>
</tr>
<tr>
<td>Application Data Rate</td>
<td>{I_7, I_8}, Group C</td>
</tr>
<tr>
<td>{30KBps to 70KBps}</td>
<td>{30KBps to 70KBps}</td>
</tr>
<tr>
<td>MAC Wireless Protocol</td>
<td>IEEE 802.11g</td>
</tr>
<tr>
<td>RTS Threshold</td>
<td>2346Bytes</td>
</tr>
<tr>
<td>MAC Bitrate</td>
<td>54Mbps</td>
</tr>
<tr>
<td>MAC Retry Limit</td>
<td>7</td>
</tr>
<tr>
<td>PHY Frequency Band</td>
<td>2.4GHz</td>
</tr>
<tr>
<td>PHY Transmission Power</td>
<td>30mW</td>
</tr>
<tr>
<td>PHY Path Loss Alpha</td>
<td>2.4</td>
</tr>
<tr>
<td>PHY Propagation Model</td>
<td>Free Space Model</td>
</tr>
<tr>
<td>PHY SNIR Threshold</td>
<td>4dB</td>
</tr>
<tr>
<td>PHY Thermal Noise</td>
<td>-110dBm</td>
</tr>
<tr>
<td>PHY Radio Sensitivity</td>
<td>-90dBm</td>
</tr>
<tr>
<td>TX_range</td>
<td>255m</td>
</tr>
<tr>
<td>PCS_range</td>
<td>255m</td>
</tr>
</tbody>
</table>

Figure 3-21 shows variations of the End-to-End ETX performance (from source to destination) over simulation time. We can see that after 500s which refers to the time where interfering Group A start their transmission, the interference is detected by ETX within the 10 second monitoring period and the channel has been switched to the next available channel which is channel 2. After switching to channel
2, the source destination route experiences interference of Group B which again can be seen in Figure 3-21.

![Figure 3-20: End-to-End ETX vs. Simulation Time (s) (Without Channel Switching)](image)

![Figure 3-21: End-to-End ETX versus Simulation Time (s) (With Channel Switching)](image)

We have set the $ETX_T$ to 15 in this simulation scenarios which shows that if the number of re-transmissions from the source to destination is estimated to be more than 15 times (definition of ETX) then a channel switching should be triggered. As in our simulation the ETX after switching to channel 2 fluctuates at about 19 so another channel switching is triggered to the next available channel which is channel 3. We can see that after switching to channel 3, suddenly ETX value drops to about 10 which satisfy the minimum QoS requirement of 15 in our case and the simulation continues on that channel without any problem.
Now if we compare the result from Figure 3-21 (with switching) with Figure 3-20 (without channel switching) we can clearly notice the performance gain.

Furthermore, we have included the end-to-end delay of the network from source to destination measured at application layer in Figure 3-22. As it can be seen, the two spikes in end-to-end delay are when the channel switching actually happens. This delay is the result of queued buffered packets which are already in the sending buffer being queued up waiting to be sent. We can see that the end-to-end delay remains stable after two consecutive switching.
Additionally, in Figure 3-24 we have provided the MAC Delay measured at MAC layer which is averaged over the 4 nodes of source, destination and the two relays. We have also provided the averaged MAC Delay when no channel switching happens in Figure 3-23 as a comparison reference. It is clear that the channel switching has exceptionally actually improved the MAC Delay.
3.10.2 Comprehensive Analysis on the Proposed Algorithm

The concentration of this section is to analyse the performance of the GDA algorithm proposed in Section 3.9 in a fully functional routing algorithm. In Section 3.10.1 we have shown that our proposed algorithm operates well under a controlled simulation scenario where nodes are static and limited in numbers. The scenario that was detailed in Section 3.10.1 was mainly to proof that an end-to-end path in the network can benefit from channel availabilities of the intermediate hops and by utilizing the SOPs we can achieve a performance gain in the end-to-end routes. As it was detailed before, the GDA algorithm was fully integrated in OLSR as the baseline of the implementation. The signalling mechanism in OLSR has been modified to support the proposed spectrum-aware signalling mechanism covered in Section 3.8. Our assumption is that nodes are equipped with multiple interfaces and each interface is tuned to a channel (aka SOP) according to the discussion provided in Section 3.7. In this section, we have analysed the performance of the proposed spectrum-aware OLSR algorithm based on variations of speed, relay node density and network load. OLSR has been enhanced with all the contributions explained in this chapter which are firstly the conclusion we made about a spectrum-aware routing metric in Section 3.6; secondly the implementation of a multi-channel, multi-interface design in Section 3.7; thirdly the spectrum-aware signalling which was explained in section 3.8; and finally the main contribution of this work which is integration of GDA into OLSR that was covered in Section 3.9. The result of all these enhancements is a new routing protocol, called spectrum-aware OLSR. The proposed protocol utilizes SOPs in the route computations which results more stable routes that support higher data capacity and better QoS.

In this section, we have analysed the performance of our implementation of Spectrum-aware OLSR under two conditions of Speed variations and Relay Node Density variations (Sections 3.10.2.1 and 3.10.2.2). To create a realistic CR environment, we have programmed a simple PU Activity Generator (PAG) which in simple words generates a subset of channels (ON/OFF model) which are available to the secondary users (consisting of both active and relay nodes) in our simulation scenario. Our implementation of PAG follows the ON-OFF model provided in the work of [96]. At each time window PAG produces a number of channels (SOPs) which are locally available to each individual SU in which case their availability is subject to an expiry time. Further into detail, at each time instance, the number of available channels and their expiry time are generated randomly based on a normal distribution. The simulation scenario is set similar to the benchmarking results provided in Section 3.3. We have 6 sources and 6 destinations i.e. 12 active host (SUs), which are distributed randomly in the simulation area. The 6 sources constantly send UDP traffic from the start of the simulation. We have defined 15 relay hosts which are distributed randomly in the simulation area to assist active hosts in relaying data in the network. As a result, active hosts (or SUs) are supposed to use our spectrum-aware OLSR to route their data through the generated SOPs by PAG, to each other; this results a multi-graph which uses GDA to
locally find the best routes in the network. As before, the simulations were performed in OMNET++
event based simulator. These analyses were performed based on variations of node’s velocity and relay
node density. The aim is to analyse performance of spectrum-aware OLSR and the baseline
implementation of OLSR in terms of End-to-End Delay (E-t-E Delay), Packet Delivery Ratio (PDR)
and Normalized Routing Overhead. To provide reliable results and improve confidence levels, we set
the transient interval (warm-up period) to 700s (total sim. Time = 2000s) and report averaged results
over 15 runs with different seed sets. The simulation parameters are summarized in Table 3-3.

Table 3-3: Simulation Parameter, spectrum-aware OLSR

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulation Time</td>
<td>2000s</td>
</tr>
<tr>
<td>Transient Interval</td>
<td>700s</td>
</tr>
<tr>
<td>Number of repeats</td>
<td>50</td>
</tr>
<tr>
<td>Simulation Area</td>
<td>2000m*2000m</td>
</tr>
<tr>
<td>Active Hosts (Default)</td>
<td>12</td>
</tr>
<tr>
<td>Relay Hosts (Default)</td>
<td>20</td>
</tr>
<tr>
<td>Maximum SOP availability (Default)</td>
<td>12</td>
</tr>
<tr>
<td>Mobility Model</td>
<td>Random Waypoint</td>
</tr>
<tr>
<td>Mobility Initial X and Y positions</td>
<td>Random</td>
</tr>
<tr>
<td>Mobility Speed (Default)</td>
<td>5m/s</td>
</tr>
<tr>
<td>Mobility Wait Time</td>
<td>Random [3s, 5s]</td>
</tr>
<tr>
<td>Number of Flows</td>
<td>6</td>
</tr>
<tr>
<td>UDP Application</td>
<td>CBR</td>
</tr>
<tr>
<td>Application Data Rate (Default)</td>
<td>10KB/s</td>
</tr>
<tr>
<td>MAC Wireless Protocol</td>
<td>IEEE 802.11g</td>
</tr>
<tr>
<td>RTS Threshold</td>
<td>2346Bytes</td>
</tr>
<tr>
<td>MAC Bitrate</td>
<td>54Mbps</td>
</tr>
<tr>
<td>MAC Retry Limit</td>
<td>7</td>
</tr>
<tr>
<td>PHY Frequency Band</td>
<td>2.4GHz</td>
</tr>
<tr>
<td>PHY Transmission Power</td>
<td>20mW</td>
</tr>
<tr>
<td>PHY Path Loss Alpha</td>
<td>2.6</td>
</tr>
<tr>
<td>PHY Propagation Model</td>
<td>Rayleigh Fading Model</td>
</tr>
<tr>
<td>Shadowing Model</td>
<td>Constant</td>
</tr>
<tr>
<td>Shadowing Mean</td>
<td>4.0 dB</td>
</tr>
<tr>
<td>PHY Thermal Noise</td>
<td>-110dBm</td>
</tr>
<tr>
<td>PHY Radio Sensitivity</td>
<td>-90dBm</td>
</tr>
<tr>
<td>TX_range</td>
<td>200m (Max.) (Rayleigh Fading Model)</td>
</tr>
<tr>
<td>PCS_range</td>
<td>250m (Max.) (Rayleigh Fading Model)</td>
</tr>
</tbody>
</table>
3.10.2.1 Speed Variations

In this section, we have compared the performance of Spectrum-aware OLSR and the baseline implementation of OLSR (with ETX metric) versus variations of speed/velocity. The Spectrum-aware OLSR has been analysed under 4 ranges of channels availabilities {3, 6, 9, 12}. These numbers dictate the maximum number of channels available to nodes in the network. However due to the PU activity the actual number of channels available is a function of time and location in the simulation area which is modelled via an ON/OFF PU algorithm.

In Figure 3-25 we can see that on an overall basis, performance of all protocols follow an upward trend against variations of speed. This can be explained based on the fact that at higher speeds a proactive signalling mechanism in both OLSR-ETX and OLSR-SA cannot keep up with frequency of changes in the network. Another overall conclusion from the graphs shown in Figure 3-25 is that the higher number of channel availabilities in the OLSR-SA has resulted better end-to-end delay performance. This shows that the spectrum-aware mechanism in our proposed algorithm utilizes the channel opportunities in order to provide better quality paths which support better QoS. As speed increases, mobility of the nodes adds to the instability of the network which results worse End-to-End delay compared to the baseline OLSR. When analysing the PDR performance of the baseline OLSR-ETX with OLSR-SA in Figure 3-26, it can be concluded that more channel availability has resulted higher PDR at all speeds.

![End-to-End Delay VS Speed (OLSR-ETX and OLSR-SpectrumAware)](image)

This is due to the fact that SOPs provide alternative paths for routing the data in the network which as it was discussed before, this leads to increased network capacity; this proves that our implementation of GDA into OLSR has resulted an improvement in the performance of the protocol. The gain comes from the fact that spectrum-aware OLSR is capable of intelligent utilization of SOPs provided by PUs
based on ETX as a quality metric. Obviously, this gain comes at the cost of signalling overhead. As shown in Figure 3-27 the normalized routing overhead increases as speed increases. The rate at which the OLSR-SA generates signalling messages in the network is higher than the baseline OLSR and that is due to the fact that spectrum-aware OLSR is responsible for updating not only the network topology but also the channel availabilities across all nodes in the network. As it was fully explained in section 3.8 and tested in section 3.10.1 we are utilizing ETX probes as a mechanism to evaluate individual SOP quality. Since the probing is performed on a per channel basis, so the added signalling load is dependent on the number of available SOPs.

Figure 3-26: PDR VS Speed (OLSR-ETX and OLSR-SpectrumAware)

Figure 3-27: Routing Overhead (Norm.) VS Speed (OLSR-ETX and OLSR-SpectrumAware)
3.10.2.2 Relay Node Density Variations

In this section, we have compared the performance of Spectrum-aware OLSR (OLSR-SA) and the baseline implementation of OLSR (with ETX metric) based on variations of Relay Node Density. Similar to the speed variation scenarios, OLSR-SA has been analysed under various ranges of channel availability to analyse the strength of our proposed spectrum-aware algorithm in utilizing the available SOPs.

According to the graph shown in Figure 3-28, on an overall basis, a higher RND results more collisions and interference resulted by transmission of signalling and data messages which degrades the end-to-end delay performance of the network. It can be noticed that higher channel availability in OLSR-SA results better end-to-end delay. To the extent that the end-to-end delay of OLSR-SA, Ch = 9 and 12 show a very stable trend against variations of RND. This is due to the fact that OLSR-SA is capable of utilizing channel availabilities in all the added relay nodes in the network which results computation of optimized routing paths. Generally, the higher relay node density results higher signalling and data relaying which is why in Figure 3-28 the End-to-End delay follows an upward trend for both OLSR baseline and spectrum-aware. While increasing node density is considered as a cost for the network when we look from the end-to-end delay point of view, it is considered as benefit in terms of the packet delivery ratio; this is due to the fact that GDA utilizes the extra relay nodes as opportunities to choose better routing paths (with consideration of ETX metric) through the available SOPs. According to Figure 3-29, the performance of OLSR-SA is fairly stable with respect to variations of RND which proofs that our proposed routing algorithm utilizes the channel availability to stabilize routes in the network.

Figure 3-28: End-to-End Delay vs RND (OLSR-ETX and OLSR-SpectrumAware)
Generally, ETX can be considered as a stability routing metric which prioritizes good quality links over lossy ones. We can observe in Figure 3-30 that as RNDs increases, normalized routing overhead also increases. The normalized routing overhead in spectrum-aware OLSR rises sharply when RNDs move toward 50 which is due to added number of signalling messages which are generated by the extra relays in the network and the low PDR observed at these levels.

Figure 3-29: PDR vs RND (OLSR-ETX and OLSR-SpectrumAware)

Figure 3-30: Routing Overhead (Norm.) vs RND (OLSR-ETX and OLSR-SpectrumAware)

3.10.2.3 Network Load Variations

In this section, we have compared the performance of Spectrum-aware OLSR (OLSR-SA) and the baseline implementation of OLSR (with ETX metric) based on variations of network load. Similar to
the speed variation scenarios, OLSR-SA has been analysed under various ranges of channel availability to analyse the strength of our proposed spectrum-aware algorithm in utilizing the available SOPs.

By looking at Figure 3-31, it can be observed that on an overall basis, higher network load results higher end-to-end delay in all protocols. This can simply be explained by the fact that the higher network load causes interference and congestion which leads to queuing and unbalanced load distribution which has a major impact on the end-to-end performance of the network. Based on the graph provided in Figure 3-31, the performance gain achieved by OLSR-SA can be clearly observed.

![Figure 3-31: End-to-End Delay vs Load (OLSR-ETX and OLSR-SpectrumAware)](image)

The higher, the number of channel availabilities for SUs results more network capacity and consequently less congestion. As a result, the end-to-end delay is improved at OLSR-SA with higher channel availabilities. The same effect can be seen for PDR according to Figure 3-32. The higher PDR
is a clear indication of higher capacity in the network resulted by higher SOPs. As it was detailed before, OLSR-SA achieves these performance gains at the cost of higher signalling which is clearly reflected at the graph of routing overhead shown in Figure 3-33.

![Figure 3-33: Routing Overhead (Norm.) vs Load (OLSR-ETX and OLSR-SpectrumAware)](image)

### 3.11 Summary

Cognitive Radio (CR) provides a new architecture to efficient utilization of spectrum bands. In this work, we have provided a thorough analysis on the state of the art techniques targeting spectrum management and routing in CR-MANETs. We focused on the close interconnection of routing with CR-MANETs and summarized challenges involved in this area. Towards the aim of implementing a spectrum-aware framework we performed simulation analysis on different routing protocols designed for MANETs and justified OLSR as a good baseline candidate for the implementation of our proposed spectrum-aware routing protocol. Furthermore, we performed simulation studies on three of the state-of-the-art routing metrics which are potentially the best to target spectrum-aware routing and analysed their performance against the base implementation of OLSR with shortest hop metric. The problem of spectrum-aware route computation was modelled as a multi-graph and a route extraction algorithm was developed and implemented to solve the multi-graph problem based on GDA. In order to efficiently distribute SOP information throughout the network, a signalling mechanism was introduced based on modification to the OLSR’s base signalling messages. Finally, the performance of the proposed OLSR-SA routing algorithm was analysed against variations of speed, network load and RND based on the three metrics of end-to-end delay, PDR and normalized routing overhead. Is was concluded that the higher number of SOPs increase the capacity of the network and results higher PDR. The spectrum-
aware algorithm successfully utilizes SOPs in the computed end-to-end routes and achieves significant performance gains in terms of end-to-end delay. However, the performance gains come at the cost of higher signalling overhead. On the other hand, based on the simulation results, it can be observed that at higher SOP availability, the range of error bars have significantly increased which reflects another shortcoming of our algorithm. This is due to the fact that with the added spectrum opportunities there are more routes available to the routing algorithm which results a higher error during the averaging process. When we look at the overall results achieved from our algorithm in this chapter, we can see that the achieved performance gain through utilization of SOPs in the network is not as stable as anticipated. This indicates that the computed routes lack stability and persistency. We see this as a shortcoming of our algorithm, which concentrates on maximization of performance through aggressive and opportunistic usage of SOPs in the computed routes. This results bursts of performance gains which cannot be maintained. The lack of stability in our algorithm proposed in this chapter is our motivation for the next contribution which is outlined in the next chapter.
Chapter 4

4 Backpressure Spectrum-aware Routing

4.1 Introduction

This chapter introduces the second contribution of this research. In the previous chapter, it was shown that an aggressive utilization of SOPs in the route computation can have bursts of performance gains which has visualised in our results as higher spread of error bars. Although this does not undermine the performance gain of the proposed spectrum-aware route computation algorithm, but it highlights the shortcoming in terms of stability in performance of this algorithm. Stability plays a crucial factor in route computation as it would affect functionality of delay sensitive applications in communication networks. On the other hand, while spectrum-aware OLSR can have a positive impact on PDR, it has a negative impact on the end-to-end delay which is a very important QoS metric. Our analysis indicates that, opportunistic utilization of SOPs in our spectrum-aware route computation algorithm has resulted unstable routes to be created. Although these added routing opportunities increase network throughput and PDR on an aggregate level, but our simulation results indicate that they have impacted the average end-to-end delay. This is mainly because, failed packet deliveries resulted by link break along the computed routes triggers regeneration of routing refresh messages and consequently route re-computation which ends up adding delay to end-to-end delivery of packets. Hence, maintaining stability in the computed routes not only results maximization of throughput but also lowers end-to-end delay on an average basis. This idea was the main motivation of the contribution made in this chapter. As discussed in chapter 2, back-pressure routing can improve load balancing and throughput optimality in the network. Also, as discussed in chapter 2, one of the main drivers of DSA systems under the umbrella of CR-networks, is fair and efficient usage of the valuable/ limited (and often under-utilized) spectral resources. SOPs are merely extra transmission opportunities in the CR-MANETS. One of the advantages of backpressure routing is that under high network load conditions, it maximizes balanced utilization of all network links. On the other hand, as it was covered in Section 2.10, load balancing can potentially play a vital role in QoS provisioning in ad hoc networks. The main motivations for our research into designing a spectrum-aware routing protocol are firstly to maximize utilization of SOPs and secondly maintaining a minimum QoS level in the offered routes in the network. On a fundamental level, backpressure routing balances queue gradients in all contributing nodes in the communication networks. Most of network instabilities are resulted by over-utilization of some resources in certain segments of
the network relative to underutilization of such resources in other parts. Overly utilized links result in excessive interference in the nearby nodes, unbaling them from accessing such resources. Backpressure algorithm can theoretically minimize this by distributing packets across all nodes in the network via distribution of queues. Balanced queues can maximize delivery of packets across all SOPs and minimize irruptive bursts of performance gains and losses. The stability that can be achieved by load balancing would result better end-to-end delay whilst maintaining a high network throughput. Theoretically, these can be achieved via integration of backpressure routing into the earlier proposed spectrum-aware OLSR routing algorithm and this is the motivation for the work presented in this chapter, where we begin by summarizing the challenges involved in implementation and integration of backpressure routing into our spectrum-aware routing algorithm. Then the formulation of the problem followed by the applied methodology are presented. Lastly, simulation based evaluation of the proposed solution is used to confirm the validity of the selected approach, based on the gains achieved.

4.2 Highlight of Literature Study in Contrast with the Contribution

In this section, we have highlighted two of the main works in the literature [97] and [98] that are the most relevant to the contribution made in this chapter. This is mainly to highlight the main challenges in the literature and contrast the contribution made in this work to address such challenges. Additionally, the simulation results in this Chapter have been analysed against these works.

ROSA (ROuting and dynamic Spectrum Allocation) [97] is a spectrum-aware routing protocol which utilizes backpressure queueing model based on a cross-layer design. The mathematical modelling in this work initially suggests that at each time instance a full network knowledge is required to centrally perform the task of routing and scheduling. As this assumption is infeasible, the formulation is then simplified to a distributed model which is no longer throughput optimal based on the backpressure queue gradients. Furthermore, the work models routing and scheduling in two separate scope where the routing algorithm does not directly incorporate the scheduling decisions made by the MAC layer. The problem that can be seen here is that routing algorithms by default, contend for the lower cost path to the destination, and the backpressure formulation attempts to utilize links that maximize queue differential backlog. These two strategies could potentially be in contradiction in the computed routes that aim for directing the packets from source to destination. This can result routing loops which can have a negative impact on end-to-end delay of delivered packets and at worst case scenario could have negative impact on network throughput due to added interference by the excessive looping effect. Furthermore, it is suggested that routing and scheduling can be achieved via single interface with the assumption of a CSMA-CA MAC design at lower layer. However, this assumption does not consider the deafness problem discussed in Section 2.5.1 based on the work presented in [41]. To adaptively reduce collision in MAC layer, this work has defined 2 fine-tuning parameters which is used in the cross-layer design.
Chapter 4: Backpressure Routing

However, as optimization of these parameters, based on dynamic network size and node capacity, is a computationally complex problem to solve, they have only optimized it based on their fixed network simulation size. The results presented by this work where the performance of the algorithm is compared with and without backpressure component shows a negligible gain in terms of throughput and average delay. This is due to the contradictory strategies which has limited the gain that is expected to be achieved by utilization of backpressure algorithm into a spectrum-aware routing protocol.

A distributed backpressure scheduling with opportunistic routing algorithm is proposed by the work presented in [98]. This work utilizes the backpressure algorithm by analysing Lyapunov drift (as discussed in Chapter 2 - Section 2.11) as an attempt to achieve throughput optimality. Furthermore, it proposes a distributed MAC algorithm to address challenges involved in distributed scheduling in CR networks. The formulation of this work, similar to [97] tries attempt to achieve throughput optimality by tuning the CSMA-CA structure MAC Contention window size. But as increasing size of queue in each node is not bounded, it is concluded that throughput optimality cannot be achieved. Hence, the work argues that by limiting the per-node queue size to a certain value, the per link probability of success can be calculated and hence the backpressure algorithm can be applied. However, the dependency of this limit to network capacity, network size and traffic load is not mathematically analysed. In a larger network size (and/or higher network load) than the one analysed by the author, this limit can cause excessive arrival packet-drop which considering a CSMA-CA structure MAC, this would result excessive retransmission. The retransmission would directly impact the contention size as collisions result longer back off periods. To the best of our knowledge this work has not addressed the highlighted problem and that is the reason why the performance results suggests that the algorithm has zero tolerance to congestion. Furthermore, it is suggested that the RTC/CTS (Request To Send/Clear To Send) mechanism of IEEE802.11 is disabled in the simulation study which can result excessive packet drop and link breakage. This would add the delay cost of re-computation of routing tables and lower throughput. The simulation assumptions in this work doesn’t mention the number of secondary users, however the presented results suggest that the number of secondary users are very limited. The packet loss rate analysed shows that, at higher packet arrival rate, the network throughput goes down to zero which suggests that the algorithm has zero tolerance to congestion. This is an expected drawback resulted by the queue length limiting factor used by this work. The work doesn’t address the PU/SU coexistence in their network model which is the main subject of CR networks. From the routing algorithm design perspective, relaying packets based on backpressure queue gradients would be in contradiction with the shortest hop route computation. However, this matter is neither addressed in [97] or [98].
In the next few sections the details of implementation and challenges of the contribution made in this chapter is highlighted. Then, in Section 4.10 we provide a comparison of our contribution compared to the two proposed algorithms in the state of the art which was presented in this section.

4.3 Implementation Challenges

The simulation platform used in this work to implement the backpressure algorithm is OMNET++ which is an event based simulator. One of the main challenges in implementation of a backpressure algorithm was the single queue structure of IEEE802.11 MAC layer. As explained in Section 2.11.1, in conventional backpressure scheduling (which aims at performing both link and packet scheduling assuming a TDMA MAC) when the rate of generation and/or arrival of data packets at a node is greater than the forwarding/delivery rate, this results in creation of packet queues/backlogs at the MAC layer. Depending on the rate of traffic injection into the network and the capacity of the links, backlog formation is an almost inevitable consequence. The main research problem is to come up with a queue management system which leads to fair utilization of resources in the network. Backpressure is proven to be throughput optimal but suffers from major delay problems. It was discussed in Section 2.11 that the delay is the result of (on an average basis), routes being computed by backpressure taking longer paths in the network in order to balance the network traffic. The main challenges in implementation of backpressure spectrum-aware OLSR as summarized below.

i. Backpressure routing requires a multi-queue structure for the proper operation of the algorithm. Hence changing the queue structure of the MAC layer in IEEE802.11 from a single queue to a multi-queue structure was one of the main challenges of this work. When packets are generated by the current node or arrive at the current node from other nodes in the network, they are required to be filtered and distinctively stacked at the appropriate queue based on their destination.

ii. Implementation and integration of the backpressure algorithm into the multi-channel structure of the currently implemented spectrum-aware OLSR covered in Chapter 3 of this thesis was one of the main challenges in this research. The whole idea of spectrum-aware routing is built around maximizing utilization of SOPs. Additionally, our interest in backpressure routing is based on its throughput optimality which translates to maximization of channel usage efficiency. Hence, integration of backpressure algorithm into the spectrum-aware OLSR is expected to maximize utilization of SOPs.

iii. Utilization of the weighted graph structure of spectrum-aware OLSR to minimize the end-to-end delay in backpressure algorithm.
4.4 Weighted Back-pressure Routing

In Chapter 3 we summarized the first contribution of this thesis which was spectrum-aware OLSR. ETX was used as a weight metric to evaluate quality of links in the network. Due to the mathematical formulation of ETX, this metric is a good measure of link stability. As it was discussed before, the choice of ETX as the metric for quality evaluation of channels was mainly due to the fact that route stability plays a very important role in CR-MANETs. There are various works done in the literature with the aim of integrating shortest path metric (aka hop count) into the backpressure routing/scheduling algorithm [99-101]. It was discussed in Section 3.6 that the choice of an appropriate routing metric for spectrum-aware routing is very important. Hop count is one of the most popular routing metrics used is many conventional routing protocols. While hop count is very simple to implement, but it is proven to be inaccurate in many network scenarios. In particular, a lower hop count does not necessarily reflect a better-quality route in the network. An example to support this argument is that, a longer routing path (with higher number of hops) that have more stable links is preferable over a short path with unstable low-quality links. Hence, instead of unifying shortest-path with the backpressure routing algorithm which are the focus of the works presented in [100] and [101], in this work we have focused on the idea of a novel weighted back-pressure spectrum-aware routing. The work presented in [102] provides a weighted back-pressure approach which takes into account the link weights in the routing/scheduling process. While our problem formulation is very similar to this work, but our solution to integration of link weights into the back-pressure algorithm is completely different. The aim is that our spectrum-aware OLSR algorithm can benefit from the throughput optimality of backpressure and integration of the weighted path selection improves the end-to-end delay and route stability in the network. A few weight metrics used to evaluate quality of links in the network was covered in Section 3.5; ETX was used as the metric of choice in the design of our spectrum-aware routing protocol in Chapter 3 which is also used in our weighted backpressure spectrum-aware algorithm in this work.

4.5 CSMA versus TDMA MAC in Backpressure Routing

As it was discussed before, the original back-pressure algorithm presented in [73] was designed based on a TDMA MAC layer. Originally, backpressure algorithm was proposed to target load balancing in multi-hop networks. Ad hoc networks are a category of multi-hop networks which can benefit the most from the load balancing properties of backpressure algorithm. However, IEEE802.11 that is one of the most popular MAC/PHY standards used in researches involving ad hoc networks has a CSMA/CA access method. The work of [103] argues that, backpressure algorithm is throughput optimal when joined with a TDMA scheduling mechanism and under a CSMA/CA access method, the throughput optimality of the algorithm would not be achieved. There is in fact a throughput gap when comparing performance of backpressure algorithm with the theoretical TDMA system compared to the adapted
version of the IEEE802.11 with CSMA/CA access method. The work of [103] analyses the extent of this performance gap and highlights that the main reason for it is control inaccuracy resulted by approximations made in link scheduling. MAC layer collisions and senders back-off results packet loss which is one of the main sources of the throughput gap which is also resulted by inaccuracy of control signalling. Most of the researches involving backpressure routing algorithms make unrealistic assumptions to realize the TDMA requirement of the algorithm which makes their work infeasible to implement in reality. The assumption in this work is that the MAC layer protocol is IEEE802.11. Hence all the limitations, simulations and analysis is based on a CSMA structure.

### 4.5.1 Analysis of Throughput Performance, Theory vs. Reality

It was discussed that there is a throughput performance gap associated with access mechanism when comparing backpressure algorithm joined with TDMA and CSMA systems. In this section, we intend to briefly analyse this performance gap which leads us to the weaknesses associated with the IEEE802.11 as the MAC layer of interest in this research.

As it was covered in Section 2.11.2, backpressure algorithm requires every node to have the information about the per-flow queue lengths of their neighbouring nodes to be able to estimate the per-flow backpressure of that link connecting that node to that neighbour. Hence all the nodes in the network must be regularly updated with this queue information to be able to fully utilize the backpressure algorithm. Now the main problem arises from the fact that, in a CSMA environment, it is physically impossible for the nodes in the network to have instant access to the information relating to the queues of the neighbouring nodes. The practical approach in distribution of these queue information is to perform a cross-layering between the network and MAC layer and distribute this information using the routing protocol. As it was discussed in the literature study, there are three types of routing protocols, i.e. reactive, proactive and hybrid. The nature of reactive routing is in conflict with the requirement of backpressure routing which needs up to date access to queue information. Hence reactive routing protocols are not suitable for integration with backpressure routing. On the other hand, proactive routing protocols can cooperate with the backpressure routing in the sense that they can proactively distribute the queue information and provide an up to date database to the backpressure algorithm. However, even with a proactive routing protocol there is a limit to the number of queue related updates that nodes can distribute. As a result, the queue related information in the nodes can possibly be out of date and affect the performance of the backpressure routing. The limitation in the rate of signalling updates that nodes distribute in the network is simply because under a CSMA environment the channels are shared amongst contending nodes and the signalling updates are performed via broadcasting mechanisms implemented by the routing protocol. More signalling updates translates to more of the channel’s capacity being utilized for non-user-data related transmissions, which has a negative impact on the network throughput.
Another type of critical data which has a negative impact on the optimal throughput performance of backpressure algorithm is the topology information. One of the assumptions made in the original backpressure routing was knowledge of full network topology graph by the centrally managed algorithm. Up to date access to the network topology graph requires an efficient signalling mechanism. Signalling is costly and consumes network resources such as the network throughput capacity. In IEEE802.11 due to collision and back-off mechanism, network capacity is very limited and valuable. As a result, there is a limitation in the number of topology updates that nodes can broadcast in the network. IEEE802.11 utilizes DCF (Distributed Coordination Function) to manage the medium access. Prior to every transmission, a node requires to sense the channel for a specified duration so called DIFS (DCF Inter-frame Space). If the channel is found to be idle during the DIFS sensing interval, then the transmission is permitted. Otherwise, if another node is sensed during this interval or a collision takes place after initializing the transmission, then the transmission is deferred based on a back-off time. As IEEE802.11 relies on the CSMA-CA probabilistic channel access model, hence creation of synchronised time slotted system required by backpressure is very challenging; this is simply because CSMA cannot guarantee a time slot for transmission which is free of any collisions and back-off times. There are delays associated with default operation of IEEE802.11 such as back-off time, inter-frame spacing and etc. which affects synchronization of a TDMA system designed based on the CSMA IEEE802.11.

4.6 Separation of Routing from Scheduling in Backpressure Algorithm

Generally routing is performed by the network layer and scheduling is managed by the MAC layer in the OSI model. Backpressure routing brought a modification to the OSI model which as a result the network, MAC and PHY layers were all joined together and centrally managed by the backpressure algorithm. The routing performed by the original idea of backpressure is completely different to the conventional definition of routing in the literature. In multi-hop networks, a route dictates the path which the data packets would traverse for reaching the destination. On the other hand, backpressure algorithm performs the routing task via hop by hop relaying of the data based on queue differential backlog until eventually the packet reaches the intended destination. Backpressure algorithm mainly relies on the scheduling side than the conventional end-to-end routing paradigm. In order to integrate backpressure algorithm with the spectrum-aware OLSR algorithm that was introduced in Chapter 3, it is essential to separate routing from scheduling. In the current implementation of our spectrum-aware OLSR, a distributed spectrum-aware signalling is performed by the nodes in the network. The signalling mechanism provides the information required by every node in the network to compute the best weight path to every destination in the network. As it was discussed in Chapter 3, our spectrum-aware routing algorithm utilizes SOPs in computation of the routing tables. Reception of every signalling message triggers computation of the end-to-end routes by the routing algorithm. However, only the next hop node
which is required to reach a destination node is recorded in the routing table. The reason that only the next hop node is recorded rather than the complete route is that our algorithm is designed for the dynamic environment of CR-MANETs. In such environment, the frequency of changes in the network topology and channel qualities are considerably high and the best performance is only achieved if every hop along the way from source to destination re-evaluate the best weight route for reaching that specific destination; this mechanism assures that out-of-date signalling information would not affect the quality of computed routes. In computer networks, source routing is a method when source nodes insert the full/partial end-to-end routes in the packets to use for routing purposes. Based on our discussion above, source routing is not suitable for the dynamic environment of CR-MANETs. An example of spectrum-aware routing table is shown in Figure 4-1. In our spectrum-aware environment, routing tables accommodate 4 entries for every destination node in the network, i.e. Next_hop_addr, Interface_addr, Weight and Hop_count. Hop by hop routing of data packets are performed by 2 critical addresses. 1) Next_hop_addr, which is the IP address of the next hop for relaying the packet. 2) Interface_addr, which is the MAC address of an interface in the current node which is tuned to transmit on a certain channel that the spectrum-aware routing algorithm has identified to be part of the minimum weight path to the destination node. The Interface_addr is an additional entry which its existence in the routing table is necessitated because of the CR idea. It is worth noting that the spectrum-aware routing presented in Chapter 3 is categorized as a single-path routing protocol, meaning that for every destination in the network, one and only one best weighted route is recorded in the routing table by our algorithm. In the Section 4.7 it will be explained that a multi-path routing is required to integrate the backpressure routing into our implementation.

<table>
<thead>
<tr>
<th>Destination_addr</th>
<th>Next_hop_addr</th>
<th>Interface_addr</th>
<th>Weight</th>
<th>Hop_count</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.0.15</td>
<td>192.168.0.13</td>
<td>00:09:5B:16:5D:40</td>
<td>50</td>
<td>10</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Figure 4-1: Sample of Routing Table in Spectrum-aware OLSR

It can be concluded from the above discussion that the routes in the routing table dictate the next hop relay for a packet in its journey from source to destination node. As a data routing request is generated, a search is initiated in the routing table to find a match. Next, the data is sent to the MAC layer queue for the intended interface and the IEEE802.11 performs the required sensing procedure explained in Section 4.5.1 to activate a link for transmission. This process is repeated on a hop by hop basis until the packet is received by the destination node. The link activation process explained in here is what in the work of backpressure is referred to, as link scheduling. The link scheduling in backpressure algorithm is designed in such way that the differential backlog inequality presented in Section 2.11.2 is satisfied. However, as it was covered above, backpressure performs the scheduling and routing in a joint manner.
Hence the main challenge in this work is to initially adapt the spectrum-aware routing mechanism with the queue gradient of backpressure and then adapt the routing algorithm with the currently deployed CSMA structure of IEEE802.11 to achieve a better and more stable throughput. The next section covers the essential step in integration of backpressure which is a deviation from single-path to multi-path routing.

4.7 Deviation from Single-path to Multi-Path Routing

The main difference between a multi-path and single-path routing is the number of routes registered in the routing table per destination nodes in the network. Multi-path routing records back up routes for the best route in the network while single-path routing does not. A multi-path enhancement to a reactive routing protocol comes at the cost of extra routing overhead in the network but this does not apply to a proactive routing protocol. This is due to the fact that in a proactive routing protocol the full network topology graph is advertised to the other nodes in the network and as a result there is enough knowledge about the network topology to compute all the paths that leads to every destination. Hence, by changing the structure of our proactive spectrum-aware OLSR routing algorithm from single-path to multi-path routing structure we would not expect to have an added signalling load in the network. The work presented in [72] argues that unless the number of alternative routes created by a multi-path routing consists of a very large set, the load distribution is almost the same as a single path routing structure. By the simulation study provided in this work we will show that in ad hoc networks, multi-path routing can in fact result better load balancing which leads to routes with better QoS factors in the network.

The process of changing the structure of spectrum-aware OLSR routing protocol from single-path to multi-path routing consists of the following steps.

i. Removing any restricting policy currently deployed in OLSR for the maximum number of topology tuples that can be stored for every destination node in the network. A topology tuple in OLSR has the format of (T_dest_addr, T_last_addr, T_seq, T_time). Topology tuples are extracted from TC messages which are periodically flooded in the network. The result of this modification is that, every destination node in the network can be accessed from unlimited number of last nodes. Registering all the topology tuples is essential in a multi-path routing protocol.

ii. Disabling the MPR signalling optimization mechanism. During the simulation test runs we identified that utilization of MPRs creates routing loops in OLSR which affects the performance of the routing protocol under a multi-path scenario. Disabling MPRs means that the TC messages are now flooded in the network and do not benefit from any optimization mechanism. The assumption in our work is that, in every node in the network, channel 1 in
the IEEE802.11 is dedicated to signalling purposes. We have not modelled any inter-channel interference in this work. Hence the added signalling load resulted by the lack of MPR optimization does not affect the capacity of other available SOPs. Another benefit of a dedicated signalling channel is that, the control channel is safely unaffected even under the circumstances when the network is heavily loaded.

iii. Changing the structure of the routing table in Spectrum-aware OLSR to accommodate the additional alternative paths for every destination. The current structure of the routing table in spectrum-aware OLSR only allows a single route to every destination in the network and addition of any extra route would result removal of the currently stored route to the intended destination. Hence, we have modified the structure of the routing table to allow up to $\mu$ alternative routes for every destination in the network. We have varied the value $\mu$ in the simulation study of this chapter (Section 4.11) to show its effect on the performance of the protocol, which is discussed later in the thesis. As a rule of thumb, a low number of alternative paths in a multi-path routing results ineffective load balancing in the network; on the other hand, a large set would deviate the performance of the algorithm from best weight paths. Hence an optimum value of $\mu$ is critical for the performance of the routing algorithm.

iv. The most fundamental change made to the routing protocol in order to add the multi-path support is in the route extraction and computation algorithm. The flowchart of the multi-path routing algorithm introduced by this work is shown in Figure 4.3. The modification is performed based on the routing algorithm presented in Section 3.9 in order to add the multi-path support. This algorithm can either be triggered by receiving any of the OLSR’s HELLO/TC/MID messages or after any changes in the underlying dynamic network SOP information which leads to re-computation of routing tables. There are a few stages that are required in route extraction in multi-path routing. In the algorithm presented in Figure 4.3, initially the information supplied by Hello messages are used to identify the 1-hop neighbours and to find the alternative links to each neighbour. Hence this information creates the first hop in the possible route to any destination. Next, in the first column of the flowchart, the links to every individual neighbour are sorted and listed according their cost metric. This is followed by the second column where the topology information extracted from TC messages are extracted and listed in the right order. As it was mentioned before, TC messages have the responsibility of updating the network topology graph throughout the network. In the second column of the flowchart we make sure that the topology information is grouped together based on destination-node, last-node pairs. The intention behind this is to find the alternative links (aka SOPs) between every node pairs. Next important step is to list the links in order of the link cost. The main functionality of the algorithm shown in Figure 4.3 is
Chapter 4: Backpressure Routing

summarized in column 3 and 4. At column 3 and beginning of column 4, the extracted data structures created in column 1 and 2 are used to create the multi-path routing table via an iterative mechanism. Our approach in choosing the alternative paths in multi-path routing is based on avoidance of conflict paths. Each link can only be used in one of the alternative routes listed in the routing table to avoid any path conflicts. Let’s take the topology shown in Figure 4-2 into account; we assume that node A in this topology is the current node which is running the algorithm shown in Figure 4-3 and the current iteration of the algorithm is computing the alternative routes to destination node F. Now if one of the computed routes to reach destination F is A-B-H-D-E-F and on the second iteration the route A-B-H-I-J-F is added. The link B-H is considered as a path conflict, due to the fact that it is used in two of the alternative paths in reaching the same destination node F. To avoid any conflict in the chosen path alternatives in the network, the second path needs to be based on the conflict-free route, A-G-H-I-J-F.

![Figure 4-2: Sample Topology for Multi-Path Scenario](image)

The algorithm shown in Figure 4-3 completely avoids conflict paths. The main idea behind having alternative paths based on multi-path routing is to use the alternative paths in performing load balancing based on backpressure gradients. This balances the traffic flows over various paths which results balanced load conditions in the network. The reason that conflict-free paths are extremely important in our multi-path routing is that the alternative paths are considered to be resources which aid in the load balancing and having links that are shared amongst multiple paths can result overutilization of those links which leads to congestion and renders the network unstable.

In conclusion, the main motivation is changing the structure of the routing protocol from single-path to a multi-path routing structure is to be able to utilize the idea of queue gradients introduced by backpressure routing into our algorithm. A source node decides to choose one of the multiple alternative paths that maximizes queue gradient based on the idea of backpressure that was explained in Section 2.11.2. This results distribution of load over the alternative paths that have the best quality and as queue gradients are taken into account upon every transmission, theoretically the load would be balanced amongst all the alternative paths.
Chapter 4: Backpressure Routing

Create and Initialize 4 objects:
1. Curr_nb = NULL
2. Dest_add = NULL
3. Last_add = NULL
4. Temp_topol = NULL

Create a new object called
Graph_edge

Create a vector of
Graph_edge called
Graph_edge_vec

Create a vector of
Graph_edges named
Global_graph_vect

Transform Best_lnk_vec
associated with each neighbour
to Graph_edge_vec
associated with each neighbour

Pick and remove one element
from the top of the
Graph_graph_ext_vec and store it
in Temp_graph_edge

Add the
Temp_graph_edge to
to the
Global_graph_ext

Run Dijkstra’s shortest
weight Algorithm on the
Global_graph_vect

Record the routes in the
routing table on a per
destination basis

Clear the
Global_graph_ext

Mark all neighbours of the
current node as unprocessed
Mark all node pairs in
topology graph as
unprocessed

Pick a
(Dest_node, Last_node) pair
and load the topology vector
associated with this pair from
the vectors of
Graph_topol_edge_ext_vec

Pick and remove one element
from the top of the
Graph_topol_edge_ext_vec and store it
in
Temp_topol_graph_edge

Create and initialize 4 objects:
1. Curr_nb = NULL
2. Cur_top_tuple = NULL
3. Best_nb = NULL
4. Best_lnk = NULL

Create a vector of topology
tuples associated with
Dest_node and Last_node
named Best_topol_vec

Copy the found topology
to Temp_topol

Search Topology_Set for
tuples that connect nodes
with the currently picked 1-hop neighbour

For each topol_tuple in the
Best_topol_vec, add the
Cost value in the
vector
Best_topol_vec

Create a vector of topology
tuples associated with
Dest_node and Last_node
named Best_topol_vec
(which groups all the tuples or
pairs)

Mark all node pairs in
topology graph as
unprocessed

Add the
Temp_graph_edge to
to the
Global_graph_ext

Mark all neighbours of the
current node as unprocessed
Mark all node pairs in
topology graph as
unprocessed

Start

TC/Hello/MID
Messages Received

Erase Routing Table

Create and Initialize 4 objects:
1. Curr_nb = NULL
2. Curr_lnk = NULL
3. Best_nb = NULL
4. Best_lnk = NULL

Create an object “Curr_nb”
Initialize it with the currently
picked 1-hop neighbour

Pick one neighbour
from 1-hop neighbours
Create an object “Curr_lnk”
Initialize it with the currently
picked 1-hop neighbour

Add Curr_lnk to the
list of current topologies in
the Topology_Set

Is there any more
topologies in the
Topology_Set

Create a vector of
Graph_edge

Create a vector of
Graph_edges named
Global_graph_vect

Transform Best_lnk_vec
associated with each neighbour
to Graph_edge_vec
associated with each neighbour

Pick one neighbour and load
the Graph_edge_vec for that
neighbour

Pick and remove one element
from the top of the
Graph_edge_vec and store it
in Temp_graph_edge

Add the
Temp_graph_edge to
to the
Global_graph_ext

Mark all neighbours of the
current node as unprocessed
Mark all node pairs in
topology graph as
unprocessed

End

Figure 4-3: Multi-path Route Extraction and Computation Algorithm
4.8 Integration of Queue Information in Signalling

As it was detailed in Section 2.11, backpressure routing requires nodes in the network to have updated queue information about their neighbouring nodes. It is only with this information that nodes can compute their per destination queue differentials and apply the backpressure algorithm in a distributed fashion. In order to supply this information to the nodes in the network some modifications have been made to the spectrum-aware signalling mechanism implemented in our routing algorithm. As backpressure algorithm requires the neighbouring nodes queue information then the modification needs to be applied to the OLSR’s HELLO messages which are periodically distributed to the 1-hop neighbourhood of the nodes. Hence, each node that broadcasts the HELLO messages needs to list the per destination queue sizes of all interfaces. The interfaces are identified by their MAC address and the queue sizes are measured in bytes. The modified OLSR HELLO message is shown in Figure 4-4 which accommodates the per destination queue information required by the backpressure routing.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Reserved | Htime | Williness
---|---|---
Dest_MAC_Address_1
Queue_Size_1
Dest_MAC_Address_2
Queue_Size_2
...
Link Code | Reserved | Link Message Size
---|---|---
Neighbour Interface Address_1
Neighbour Interface Address_2
...

Figure 4-4: OLSR’s HELLO Message with the Queue Information

As HELLO message Emission Interval (HEI) dictates the periodic distribution of HELLO messages (set to 2s in OLSR by standard), depending on the dynamic network environment, queue information can end up being out of date. The frequency of queue updates to the 1-hop neighbourhood can be tuned with the HEI parameter. However, there is a limit to how small the HEI can be or in the other words how fast is the emission frequency of the HELLO messages. The work of [104] has analysed the performance of OLSR under various TC/HELLO message emission intervals. But it can be seen that even in this work there is a limit to the frequency of emissions. This restriction is due to the fact that broadcasting HELLO messages consumes the network capacity and has a negative impact on the overall QoS performance of the network.
Chapter 4: Backpressure Routing

Hence with the modifications made to the OLSR message structure as it was shown in Figure 4-4, 1-hop neighbours will be updated with the per destination queue information which is used in this work to calculate queue differentials.

4.9 Backpressure Full Integration

As it was explained in Section 4.5, backpressure routing is designed with the assumption of a TDMA MAC layer protocol and its implementation under a CSMA system is a challenging area of research. In Section 4.7 we have covered the necessary modifications that needs to be made to the original spectrum-aware routing algorithm that was introduced in Section 3.9 to provide the multi-path routing support. Now, the multi-path Spectrum-aware OLSR is capable of storing $\mu$ number of alternative paths for each destination in the network. Based on the algorithm shown in Figure 4-3, rather than storing the entire routing paths, every source node stores the best cost $\mu$ number of next hops which leads it to all destinations in the network. The multipath algorithm introduced in this work is mathematically supported and based on the generalization of Dijkstra’s algorithm to multi-layered graphs (multigraphs) introduced in [95]. Hence the algorithm functions in a distributed fashion in the network without any central control mechanism and can find the best cost routing paths.

The next step towards implementation of the backpressure algorithm is to change the structure of the queues in the MAC layer. As it was covered in Section 2.11, backpressure algorithm requires distinct per destination queues to apply the backpressure queue gradients on a per flow basis. It was covered in Section 4.8 that the per destination queue information can be integrated in OLSR signalling mechanism
to enable backpressure algorithm. One of the missing puzzles from full integration of backpressure routing is to enable per destination multi queue support at the MAC level. Towards this aim, we have modified the queue structure of the IEEE802.11 MAC from a single-queue to a multi-queue structure.

By referring to Figure 4-5, when packets are generated in the application layer of the OSI model, they are tagged with the address of a destination node in the network which then the network layer is responsible to provide an appropriate next hop for routing the packet to the intended destination. When the next hop is identified, then packets are queued at the MAC sublayer under the datalink layer to be forwarded to the next hop node. MAC layer supports a rate at which packets can be transmitted and this rate is dependent on many factors which is not the area of concentration in this thesis. As it was detailed in Section 2.11, if the rate at which the application layers generates the packets and the network layer forwards them down to the MAC layer is greater than the rate at which they can be transmitted to the next hop in the network, then packets need to be queued and served one at a time when the MAC layer can access the communication medium again. Conventionally there is just one queue for the packets in the MAC layer, regardless of their destination. The backpressure algorithm requires per destination queue capability. Additionally, in a CR environment and under a spectrum-aware scenario, one of the assumptions made in this work (in Chapter 3) is that nodes have multi-interface capability and each interface is tuned to a specific channel. Ideally, under and orthogonal channel model, a data transmission made on one channel would not cause any interference on other interfaces. In the spectrum-aware OLSR algorithm implemented in this work, the local interface to reach that next hop node is essential in the hop by hop relaying of the data in the network. In order to enable the multi-queue structure required by the backpressure routing in our spectrum-aware OLSR algorithm we have introduced per destination queues on a per interface basis. Since per destination queues on one interface which is tuned to a specific channel would not have an effect on the queues in another interface, this is in line with the original idea of backpressure.

Based on our proposed multi-queue multi-interface design, each and every individual interface in a node have multiple per destination queues. Hence when a packet is sent down to the MAC layer via the network layer, based on the next hop node, an appropriate interface (tuned to the right channel) will be chosen to forward the data packet. If the chosen interface is busy or there is a queue for that specific destination on that interface then the data packet is directed to the appropriate queue and is served on a “First In, First Out” (FIFO) basis.

Now the main question here is the mechanism used in the network layer to choose one of the alternative paths for forwarding the data to the intended destination. As it was explained in Section 4.7, the main intention behind deviating from a single-path to a multi-path routing structure was to factor in the backpressure queue gradients in the decision-making processes of the routing protocol. Let’s consider the multi-channel cognitive network shown in Figure 4-6 under the case that a packet is
generated at Node C and needs to be forwarded to destination Node H. Table 4-1 summarizes a filtered version of the routing table data that is stored in Node C for destination Node H. In this example, the $\mu$ parameter is set to 4 which means that the number of alternative routes registered in the multi-path spectrum-aware OLSR is limited to 4.

![Figure 4-6: Example of a Multi-Channel Network](image)

According to the data shown in Table 4-1, there are 4 alternative next hops registered for the destination node H. According to the algorithm that was presented in Figure 4-3, routes are listed in order of best (low cost) to worst (high cost). Queue differentials are calculated based on the local data and the queue information provided by the signalling mechanism that was covered in Section 4.8. Nodes have local access to the information on their destination based queues, hence upon reception of HELLO messages from the neighbouring nodes, the queue differentials can be calculated according to Eq. 4.1. Where in this equation, the queue differential $D$ towards destination node $d$ relative to next hop node $n$ is calculated by the queue length differential of current node $c$ and next hop node $n$ towards destination node $d$ on interface $i$ at time $t$.

$$D_{(d,n)}^{(c,i)}(t) = Q_d^{(c,i)}(t) - Q_d^{(n,i)}(t)$$  \hspace{1cm} Eq. 4.1

The main objective in our implementation of backpressure spectrum-aware OLSR is to maximize the queue differential equation provided in Eq. 4.1 amongst the alternative paths provided by the multipath routing that was covered in Section 4.7. Now if we look at the routing table provided in Table 4-1, based on our algorithm the next hop node E which utilizes the local interface Int_2 (tuned to channel 3) is the best option to reach the destination node H because it maximizes the queue differential towards that destination. As this algorithm’s optimization mechanism takes place on a packet level, as soon as the queue differentials on some links towards a certain destination increases, the rate at which those queues are served would increase as well. As our implementation of backpressure algorithm takes advantage of the SOPs, hence it results network load balancing over the channels towards the intended
destinations. The main advantage of our algorithm is that it does not require a TDMA MAC protocol and is fully functional with the CSMA-CA structure of the IEEE802.11. It must be noted that the implementation of backpressure in this work is not globally throughput optimal but rather throughput optimal over the alternative paths resulted by multi-path routing. The downside to utilizing a CSMA MAC protocol is that unlike a TDMA system, a comprehensive mathematical analysis of our algorithm is extremely complex.

Table 4-1: Example of Routing Table

<table>
<thead>
<tr>
<th>Destination</th>
<th>Next Hop</th>
<th>Local Interface</th>
<th>Queue differential</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>H</td>
<td>F</td>
<td>Int_6 (Ch11)</td>
<td>64</td>
<td>3</td>
</tr>
<tr>
<td>H</td>
<td>E</td>
<td>Int_2 (Ch3)</td>
<td>128</td>
<td>5</td>
</tr>
<tr>
<td>H</td>
<td>E</td>
<td>Int_4 (Ch9)</td>
<td>-32</td>
<td>11</td>
</tr>
<tr>
<td>H</td>
<td>F</td>
<td>Int_1 (Ch1)</td>
<td>24</td>
<td>15</td>
</tr>
</tbody>
</table>

In the next section, we have compared the proposed algorithm with the state of the art protocols discussed in Section 4.2.

4.10 Design comparison of Backpressure Spectrum-aware OLSR with State of the Art Protocols

In Section 4.2 we discussed the two of the state of the art works that are based on the concept of backpressure routing in CR-MANETs. Although neither of these works are considered to be spectrum-aware by our definition, but they are the most up to date contributions made in the literature that stand as the basis of our comparisons in this section.

As it was pointed out in Section 4.2, the work in [97] mandates a full network knowledge to be centrally available to perform the routing and scheduling based on backpressure algorithm. Seeing this as an infeasible assumption, our algorithm follows a proactive design which means that all the network topology is acquired prior to any route computation. Furthermore, as it was pointed out in Section 4.7, our algorithm performs the route computation on a distributed basis without the need for a centrally managed routing entity. This is in line with the dynamic nature of CR-MANETs. The work in [97] models routing and scheduling on separate scopes where the routing algorithm does not incorporate the backpressure queue gradients directly as a metric into the route decision making processes. We
discussed the problem with this approach in Section 4.2, highlighting the fact that this results routing loops to be created and majorly impacts the performance of larger networks. In order to address this, we have proposed a multi-path approach to routing. In our multi-path route computation and storage, the backpressure queue gradient is only applied to a subset $\mu$ of low cost source-destination routes. Hence, this would result a performance trade-off between load balancing and QoS support. Our algorithm has proven to be loop-free, as route computations applies backpressure to only a subset $\mu$ of low cost end-to-end paths on a hierarchical basis as explained in Section 4.7. In the other words, low cost paths based on SOPs and ETX metric proposed in Chapter 3 is prioritized over backpressure algorithm. As backpressure was originally designed for fixed networks based on a TDMA channel structure, it is known to cause looping effects in a network with mobility and based on a CSMA MAC design. The novelty of hierarchical structure of our algorithm is that it avoids such looping effects. Furthermore, the work in [97] assumes nodes with single interfaces which can perform cognitive channel switching. We have argued in Section 2.5.1 that a single interface design is subject to deafness problem [41]. Our algorithm is based on a multi-channel, multi-interface design which overcomes this problem. Furthermore, our algorithm design does not depend on the network size and capacity, unlike the optimization dependency of the work in [97]. The fine-tuning contention window parameters introduced by this work requires continuous optimization to support various network sizes and topology models. The performance gain of applying backpressure algorithm to spectrum-aware OLSR is well justified in the next section of our work however this gain seems to be very negligible in [97] as it was discussed in Section 4.2. Similar to the work presented in [97], the work in [98] follows a dynamic design of contention window parameter which adds negligible performance gain but excessive complexity to the system. Per our discussion in Section 4.2, this work applies a limit to the per node queue size in order to fix per-link transmission probability. It is claimed that this would satisfy the throughput optimality in the mathematical modelling of backpressure algorithm, however the simulation results presented suggest that the algorithm has zero tolerance to high network load condition which is contradictory to the proven throughput optimality of backpressure algorithm. We have addressed this problem by the discussed, multi-path (and multi-metric) algorithm design which balances route stability by taking into account route quality and backpressure queue gradients on a hierarchical basis. One of the main strength of our algorithm is the spectrum-aware route computation which dynamically incorporates SOPs into the end-to-end paths, however the work in [98] has not considered this aspect of CR networks. One of the major contradictions that has been addressed by the contribution made in this chapter is incorporation of backpressure queue gradients into our spectrum-aware utility function, which was the subject of Sections 4.3, 4.4, 4.5.1 and 4.7. To the best of our knowledge, this contradictory routing strategy has not be addressed in either of the works presented in [97] and [98].
4.11 Simulation Study on the Backpressure Spectrum-aware OLSR

In this section, we have summarized the simulation study on our proposed Backpressure Spectrum-aware OLSR algorithm and provided a thorough discussion on these results.

4.11.1 Simulation Setup

The simulation setup is similar to the ones summarised in Chapter 3 in order to make the results comparable and to be able to highlight the relative gains compared to the Spectrum-aware OLSR that was proposed in Chapter 3.

It is very important to consider that for all the statistical averaging and error bar analysis performed on the simulation results presented in this chapter a confidence level of 95% has been used.

The simulated network comprises of 12 active nodes (6 source and 6 destination nodes). Therein addition, there are 15 relay nodes which are used to provide the alternative routing paths between the source/destination pairs. All nodes are distributed randomly over the simulation area to simulate a realistic ad hoc scenario. In our simulation scenario, an active host can relay its application layer traffic through other relay/active hosts. The simulations are averaged over 15 runs to increase their confidence level. Additionally, in order to overcome the transient instability in the network, the initial 700s of each simulation run is omitted from the averaging performed over the final results. The performance of the backpressure spectrum-aware OLSR routing algorithm has been analysed under three scenarios, involving variations of speed, network load and relay node density and analysing the impact on End-to-End Delay, PDR and Routing overhead. It is worth noting that ETX is used as the cost metric of choice in our algorithm.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulation Time</td>
<td>2000s</td>
</tr>
<tr>
<td>Transient Interval</td>
<td>700s</td>
</tr>
<tr>
<td>Number of repeats</td>
<td>50</td>
</tr>
<tr>
<td>Simulation Area</td>
<td>2000m*2000m</td>
</tr>
<tr>
<td>Active Hosts (Default)</td>
<td>12</td>
</tr>
<tr>
<td>Relay Hosts (Default)</td>
<td>20</td>
</tr>
<tr>
<td>Mobility Model</td>
<td>Random Waypoint</td>
</tr>
<tr>
<td>Mobility Initial X and Y positions</td>
<td>Random</td>
</tr>
<tr>
<td>Mobility Speed (Default)</td>
<td>5m/s</td>
</tr>
<tr>
<td>Mobility Wait Time</td>
<td>Random [ 3s , 5s ]</td>
</tr>
<tr>
<td>Number of Flows</td>
<td>6</td>
</tr>
<tr>
<td>UDP Application</td>
<td>CBR</td>
</tr>
</tbody>
</table>
4.11.2 Results and Discussion

In this section, the simulation results of the backpressure spectrum-aware OLSR (OLSR-BSA) routing algorithm is compared with the baseline OLSR (OLSR-ETX) and spectrum-aware OLSR (OLSR-SA) routing algorithms. The simulation study is concentrated on the effect of changing the $\mu$ parameter (covered in Section 4.7) on the performance of OLSR-BSA under variations of Speed, Node Density and Network Load. The simulation environment and parameters used for the study in this section has remained the same as the simulations studies provided in Section 3.5 and 3.10 of Chapter 3 in order to keep the results comparable.

4.11.2.1 Speed Variation

As it was covered before, the performance of OLSR-BSA is compared against OLSR-SA and OLSR-ETX under variations of speed. The mobility model used for these simulations is the Random Waypoint model with the wait time parameter randomly chosen (with a uniform distribution) from the interval of $[3s, 5s]$. The speed has been varied from 0 m/s (stationary) to 35 m/s (max.) which is the conventional range of mobility analysis on many researches involving MANTEs.

Based on the end-to-end delay graph shown in Figure 4-7, the overall trend of OLSR-BSA follows the same behaviour as OLSR-SA and OLSR-ETX. This is due to the fact that when nodes are stationary,
there is less opportunity for transmission compared to when they are moving at a relatively low speed of 5 m/s. This was also observed in the benchmarking results provided in Chapter 3, which shows that the backpressure algorithm has not affected this behaviour. Under a stationary scenario, on an average basis, active hosts that end up in a sparse network topology have major impact on the overall end-to-end delay of the protocol. On the other hand, all trend lines indicate that at speeds higher than 5 m/s the end-to-end delay increases and that is due to the failure of routing protocol in providing up to date information in line with the frequency of changes in the network topology.

![End-to-End Delay](image)

*Figure 4-7: End-to-End Delay (s) vs Speed (m/s) OLSR-BSA, OLSR-SA and OLSR-ETX*

Furthermore, one of the most important observations in Figure 4-7 is that as the $\mu$ parameter has been increased from 5 to 20, it has resulted higher end-to-end delay which affects the average QoS performance of the routes. Higher $\mu$ parameter translates to higher number of alternative paths. The more alternative next hop options that are available to our backpressure algorithm, the higher the chance of choosing a path that deviates from the best cost path. The fact that the end-to-end performance of OLSR-BSA decreases with increasing $\mu$, shows that the paths chosen by our proposed backpressure algorithm is deviating away from the best cost paths chosen by OLSR-SA and OLSR-ETX. Now the main question is whether this deviation from best cost path is beneficial for the overall performance of OLSR-BSA. To be able to answer this question we need to look at the PDR graph shown in Figure 4-8.

According to Figure 4-8, it can be seen that OLSR-BSA follows the same trend in terms of PDR as OLSR-SA and OLSR-ETX. All protocols indicate that the PDR is higher when there is a minor mobility compared to a stationary case, which is in line with our reasoning for the End-to-End delay graph shown in Figure 4-7. On the other hand, PDR decreases for all protocols at speeds higher than 5 m/s which shows an overall instability of routes under highly dynamic scenarios. Given the proactive structure of OLSR, it is perfectly normal to have higher packet loss at high mobility where the probability of errored routes in the routing tables is higher. It can be clearly observed that OLSR-BSA with $\mu = 5$ has a
performance gain compared to the baseline OLSR-ETX and OLSR-SA, which is resulted by the backpressure algorithm. Furthermore, OLSR-BSA with $\mu = 10$ has resulted the highest PDR gain compared to all other cases. One of the most important observations here that should be highlighted is that as $\mu$ parameter is increased above value 10, the PDR performance degrades and falls below that of OLSR-BSA $\mu = 5$ and even OLSR-SA. This shows a trade-off between choosing best cost paths and satisfying the queue gradients dictated by backpressure routing.

![Packet Delivery Ratio vs Speed (m/s) OLSR-BSA, OLSR-SA and OLSR-ETX](chart)

**Figure 4-8: Packet Delivery Ratio vs Speed (m/s) OLSR-BSA, OLSR-SA and OLSR-ETX**

![Normalized Routing Overhead vs Speed (m/s) OLSR-BSA, OLSR-SA and OLSR-ETX](chart)

**Figure 4-9: Normalized Routing Overhead vs Speed (m/s) OLSR-BSA, OLSR-SA and OLSR-ETX**

The higher alternative paths create more alternative options to be chosen based on the queue gradients of the backpressure algorithm. Routes that satisfy the backpressure gradients may possibly be in contradiction with the best cost routes chosen by OLSR-ETX and OLSR-SA, which is the reason why
the number of alternative best cost routes provided to the backpressure algorithm, should be limited so that the excessive deviation from the best cost routes does not result in performance degradation. The performance degradation resulted by this phenomenon can be clearly seen at $\mu$ parameters above 10.

The graph shown in Figure 4-9 compares the normalized routing overhead of the proposed routing algorithm with the baseline OLSR-ETX and OLSR-SA. The routing overhead in this graph is normalized over the number of successfully delivered packets. Normalization over the successful delivery of packets helps in analysing the effectiveness (in terms of PDR) of the extra signalling under dynamic scenarios. It can be observed that the routing overhead generated by OLSR-BSA is significantly higher than OLSR-SA and OLSR-ETX. As it was detailed in Section 4.8, due to the requirements of backpressure algorithm, queue information needs to be integrated in OLSR’s signalling mechanism which results higher routing overhead compared to the baseline protocol OLSR-ETX and OLSR-SA. On an overall basis, normalized routing overhead increases as speed increases which is firstly due to the extra signalling triggers generated in dynamic scenarios compared to stationary ones and secondly due to the fact that at higher speeds less data is delivered and the normalization results a larger signalling to successful packet delivery ratio. As it can be observed in Figure 4-9, OLSR-BSA, $\mu = 10$ has lower signalling overhead compared to the cases where $\mu = 5$, 15 or 20. This result is mainly due to the normalization process, as $\mu = 10$ case has the highest PDR across all speeds, hence we expect a lower normalized routing overhead compared to other values of $\mu$, in OLSR-BSA. The worst performance in terms of the routing overhead is observed in the case of OLSR-BSA, $\mu = 15$ and 20 which is justifiable given the low PDR in these cases. As it can be observed from the results provided in Figure 4-9, the routing overhead generated by the OLSR-BSA is significantly high; but the reason that these signalling packets do not affect the performance of our proposed protocol in terms of PDR is that we have allocated an individual channel to signalling purposes in order to avoid the excessive interference caused by the signalling messages on the application layer data. This is in line with the concept of SOPs and dynamic channel environment provided by cognitive radio.

4.11.2.2 Node Density Variation

In this section, the performance of the proposed backpressure routing protocol is analysed under variations of the number of relay nodes (aka Relay Node Density, RND). As it was explained, we have 12 active hosts consisting of 6 senders and 6 receivers. The sender nodes route their application layer traffic via the relay nodes. As the number of relay nodes increases, there are more opportunities for routing data packets but also a lot more interference caused due to the higher concentration of nodes in the simulation area.

Based on the graph shown in Figure 4-10, the rate of change in end-to-end delay for both OLSR-ETX and OLSR-SA is sharper than the backpressure algorithm. This is due to the fact that OLSR-ETX
and OLSR-SA do not support a load balancing algorithm to distribute the network load via the extra routing opportunities resulted by the higher relay nodes. As a result, the best routing paths are overly utilized and the packet drop resulted by collision and interference affects the end-to-end delay of the network. On the other hand, OLSR-BSA utilizes backpressure queue gradients to perform load balancing and results better end-to-end delay at higher node densities. When the parameter $\mu$ is set to values above 10, the end-to-end delay is significantly improved at higher RNDs. This is mainly due to the load balancing resulted by backpressure algorithm. The extra relay nodes are utilized by the backpressure algorithm to balance the network load and hence results better end-to-end delay. The reason that the performance gain is concentrated in the higher range of $\mu$ parameter is that in this range backpressure algorithm has more load balancing opportunities to utilize. But there is of course a limit to the gain that can be achieved by increasing the parameter $\mu$.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{end-to-end-delay.png}
\caption{End-to-End Delay (s) vs RND OLSR-BSA, OLSR-SA and OLSR-ETX}
\end{figure}

It can be seen in Figure 4-10 that the differential gain of end-to-end delay when the $\mu$ parameter approaches the value 20 is less than the differential gain at lower values. By increasing the parameter $\mu$ above 20 we would not expect any significant gain and this has been confirmed by simulation studies which due to repetition, it has been omitted from these graphs. Furthermore, the increase in the range of error bars at higher values of RND highlights significant alteration of routing paths between the source/destination pairs.

The graph of PDR against RND is shown in Figure 4-11. On an overall basis, it can be observed that the PDR is most optimal when RND is in the range of 20 to 35. Furthermore, PDR trends downward as RND increases on the higher range of 50 relay nodes; this is due to the excessive interference and collision caused due to the high density of relay nodes. It can be seen that the backpressure algorithm has resulted a gain in PDR when comparing the OLSR-BSA (with $\mu = 5, 10$) with OLSR-SA and OLSR-ETX. This gain is due to the added routing opportunities resulted by higher number of relay nodes. The
performance of OLSR-BSA degrades for values of $\mu$ higher than 10 which is due to the trade-off between the routing paths that satisfy the best cost and those which minimize the queue differentials. Based on our backpressure algorithm, when there are many alternative routing paths available, with a higher $\mu$, the probability of deviating from the best cost path increases. As the algorithm prioritizes the backpressure queue gradient over the best cost $\mu$ number of alternative next hop nodes, due to load balancing, routes can become long and unstable which leads to lower PDR.

The graph of normalized routing overhead against RND is shown in Figure 4-12. It can be observed that on an overall basis OLSR-BSA imposes a higher signalling load on the network compared to OLSR-SA and OLSR-ETX. Given the added signalling load due to the backpressure queue
requirements, this higher signalling load is acceptable. As RND increases, all routing protocols have added signalling load which is an expected behaviour in the graphs. The higher density of relay nodes results more signalling messages to be generated which adds to the overall routing overhead of the network.

It is worth noting that OLSR-BSA, $\mu = 10$ has resulted the best normalized routing overhead amongst all other $\mu$ parameters. This is again due to the high PDR at $\mu = 10$ which considering the normalization which is performed on successful delivery of packet, it results a moderate normalized routing overhead.

### 4.11.2.3 Network Load Variation

In this section, the performance of the proposed spectrum-aware backpressure OLSR is analysed against variations of network load. In our simulation setup, the network load is generated and processed via the 12 active hosts in the network. As it was mentioned before, 6 of these active hosts are sender nodes which generate CBR, UDP traffic at a specified rate to the 6 other destination nodes. The values of load in these sections are in fact the application layer data rates. Hence it must be noted that the load that is imposed by the 6 source nodes must to be multiplied by 6 to reflect the total network load. But given the relative analysis and for the sake of simplicity we refer to the value of loads on a per source basis.

According to the graph of end-to-end delay versus network load shown in Figure 4-13, it can be observed that on an overall basis, end-to-end delay degrades as network load increases. This is due to the fact that a higher network load results higher congestion, MAC layer collision and excessive interference which leads to a poor end-to-end delay performance in the network. At values of load below 10 KB/s, OLSR-ETX and OLSR-SA perform better than the proposed OLSR-BSA but as network load increases to above 10 KB/s, OLSR-BSA outperforms the other two baseline protocols. This can be explained by the fact that OLSR-BSA is capable of load balancing over the available alternative routing paths which is why its performance gain is mainly highlighted at high network loads. It was elaborated in Section 2.11 that based on literature, backpressure algorithm is known to suffer from routing loops and lack of performance at low load situations. This can be well observed in the result shown in Figure 4-13. The reason behind this is that at low load conditions, the queues are short and load balancing can result source/destination pairs to take unnecessary lengthy paths to deliver the data. On the other hand, when the network is under heavy load conditions, the load balancing benefit of the backpressure algorithm outweighs the gain resulted by best cost routes (which is mainly used in OLSR-ETX and OLSR-SA).
The graph shown in Figure 4-14 lists the PDR performance of all routing protocols against variations of network load. One of the main points that can be observed in this graph is that as the network load is increased from 1 KB/s to 10 KB/s, the PDR performance of OLSR-ETX and OLSR-SA remains the same while a gain can be observed in the PDR performance of OLSR-BSA routing protocol.

This gain is due to the fact that OLSR-BSA can achieve better load balancing at relatively high network loads compared to the OLSR-ETX and OLSR-SA. However, the overall trend of all protocols is downward when the network load of 10 KB/s is exceeded. This behaviour is normal for OLSR-ETX and OLSR-SA given that there is no mechanism to balance the excessive load in the network. However, the reason that OLSR-BSA is not capable of maintaining a stable gain at high loads is that, at high loads the rate of queue changes is extremely high. Under such condition, due to the proactive structure of all routing protocols, the rate of signalling updates cannot keep up with the frequency of queue changes at various nodes in the network. This results instability in the queue optimization mechanism in the
backpressure algorithm which results a lower PDR compared to lower load conditions. The gain achieved by OLSR-BSA is again maximized at $\mu = 10$, which is in line with the gains achieved at this value in the simulations involving variations of speed and node density. Another very important observation in the graph of Figure 4-14 is that the PDR performance degrades at values of $\mu$ above 10 and this result is magnified at the higher range of network load. This is again due to the high frequency of queue changes (at high network loads) and the inability of our proactive signalling to keep nodes up to date with those changes. The reason that this result is magnified at higher range of $\mu$ parameter is that the backpressure algorithm utilizes the inaccurate queue information over a large range of next hop options and performs more load balancing based on this information. As the queue data is not accurate under these conditions, the decisions made by the backpressure results extreme instability in the computed routes.

Figure 4-15: Normalized Routing Overhead vs Load (KB/s) OLSR-BSA, OLSR-SA and OLSR-ETX

The results of normalized routing overhead against variations of network load is shown in Figure 4-15. All routing protocols in this graph show a downtrend in the normalized routing overhead against network load. The higher number of successfully delivered packets at higher network load is the reason for this downtrend in the normalized routing overhead. It can be seen that OLSR-BSA has a higher routing overhead when compared to the baseline OLSR-SA and OLSR-ETX which based on the backpressure signalling mechanism, it is well expected. OLSR-BSA, $\mu = 10$ shows the best performance when compared to other $\mu$ values in the graph shown in Figure 4-15. This is due to the high PDR at $\mu = 10$ which results a lower routing overhead in the normalization process. The performance of other values of $\mu$ parameter in OLSR-BSA can be explained using the same concept. It is worth noting again that the excessive routing overhead resulted by the backpressure signalling mechanism does not affect the data transmission due to the dedicated signalling channel which is a concept enabled by the spectrum dynamism of CR-MANETs.
4.11.2.4 Simulation analysis of OLSR-BSA against state of the art protocols

In this section, the simulation results of the backpressure spectrum-aware OLSR (OLSR-BSA) (with $\mu = 10$) is compared against the two state of the art routing algorithms proposed in the work of [97] (referred to as ROSA) and [98] (referred to as DIBAPS). All the tuning parameters suggested by these works have been applied in the simulation study performed in this section, without any modification. As these algorithms perform significantly different to our proposed routing algorithm, we have summarized the simulation results separately in this section in order to be able to concentrate our analysis on the differences between them on a one to one basis. This analysis is performed under variations of Speed, Relay Node Density and Network Load to make the results comparable with the ones given in the previous section. The simulation environment and parameters used for the study in this section has remained the same as the simulation studies provided in introduction of this section (Section 4.11) in order to keep the results comparable.

4.11.2.4.1 Analysis based on variations of Speed, Load and RND

As it can be seen in the results shown in Figure 4-16, Figure 4-18 and Figure 4-20, OLSR-BSA outperforms state of the art protocols (ROSA and DIBAPS) in terms of end-to-end delay. This is mainly resulted by the fact that these algorithms have not considered any QoS metric (except hop count) to optimize computed paths based on delay.

![Figure 4-16: End-to-End Delay (s) vs Speed (m/s), OLSR-BSA, ROSA and DIBAPS](image-url)

The simulation results (in terms of delay) presented in the work of [97] reports delay in magnitudes of seconds which is in line with the high delay reported in our results. The simulation setup considered in the work of [98] is a simplistic scenario which is seen the rapid change of delay in their results. The delay analysis of their algorithm under a realistic simulation scenario indicates high instability in the...
routes computed by the algorithm. Another reason for this delay is the looping effect that was explained in Section 4.2 and 4.10. Furthermore, the error bars for these two algorithms in terms of end-to-end delay suggests instability in the routing paths which was expected based on our analysis in Section 4.2 and 4.10.

The lack of coordination between the joint contradictory strategies in routing that are, lowest cost paths and MAC level scheduling (based on backpressure algorithm) results excessive delay in the computed paths which was previously discussed in Section 4.2 and 4.10. on an overall basis, state of the art routing algorithms show an uptrend against variation of speed and Load and a down trend in terms of RND in the simulation area. The downtrend in terms of RND indicates that the backpressure algorithm utilizes these nodes to distribute the traffic better and that results lower congestion and faster delivery of packets.

Figure 4-17: Packet Delivery Ratio vs Speed (m/s), OLSR-BSA, ROSA and DIBAPS

When we look at the performance of our algorithm against ROSA and DIBAPS in terms of PDR in the results presented in Figure 4-17, Figure 4-19 and Figure 4-21, we can see that our algorithm has outperformed the two. In our simulation study, we noticed excessive looping effect in the two state of the art routing algorithms which results the delivery of packets failing as they reach maximum retry limit in the network layer. We noticed that increasing the maximum retry limit doesn’t improve the performance significantly which justified that both algorithms suffer from contradictory routing strategies discussed in Section 4.2 and 4.10. This contradiction results forwarding the packets based on backpressure queue gradients, which occasionally results the packets to be deviated away from its path from source to destination in order to balance the load. As a result, packets end up going in loops that do not necessarily converge to the destination in a realistic simulation scenario used in our work. It was discussed in Section 4.10 that our algorithm does not suffer from such looping effect due to the hierarchical algorithm design. Based on the multi-path approach of our routing algorithm, the
backpressure gradients are only applied to a subset $\mu$ of all the existing routes between any source-destination pair. This subset is chosen based on the QoS requirement of our algorithm utilizing ETX metric discussed in Chapter 3.

![End-to-End Delay](image1)

**Figure 4-18: End-to-End Delay (s) vs Load (KB/s), OLSR-BSA, ROSA and DIBAPS**

![Packet Delivery Ratio](image2)

**Figure 4-19: Packet Delivery Ratio vs Load (KB/s), OLSR-BSA, ROSA and DIBAPS**

In our proposed routing algorithm, the QoS is prioritized over the load balancing capability of backpressure algorithm and this ensures that packets are directed to the vicinity of destination node when applying the backpressure queue gradients. As it was discussed in Section 2.11, the original proposal of backpressure algorithm was based on a TDMA system and on a fixed network design. The novelty in our algorithm is that it has aligned backpressure algorithm to the challenges involved in routing of CR-MANETs.
Chapter 4: Backpressure Routing

4.11.2.4.2 Analysis based on Queue Length

On an overall basis, ROSA and DIBAPS follow a downtrend against variations of speed, Load and RND. However, similar to our algorithm, the two state of the art algorithms show a performance gain when the network load is increased from 1 to 10KB/s. This behaviour is normal, as backpressure algorithm necessitates a minimum level of queues to be created in nodes to trigger traffic distribution. In other words, backpressure queue gradients are only applied to nodes when there is a major difference in measurement of their queue levels. This is in line with the formulation of backpressure provided in Section 2.11.2.

Figure 4-20: End-to-End Delay (s) vs RND, OLSR-BSA, ROSA and DIBAPS

Figure 4-21: Packet Delivery Ratio vs RND, OLSR-BSA, ROSA and DIBAPS
In this section, we have analysed the performance of OLSR-BSA, ROSA and DIBAPS based on the average queue length, against variations of Speed, Load and RND. The simulation environment and parameters used for the study in this section has remained the same as the simulation studies provided in introduction of this section (Section 4.11), in order to keep the results comparable. The average queue length is averaged over all the active and relay nodes. This was on the basis that all active and relay nodes participate in relaying the data from source to destinations and their queues are affected by the backpressure algorithm in the routing protocol proposed by this work.

The simulation results for analysis of average queue length against variations of Speed, Load and Relay Node Density are provided in Figure 4-22, Figure 4-23 and Figure 4-24. These results are key in our understanding of the performance of our proposed routing algorithm against the state of the art ROSA and DIBAPS. The stability of average queue length and its impact based on variations of network conditions analysed in this section is key to understanding the impact of backpressure algorithm on our approach. On a fundamental level, as it was pointed out in Section 2.11, backpressure algorithm was proposed to achieve throughput optimality on the basis of balancing queues across the network. Based on the results shown in Figure 4-22, Figure 4-23 and Figure 4-24, the significant spread of error bars for ROSA and DIBAPS (compared against OLSR-BSA) suggests instability of the algorithms in maintaining the queue lengths. Effectiveness of backpressure algorithm can be analysed by how low and balanced queues are distributed against the changes applied to network conditions.
Based on the performance of OLSR-BSA compared to ROSA and DIBAPS shown in Figure 4-22, it can be seen that the proposed algorithm is capable of maintaining stable queue lengths when compared against the state of the art. This is in line with the better end-to-end delay performance of our algorithm against the state of the art depicted in the previous section. The uptrend seen in the results presented in Figure 4-23, indicates that the higher network load adds to the queuing of the intermediate nodes in the network resulting in a higher average on an overall basis. However, as it can be seen, OLSR-BSA reports a lower queue level compared to the two state of the art protocols.

The results shown in Figure 4-24 shows the very interesting trend of OLSR-BSA compared to ROSA and DIBAPS. Performance of OLSR-BSA in terms of average queue length shows a downtrend against RND while ROSA and DIBAPS follow an uptrend. This indicates the capability OLSR-BSA in balancing load when more relay nodes are made available in the network. This is purely due to the backpressure design that was adapted in this work. However, this is not the case for ROSA and DIBAPS due to the looping effect reported in the previous section. The looping effect results excessive
unsuccessful packet delivery which visualizes as added network load in our performance analysis. As this load does not clear in time before the next packet arrival, it creates instability in the queues as it can be seen in all results shown in this section.

4.12 Summary

In this chapter, a load balancing mechanism was implemented in the simulation environment based on the original spectrum-aware OLSR routing algorithm that was introduced in Chapter 3. The queue differential property of backpressure algorithm was utilized to provide load balancing in the CR-MANETs. It was elaborated that due to the TDMA MAC layer requirement of the original backpressure algorithm, implementation of it under the CSMA structure of IEEE802.11 is challenging and a solution was proposed to target it. The original single path OLSR routing algorithm cannot accommodate the proposed solution and hence a multipath design was proposed and successfully implemented. As backpressure algorithm requires distribution of queue related information among nodes in the network, a backpressure signalling mechanism was proposed and integrated in the base signalling structure of OLSR routing algorithm. Furthermore, the backpressure algorithm was unified with the dynamic spectrum structure of CR-MANETs by incorporating SOPs into the load balancing decision making process. As a result of this optimization, the distribution of network load not only utilizes the alternative routes in the network but also take advantage of the SOPs. Hence the spectrum-aware backpressure OLSR not only optimizes route computation based on the predefined cost metric but also incorporates the queue gradients of backpressure algorithm to perform load balancing. In the simulation study, it was confirmed that the backpressure algorithm has significant improvement on PDR in all cases of speed, RND and load variations. Additionally, it was concluded that the proposed algorithm improves the end-to-end delay at high ranges of relay node densities and network loads. To target the excessive signalling load that the backpressure algorithm imposes on the network a dedicated signalling channel was proposed which confines the negative effect of high signalling load to a single channel and avoids interference with data channels.

The performance of OLSR-BSA was further analysed against the two state of the art protocols which shows significant stability and performance gain. Although, a significant gain was achieved in OLSR-BSA in terms of PDR and end-to-end delay but the relatively large variance as shown by error bars, indicates a room for improvement. Due to the extremely dynamic structure of CR-MANETs, nodes suffer from lack of a guaranteed QoS levels. Although backpressure algorithm is known to be throughput optimal under a TDMA MAC structure, to the best of our knowledge, there is no work to support its throughput optimality under a CSMA MAC scenario. One of the unrealistic assumptions in the original idea of backpressure algorithm is that nodes would have instant access to the queue information of the neighbouring nodes which is impossible under a realistic communication environment. As a matter of
fact, the delay caused by distribution of network topology graph and queue information can result in unstable routes to be computed due to inaccuracy of queue or topology information. In order to address the aforementioned challenges, a new approach based on quantum game theory, is studied and evaluated in the next chapter. The motivation for the application of quantum game theory principles to load-balancing, lies on its potential in providing a solution to the aforementioned inaccuracy of information. The main idea behind backpressure algorithm is that balancing queues at nodes in the network would lead to balancing the network traffic, but what if nodes can balance their own traffic before any queues are created (preventing queue/backlog formation in the first place) and maintain this balanced distribution of the traffic throughout the network’s lifetime. The application of Quantum game theory enables nodes to perform load balancing without the need to access any queue information. Quantum game theory and its application in load balancing is the topic of the next chapter of this thesis.
Chapter 5

5 Load Balancing Based on Quantum Game Theory

5.1 Introduction

This chapter presents the third contribution of this thesis which is load balancing based on quantum game theory. In was discussed in Chapter 4 that the main idea behind backpressure algorithm is that balancing the relative queue sizes at nodes in the network would lead to balancing the overall network traffic. But what if nodes can balance their traffic prior to creation of any queues and maintain this balanced distribution of the traffic throughout the network’s lifetime. Quantum game theory provides a framework which enables synchronized entangled decision-making process which can be used to perform load balancing with minimal signalling data. In Section 2.12, we covered the concept of quantum game theory and its connection with load balancing. Quantum game theory provides a framework to utilize entangled particles with the aim of affecting decision-making process of distant players without transmission of any information. This is enabled by the properties of entangled particles which creates a communication-less instantaneous channel which can be used to influence the strategies made by the players in a quantum game. A system is known to be quantum when the behavioral states of that system is in violation of the bell inequalities [81]. One of the properties of a quantum system is the larger accessible space of states which can be utilized to maximize a pre-defined utility function. In this chapter, we show that by utilization of entangles particles in a quantum game setup, a considerable gain can be achieved in traffic management and load balancing. We have proven that the quantum game strategies can be designed to maximize fair distribution of network load and the sender nodes can utilize the full potential of these strategies to perform load balancing. For a detailed understanding of the contribution made in this chapter, the interested reader may refer to our publication [105].

5.2 Quantum entanglement, from concept to application

In Chapter 4, we discussed the significance of load balancing in CR-MANETs and proposed a new algorithm which outperformed the state of the art protocols utilizing backpressure algorithm. We showed that load balancing results higher network throughput and better end-to-end delay. It was discussed that
unbalanced distribution of traffic in the network can result sporadic queues to be created which impacts the QoS performance of the network. Although backpressure proved to be very effective in addressing such problem, however the performance improvement is limited. Furthermore, the simulation results indicate high error bars under dynamic network scenarios (high load/mobility). As the main goal in design of any routing protocol is delivery of packets from source to destination via efficiently computed routes, load balancing can only be integrated into the routing algorithm as a secondary criterion.

This was our motivation to address load balancing from a completely different aspect. The main barrier in load balancing is the dynamic structure of CR-MANETs which doesn’t allow an efficient signalling medium where queue information can be shared. All load balancing techniques require signalling information to be efficiently distributed across nodes in the network so that routing algorithms can plan strategies which balances load across the network. Regardless of how accurate the signalling protocols are designed; the received information does not reflect the most up to date status of the network load. Consequently, utilization of this information in the network layer does not converge to an optimal solution for load balancing. As it was discussed in Section 2.12, quantum entanglement provides a tool to synchronise decisions of remote players without the need to transfer any information. This is achieved by the properties of entangled particles. If two particles are entangled, regardless of their separation distance, the measurement of spin of one particle affects the spin of the other entangled particle. If this spin is assigned as a tuning operator for the decision-making process of two remote quantum game players, then with pre-calculated probabilities, the outcome of this game can be predicted and controlled.

The motivation in the proposal made in this chapter lies in the fact that quantum entanglement can enable distant players to harmonise their decision-making for a mutually agreed win scenario. In this Chapter, we have utilized this capability for load balancing in ad hoc networks, however this property can be used in many different interdisciplinary research topics. We look at quantum game theory as a communication-less mean for transmission of an event (although it is not really considered to be a transmission). However, based on the quantum nature of the analysis, the accuracy of communicating this event in known to be less than 100%. In this work, we have taken two simplified scenarios where there is potential to analyse the capabilities of quantum load balancing. In this work, we have shown the maximum win probability when quantum game theory is applied compared to a non-quantum case scenario. This can stand as the theoretical limit for application of quantum game theory specific to the load balancing scenarios analysed in this Chapter. Furthermore, the simulation study stands as the first proof of concept in the literature which shows the effectiveness of quantum load balancing in fair distribution of traffic in the network. The novelty of the work presented in this Chapter stands as a foundation proof of concept which shows the extensive capabilities of Quantum load balancing in communication networks.
5.3 Quantum Game and Entanglement

As it was explained in Section 2.12, quantum game theory focuses on design of games that maximizes players’ utility enabled by the properties of entangled particles. This section focuses on the entanglement of particles and the two scenarios where nodes in ad hoc networks can benefit from the properties of these quantum games.

5.3.1 Rotation and quantum strategies

The focus of this section is to clarify the method by which quantum game theory benefits from quantum operators to suggest best strategies to players. It is worth noting that the main difference between classical game theory and quantum is that in quantum game theory, players are able to use entangled particles. Due to this unique difference, an explanation about entangled states and quantum operators is necessary to understand this section.

In general, entangled states are created when an additive physical property of two particles, e.g. their spin, is measured. The main characteristic of entangled particles is that measuring the physical property of one of the particles results the other one to be reduced to a definite state. For instance, consider an entangled state comprised of two $\frac{1}{2}$-spin particles. Having knowledge about the spin of one particle is equivalent to knowledge of the other. This happens instantaneously regardless of the spatial distance between the two particles. It is worth noting that this process is not considered as a signaling mechanism. In other words, no information is transmitted between the two particles. This is addressed in quantum mechanics as non-locality whose existence was proven by Bell in 1964 [106].

To illustrate different parts of our theory, we limit ourselves to particles with spin of $\frac{1}{2}$. Our formulation can be easily generalized to the higher spin particles. This generalization is discussed in Section 5.3.4 for particles with spin 1. So, we consider specific entangled state of $\frac{\uparrow\uparrow}{2}$ + $\frac{\downarrow\downarrow}{2}$, which represents two entangled electrons with spins $\frac{1}{2}$. This particular case is the subject of interest for all of this section. Our game is comprised of two players. Suppose that one of these particles is available to player 1 and the other one is available to the other player. Each player can rotate their electron (as their particle) individually and independently.

Rotation operators in three dimensions can be parameterized by three angles denoted by $\theta, \varphi$ and $\alpha$. For $\frac{1}{2}$-spin particles, the rotation operators has a $2 \times 2$ matrix representation. So, we may represent a rotation operator by $U(\theta, \varphi, \alpha)$ with specific matrix shown in Eq. 5.1 to perform this rotation.
\[ U \equiv R_n(\hat{n}) = \begin{pmatrix} \cos\left(\frac{\alpha}{2}\right) - in_x \sin\left(\frac{\alpha}{2}\right) & (-in_z - n_y) \sin\left(\frac{\alpha}{2}\right) \\ (-in_x + n_y) \sin\left(\frac{\alpha}{2}\right) & \cos\left(\frac{\alpha}{2}\right) + in_z \sin\left(\frac{\alpha}{2}\right) \end{pmatrix} \quad \text{Eq. 5.1} \]

Where in this matrix \( n_i \) are the components of unit vector \( \hat{i} \) with polar coordinates of \( \theta \) and \( \phi \). In fact, this is a rotation characterized by \( \alpha \) around \( \hat{i} \) vector, which has components shown in Eq. 5.2.

\[
\begin{align*}
    n_x &= \sin \theta \cos \phi \\
    n_y &= \sin \theta \sin \phi \\
    n_z &= \cos \theta
\end{align*} \quad \text{Eq. 5.2}
\]

Spinor \( |\uparrow\rangle \) has a matrix representation as \( \begin{pmatrix} 1 \\ 0 \end{pmatrix} \) and rotating it by an operator can be calculated according to Eq. 5.3.

\[
U|\uparrow\rangle = R_n(\hat{n}) \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} \cos\left(\frac{\alpha}{2}\right) - i\cos(\theta)\sin\left(\frac{\alpha}{2}\right) \\ -i\sin\theta e^{i\phi} \sin\left(\frac{\alpha}{2}\right) \end{pmatrix}. \quad \text{Eq. 5.3}
\]

So, we have \( U|\uparrow\rangle = \left( \cos\left(\frac{\alpha}{2}\right) - i\cos(\theta)\sin\left(\frac{\alpha}{2}\right) \right)|\uparrow\rangle + \left( -i\sin\theta e^{i\phi} \sin\left(\frac{\alpha}{2}\right) \right)|\downarrow\rangle \). The same process can similarly be applied to \( |\downarrow\rangle \). The absolute value of the coefficients powered by 2 determines the probability of obtaining each state after measuring the spin. Without loss of generality, the rotations around y axis can be considered with utilizing \( \theta = \frac{\pi}{2} \) and \( \phi = \frac{\pi}{2} \).

Suppose that \( U_1\left(\frac{\pi}{2}, \frac{\pi}{2}, \alpha_1\right) \) and \( U_2\left(\frac{\pi}{2}, \frac{\pi}{2}, \alpha_2\right) \) are the rotation operators used by player 1 and 2, respectively. The result of this rotation is shown in Eq. 5.4.

\[
U_1U_2\left(\frac{|\uparrow\rangle + |\downarrow\rangle}{2}\right) = \frac{1}{\sqrt{2}} (\cos\left(\frac{\alpha_1 + \alpha_2}{2}\right)|\uparrow\rangle - i \sin\left(\frac{\alpha_1 + \alpha_2}{2}\right)|\downarrow\rangle + -i \sin\left(\frac{\alpha_1 + \alpha_2}{2}\right)|\uparrow\rangle + \cos\left(\frac{\alpha_1 + \alpha_2}{2}\right)|\downarrow\rangle). \quad \text{Eq. 5.4}
\]

So, it can be concluded that Eq. 5.5 represents the probability of obtaining \( |\uparrow\uparrow\rangle \) after measuring the spin of both particles.
The formulation developed above can be generalized for higher spin particles. Higher spins can be used for games with more available strategies. The formulation above is used in Section 5.3.3 and Section 5.3.4, for two fundamental network topologies to model the best quantum advises for players of the game.

5.3.2 Quantum advice and entangled opinions

In this section, we explain how quantum mechanics can be utilized to create advices which leads to improvement in network flow management. To do this, we define a game whose players are sender nodes in network. If the sender nodes in the network choose a strategy that optimizes energy efficiency and load balancing via the intermediate nodes, they win the game, otherwise, they have lost. As it was demonstrated in the previous section, quantum entanglement provides a tool which increases the win probability of players in comparison with classic scenarios.

It must be noted that the calculations involving quantum games depends on the network topology. However, there are techniques which can expand this idea to larger network topologies but the concentration of this chapter is to proof the fundamental concept behind this novel idea. Hence, two simplified topologies of doublet and triplet are investigated. The reason for choosing such topologies is to firstly proof the concept of load balancing using quantum game theory and secondly the potential of these topologies to be generalized to more complex cases. The basic idea behind our theory is that entangled particles can be used for entangling opinions of two distant players (aka sender nodes). Hence, utilize this correlation to impact the decision of one player which leads to a reduced number of allowed choices in the other player of the game. With a lower number of available choices, the other player will be forced to play towards satisfying the utility function represented in this work and achieve load balancing.

The formulation presented in the previous section can be used in a realistic network traffic management game to provide an advice to the players to increase their win probability. In the Sections 5.3.3 and 5.3.4 we will illustrate design of a traffic management game based on the proposed game theory and the calculations provided.

5.3.3 Doublet topology

Doublet topology is defined as a case where two relay nodes are shared among two distant senders to forward the data to two destination nodes. Since each sender node has two available next hop to route

\[
\frac{1}{2} \left[ \cos \left( \frac{\alpha_1 + \alpha_2}{2} \right) \right]^2
\]

Eq. 5.5
their data, we have utilized a 2-state particle entanglement scenario. Therefore, $\frac{1}{2}$-spin particles are used throughout this section.

Initially we need to develop our quantum game based on the doublet topology. At the starting point, let’s consider two entangled particles with spin $\frac{1}{2}$ have been produced. The state of the particles can be represented as $\frac{1}{2} \left( |\uparrow\uparrow\rangle + |\downarrow\downarrow\rangle \right)$. In simple words, this means that highly entangled particles are considered but this problem can only be solved by considering other entangled states. Let us assume that the first particle is given to player 1 and the second one to player 2.

Now, we need to discuss how players could influence each other’s decisions by using entanglement and the rotational operators. Every player can independently rotate its particle. Player 1 and 2 use the rotation operator $U_1(\alpha_1)$ and $U_2(\alpha_2)$ for rotating the particles allocated to them. Therefore, the expression for entangled state after players rotate their particles is based on Eq. 5.6.

$$|\psi\rangle = U_1 U_2 \left( \frac{1}{2} \left( |\uparrow\uparrow\rangle + |\downarrow\downarrow\rangle \right) \right).$$

Eq. 5.6

The topology configuration of the doublet case is shown in Figure 5-1. We will first go through the game aspect of this topology and then cover the quantum counterpart which is applicable to this network configuration.

![Doublet topology](image)

**Figure 5-1: Doublet topology**

As shown in Figure 5-1, let us consider node A and B as sender nodes and node G and H as destination. This simple topology is setup so that node A and B have no direct access to the destination nodes G and H and any data transmission has to be relayed over the intermediate nodes C and D. In
other words, A and B have two different routing paths to reach node G and H. We denote the links that connect any sender node to node C as +1 and the links that connect any of the sender nodes to node D as -1. Suppose that there are two categories of data transmissions, i.e. High Bit Rate (HBR) and Low Bit Rate (LBR). We characterize the data emerging each node to have a probability of \( P \); without loss of generality, one can suppose that \( P \geq 0.5 \).

To complete the traffic game, we have to define a winning case for the players in the game. Toward this aim, a general network configuration should be taken into consideration. By taking factual concepts from traffic management in the topic of networks we can come up with a winning scenario in our game. HBR data needs considerably more channel capacity for a longer duration compared to LBR. As a LBR data transmission does not require high channel capacity, it can be considered as an opportunity to other HBR flows. On the other hand, from the energy consumption point of view a LBR flow can be engineered to result less relay node involvement to forward the data which leads to lower energy consumption. Every intermediate node consumes energy when forwarding data and so to obtain more efficiency we try to minimize their utilization under a LBR traffic. Additionally, an efficient utilization of intermediate nodes creates more transmission opportunities for the other nodes in the network.

Having the above factual information in mind, the rule of the game can be defined as follows. If at least one of sender nodes requires to send HBR data, it is more efficient to evenly distribute the traffic across the relay nodes in order to efficiently support the high demands of the HBR traffic. In contrast, to save energy and provide more relaying opportunities to other nodes in the network, it is best for the LBR traffic to be forwarded via the same relay nodes. Therefore, in the former case players win the game if they choose the opposite relays (which leads to different routing paths) and in the other case, they win by choosing the same relays to forward their data. In order to enable such game scenario, we will next explain how entanglement is capable of increasing the probability winning this game.

In the classical viewpoint, the best strategy, which maximizes win probability, is to choose different paths at all time (note that this is the result of supposing \( P \geq 0.5 \)). So, one of the players always uses the path +1 and the other one uses the path -1. This strategy has the win probability equals represented by Eq. 5.7.

\[
P_{\text{win}}^{\text{Classic}} = 1 - (1 - P^2).
\]  

Eq. 5.7

The win probability is written by considering that the probability of HBR data type emergence is equal to win probability (using different paths in the case of at least one sender node that is sending HBR data). Eq. 5.7 represents the emergence probability of HBR data in at least one of the sender nodes.

In the contrary to classical view, quantum entanglement is capable of increasing this probability. Under a quantum scenario, we consider the state function given by Eq. 5.6, which is obtained after
operating rotations of $U_1$ and $U_2$. Each player rotates its particle spin by $\theta_H$ in the case of HBR traffic and by $\theta_L$ otherwise. After rotation, each player selects its path based on the spin of its particle. A spin of $+\frac{1}{2}$ results the path $+1$ to be taken and a spin of $-\frac{1}{2}$ leads to utilization of path $-1$. Therefore, by using Eq. 5.5, the win probability can be calculated using the expression given by Eq. 5.8.

$$P_{\text{win}}^{\text{Quantum}} = \frac{1}{2} \left( \sin^2 \left( \frac{\theta_L + \theta_H}{2} \right) + \frac{1}{2} \left( \cos^2 (\theta_L) + \sin^2 (\theta_H) \right) \right).$$ \hspace{1cm} \text{Eq. 5.8}$$

The formula given by Eq. 5.8 represents the total probability for selection of opposite paths in case of at least one HBR data and the same paths otherwise. The main task here is to maximize Eq. 5.8 as a function of $\theta_H$ and $\theta_L$ to result the best values of them. This operation can be performed for all values of $P$ but without loss of generality, we have assumed $P = \frac{1}{2}$ for the rest of calculations provided in this section. This assumption is to be able to numerically maximize Eq. 5.8 and obtain numerical results for the values of $\theta_H$ and $\theta_L$. However, any other values of $P$ is also acceptable. For this value of $P$, Eq. 5.8 results a maximum of 85% success which is presented by Eq. 5.9.

$$\theta_H = 292.5$$
$$\theta_L = 202.5$$ \hspace{1cm} \text{Eq. 5.9}$$

Based on the above findings, we can design a routing protocol that incorporates the results of our game design. The main principles of this protocol are based on advices listed below which is given to each sender node in the network.

1. In the case of sending HBR data, rotate your particle spin by $\theta_H$.
2. In the case of sending LBR data, rotate your particle spin by $\theta_L$.
3. Measure the spin of your particle,
4. If you obtained $+\frac{1}{2} (-\frac{1}{2})$, choose path $+1 (-1)$.

As it was mentioned before, the above routing protocol needs to be designed to maximize win probability. Route selection probabilities show a very promising result for balancing the network load among the intermediate nodes. The probabilities for the situation where at least one of the nodes is sending HBR traffic is listed in Eq. 5.10.

$$P(+1,-1) = 42.5\%$$
$$P(-1,+1) = 42.5\%$$ \hspace{1cm} \text{Eq. 5.10}$$
\[ P(+1,+1) = 7.5\% \]
\[ P(-1,-1) = 7.5\% \]

Similarly, the probabilities for when nodes send LBR traffic is listed in Eq. 5.11.

\[ P(+1,-1) = 7.5\% \]
\[ P(-1,+1) = 7.5\% \]
\[ P(+1,+1) = 42.5\% \]
\[ P(-1,-1) = 42.5\% \]

The probabilities resulted by the calculations above are symmetric which leads to balanced distribution of load under a HDR traffic and opportunity creation (or lower power consumption) under a LDR scenario. The simulation study in Section 5.8.1 supports the gain achieved in the proposed theory under a doublet topology configuration.

5.3.4 Triplet topology

The same formulation that was developed for the scenario of doublet topology can be applied to the triplet topology. The main difference here is that the number of relay nodes to forward the traffic from source to destinations is three and based on that the formulation needs to be adapted to efficiently utilize the capability of the third relay node in the quantum game.

\[ \text{Figure 5-2: Triplet topology} \]
Chapter 5: Quantum Load Balancing

The triplet topology is designed to further illustrate the mechanism with which the quantum game theory can be expanded to cover more complex topology cases. Toward this aim, the topology configuration of the triplet scenario is shown in Figure 5-2.

As a result of addition of the third relay, source nodes A and B have three choices to relay their data to the destination nodes G and H. We have characterized the three routing paths to be +1, 0, -1. The same winning strategies that applies to the doublet topology is applied here as well. In the classical formulation, the maximum likelihood of winning for sender nodes (as quantum players) which is resulted by a non-overlapping strategy is 75%.

To model this problem using the quantum game theory we need to consider three accessible intermediate paths which requires a 3-state spin (particles with spin of 1). For this case, rotation matrix can be formulated according to Eq. 5.12

\[
U(\theta, \alpha, \varphi) = \begin{pmatrix}
\frac{1}{2} e^{i(\alpha+\varphi)}(1 + \cos \theta) & -\frac{1}{\sqrt{2}} e^{-i(\alpha)}(\sin \theta) & \frac{1}{2} e^{-i(\alpha+\varphi)}(1 - \cos \theta) \\
\frac{1}{\sqrt{2}} e^{i(\varphi)}(\sin \theta) & \cos \theta & -\frac{1}{\sqrt{2}} e^{i(\varphi)}(\sin \theta) \\
-\frac{1}{2} e^{-i(\alpha+\varphi)}(1 - \cos \theta) & \frac{1}{\sqrt{2}} e^{-i(\alpha)}(\sin \theta) & \frac{1}{2} e^{-i(\alpha+\varphi)}(1 - \cos \theta)
\end{pmatrix}
\]  
Eq. 5.12

Similar to doublet case, a quantum entangled state comprised of two 1-spin particles needs to be considered. In order to violate the bell inequalities and be able to benefit from the quantum states we have focused on the state formulated by Eq. 5.13.

\[
|\psi\rangle_{in} = \begin{pmatrix}
0.60 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + 0.65 \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} + 0.47 \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} & \\
\end{pmatrix}
\]  
Eq. 5.13

Similar to the previous doublet topology, suppose that two types of data (HBR and LBR) traffic can emerge at each sender node with probability of \(\frac{1}{2}\). Using the state function in Eq. 5.13 and rotation operator presented in Eq. 5.12, the winning probabilities can be calculated. Similar to the previous case the advisors need to tell their players to rotate their particles by \(\theta_H, \alpha_H, \varphi_H, \theta_L, \alpha_L, \varphi_L\) in the case of HBR (LBR) transmission. After formulation of the win probability and maximization of it as a function of \(\theta_H, \alpha_H, \varphi_H, \theta_L, \alpha_L, \varphi_L\) the angles in the optimum point are obtained and listed in Eq. 5.14.
\[ \theta_H = 142.6 \quad \theta_L = 81.7 \]
\[ \alpha_H = 60.0 \quad \alpha_L = 340.2 \quad \text{Eq. 5.14} \]
\[ \varphi_H = 27.1 \quad \varphi_L = 244.7 \]

For the values listed in Eq. 5.14, the \( P_{\text{win}} \) is calculated to be 91%. Hence, utilization of entangled particles can increase the probability of winning by 16% compared to classical methods. Similar to the doublet topology, the probabilities of choosing opposite paths is symmetric in the triplet topology which can theoretically lead to load balancing at the intermediate relay nodes. This claim is supported by the simulation study of this theory provided in Section 5.8.2.

When sender nodes rotate the spin of their particles by above angles (Eq. 5.14), the strategies of choosing a joint path compared to a load balanced one is chosen by the sender nodes based on various transmission probabilities.

<table>
<thead>
<tr>
<th>PROBABILITIES</th>
<th>HH (%)</th>
<th>HL (%)</th>
<th>LH (%)</th>
<th>LL (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P (+1, 0) )</td>
<td>10</td>
<td>19</td>
<td>16</td>
<td>0</td>
</tr>
<tr>
<td>( P (+1,-1) )</td>
<td>20</td>
<td>10</td>
<td>16</td>
<td>0</td>
</tr>
<tr>
<td>( P (0,+1) )</td>
<td>10</td>
<td>16</td>
<td>19</td>
<td>0</td>
</tr>
<tr>
<td>( P (0,-1) )</td>
<td>15</td>
<td>9</td>
<td>19</td>
<td>0</td>
</tr>
<tr>
<td>( P (-1,+1) )</td>
<td>20</td>
<td>16</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>( P (-1, 0) )</td>
<td>15</td>
<td>19</td>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td>( P (+1,+1) )</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>36</td>
</tr>
<tr>
<td>( P (0, 0) )</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>42</td>
</tr>
<tr>
<td>( P (-1,-1) )</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>22</td>
</tr>
</tbody>
</table>

In Table 5-1, HL means Source Node 1 sends HBR and as an example Source Node 2 sends LBR data. For the angle values listed in Eq. 5.14, the win probability is calculated which shows that utilization of entangled particles can increase the probability of winning by 16% compared to classical methods. Similar to the doublet topology, the probabilities of choosing opposite paths are symmetric in the triplet topology which can theoretically lead to load balancing at the intermediate relay nodes. This claim is
supported by the simulation study of this theory provided in Section 5.8. The quantum advice for the case of triplet topology, is given in the next section.

5.3.5 Quantum advice

In this section, the quantum advice for both the doublet and triplet topologies is summarized. Exploitation of these quantum advices by the players in the load balancing game, leads to load balancing in the network. The quantum advices are listed below.

- **Doublet topology:**

  In the case of doublet topology, the quantum advice is as stated below.
  1. In the case of sending HBR data, rotate your particle spin by $\theta_H$,
  2. In the case of sending LBR data, rotate your particle spin by $\theta_L$,
  3. Measure the spin of your particle,
  4. If you obtained $\frac{1}{2}$ ($\frac{1}{2}$), choose path +1 (-1).

  Theoretically under the doublet topology the win probability is increased by 10% compared to classical methods.

- **Triplet topology:**

  In the case of triplet topology, the quantum advice is as stated below.
  1. In the case of sending HBR data, rotate your particle spin by $\theta_H, \alpha_H, \varphi_H$,
  2. In the case of sending LBR data, rotate your particle spin by $\theta_L, \alpha_L, \varphi_H$,
  3. Measure the spin of your particle,
  4. The result of your spin measurement decides which path to take.

  Theoretically under the triplet topology the probability of win is increased by 16% compared to the classical methods.

5.4 Ad hoc Load Balancing Problem Definition

As mentioned earlier, the problem targeted in this chapter is load balancing in ad hoc networks. As it was elaborated in Section 2.10, load balancing is one of the major challenges in ad hoc networks. The work presented in this chapter and the simulation followed by it is to introduce an entirely novel load
balancing technique using Quantum Game Theory as explained in Section 5.3. Furthermore, to incorporate this load balancing algorithm into a fully functional routing protocol. The main focus of this section is to define the problem of load balancing from our research point of view.

One of the cases in ad hoc networks where load balancing is critically important is when there exist some relay nodes among the source/destination pairs and the relaying of traffic is shared among them. An example of such case which has been studied in this work is shown in Figure 5-3, where source nodes 5 and 6 share relay nodes 3 and 4 to forward their traffic to destination nodes 1 and 2.

![Figure 5-3: Ad hoc Topology Scenario](image)

Balance distribution of network load among relay nodes 3 and 4 has direct impact on end-to-end delay, packet jitter, throughput, network stability and connectivity. Current conventional ad hoc routing protocols do not take load balancing into account which is the motivation of this research in targeting this problem. Referring back to Figure 5-3, in the current ad hoc routing protocols source nodes 5 and 6 randomly choose relay nodes 3 and 4 as their next hop route without considering the level of traffic load that is being sent by them. Under high network load conditions this would result unbalance distribution of load among the relay nodes. The unbalanced relaying of data packets creates random queues at relay nodes which results relatively high end-to-end delays. On the other hand, under heavy network load conditions, packets perform random switching among free and busy relays which results unstable packet jitters. Additionally, depending on the network load, long queues at relay nodes can cause buffer overflows which affects PDR as well as the throughput. Hence, we can conclude that balance distribution of network load among relay nodes can have great impact on QoS related measures.

### 5.5 OLSR Routing Protocol as the Baseline of Implementation

The importance of balanced relaying of the network traffic forwarded by relay nodes is extensively studied in this work. Via simulation studies we have shown that balance forwarding of the network traffic among the relays can directly impact and improve QoS factors such as end-to-end delay and jitter
in the network. As it was explained in Section 2.10, routing protocols are the most suitable agents to implement load balancing algorithms. Towards this aim, OLSR [107] is used as the base routing protocol to implement our load balancing algorithm. OLSR is a pro-active routing protocol which uses a proactive signaling mechanisms to create and update the network topology graph. The load balanced routing protocol implemented by this work is so called Quantum Load Balanced-OLSR (QLB-OLSR). The main reason why OLSR was the routing protocol of choice in this work is its proactive structure. It is important to note that the load balancing mechanism implemented in this work requires the full network topology graph. Having full network graph is necessary in identification of the segments of the network where Quantum Load Balancing (QLB) can be applied. The identification process consists of evaluating whether a number of relay nodes are common among the source/destination pairs as shown in Figure 5-3. As OLSR plays an important role in the implementation of our load balancing algorithm, in the next section we will explain the base operation of OLSR which is essential to the functioning of our routing algorithm.

5.5.1 OLSR Basic Operation

OLSR is a proactive routing protocol meaning that the protocol is designed based on periodic distribution of signaling messages. Some of the signaling messages are set to be sent locally to the vicinity of the current node and some are designed to be propagated throughout the network and help other nodes complete their topology graph. Proactive routing protocols cache routes in their routing table and hence have routes ready upon request. As signaling messages are sent periodically, routes are updated regardless of whether there is any data packets to be sent. During each full cycle of the signaling mechanism, routes are updated at every individual node in the network. On the other hand, in a reactive routing protocol routes are created and updated as soon as there is demand for routing of application layer traffic. When there is no data packets to be sent, routes remain out of date until there is a request for another transmission. A reactive routing protocol lowers the signaling load in the network at the cost of higher end-to-end delay.

OLSR uses HELLO messages to perform neighbor sensing. HELLO messages are periodically broadcasted by every node in the network to update the 1-hop neighborhood of each node. The base implementation of OLSR uses MPRs (MultiPoint Relays) which is a signaling optimization used to minimize signaling load resulted from the excessive broadcasts of the routing protocol [107]. However, due to the looping effect and lack of stability, MPR optimization has been completely disabled in our implementation of OLSR in this work.

Another type of signaling messages designed in OLSR is called Topology Control (TC) messages. TC messages are sent periodically by each node in the network to update the network topology graph. Upon reception of TC messages, nodes extract the address list and message originator address and use
the most up to date information acquired by HELLO messages to initially compute the topology graph and consequently create the routing table.

5.6 Implementation of QLB in OLSR

Quantum Load Balancing was explained from the quantum game theoretical point of view in section 2.12 of the literature study. In Section 5.3 we proposed a quantum game to target the problem of load balancing in ad hoc networks on a fundamental level. In this section, we have covered the implementation of the QLB in OLSR as the baseline routing protocol in the simulation environment.

5.6.1 Topology Identification

As it was covered in Section 5.3, we have modelled our quantum game theory based on the cases where we have either 2 common Relays or 3 common Relays among the source/destination pairs. Our quantum load balancing algorithm can automatically detect the existence of 2 or 3 common relays and based on that use the appropriate probability sets which is the result of the mathematical modeling summarized in Sections 5.3.3 and 5.3.4. Hence, adaptation of the network topology to the theoretical definition of load balancing strategies that QLB is capable of managing is vital. The assumption in this work is that OLSR operates normally to compute the routing tables and the QLB algorithm affects the computed routes by OLSR when necessary. Based on QLB algorithm, the source nodes are notified via OLSR signaling if they have common relays with the destinations nodes. We have modified the structure of OLSR’s routing table so that for every destination, it accommodates the address of possible common relays it has with the current node. In order to achieve this, we have added a new mechanism in the way OLSR’s HELLO messages are processed. When a node receives HELLO messages from different senders which have one common address listed in their address field, then the sender nodes with the common address fields would be listed as the common relays for that specific destination in the routing table. The destination will be taken from the common address in the sender’s address field. By this method, OLSR’s proactive signaling mechanism helps the source nodes to identify the common relays.

5.6.2 Traffic Based Load Balancing

It was explained in Section 5.3 that our quantum game for load balancing has been designed based on the level of emerging application layer traffic. There are two types of traffic defined in this work, one representing a high network load and the other a low network load. Based on these two categories of traffic the quantum game defines two strategies, one being load balancing and the other resource conservation. The logic behind defining these two strategies is that under a low load condition the traffic can be relayed over one common relay node which results conservation resources. On the other hand,
Chapter 5: Quantum Load Balancing

under a heavy load condition, its best to evenly distribute the traffic over the common relay nodes based on the reasons explained in Section 5.4. The probabilities calculated in Sections 5.3.3 and 5.3.4 are based on maximization of winning probability in the load balancing quantum game theory proposed by this work.

5.6.3 Instantaneous Load Balancing

One of the main challenges faced in the implementation of the quantum load balancing is the assumption that source nodes which share common relays, start their transmission instantaneously. Looking back at Figure 5-3, quantum load balancing game is theoretically defined with the assumption that source nodes 5 and 6 would always have a full data buffer and for every packet that is being sent by source node 5 there is another packet being simultaneously transmitted at the exact same time by node 6. It is only with this assumption that the probability of choosing either relay node 3 or 4 as the next hop can be decided based on the probabilities calculated in Sections 5.3.3 and 5.3.4. However, this assumption is impractical under a realistic network environment where nodes can start and end their transmission at unexpected random times; this is resulted by the fact that in a contention based channel access mechanism, such as the IEEE802.11 MAC layer, packets may experience random delays which makes synchronization of their transmission time almost impossible and impractical. To address this problem, we came up with the idea of periodic time scheduling in order to apply the symmetry of the probabilities resulted by quantum load balancing over a short time interval $\Delta t$. In a simulation environment, probabilities are managed by generation of random numbers and matching their outcome space to the value of the intended probabilities in the quantum game case. The idea is that at every time interval $\Delta t$, a new random number is generated and cached to be used by source nodes 5 and 6 during the next time interval defined as $\Delta t$. During this time the source nodes use the cached value of the random number to implement the game strategies dictated by the quantum load balancing.

5.6.4 QLB Algorithm

In Section 5.6.1, it was elaborated that prior to applying the quantum load balancing, a topology identification process is required. This mechanism enables our algorithm to identify the potential relays for applying the QLB algorithm. The QLB algorithm requires two inputs from the node to be able to perform load balancing. One of which is the destination node where the data packet is being sent to and the other is the current application layer transmission data rate. The knowledge of application layer data rate and the destination nodes helps the algorithm to apply the correct quantum strategy based on the quantum advice detailed in Section 5.3.5. Figure 5-4 summarizes the pseudocode of our proposed QLB algorithm in this work. The complete line by line analysis on the QLB algorithm is discussed next.
The algorithm starts at line 1 with an IF Statement containing a listener call from application layer for data transmission. The activation of this listener is the trigger point for execution of our QLB algorithm. The listener call carries the object appDataReq. In line 2, while there is data to be sent in the appDataReq object, the algorithm needs to evaluate the suitability for load balancing. Initially the destNode is extracted from the appDataReq object which as explained before, it is one of the inputs required to perform the QLB algorithm. As it was explained in the Section 5.6.1, QLB algorithm performs topology matching to identify the potential topologies where load balancing can be applied. The hasCommRelay method in line 4 returns a Boolean to confirm that the current node has a common relay with the destination node. Under the condition where this relay exist, then the QLB algorithm can apply the load balancing mechanism. At line 5, the data rate of the application layer traffic is extracted from the appDataReq object, hence the application layer is required to attach the current traffic data rate with the traffic request objects. As it was elaborated in Section 5.6.3, a quantum cached probability scheduling mechanism is implemented by this work to address the requirement of the game strategies defined by quantum load balancing. At line 6 the value of this cached probability is requested by reqCurrCachRandNum method and stored in chachRandNum. At line 7 and 12 the algorithm identifies whether the identified topology is categorized under a 2 or 3 common relay case. Based on the number of common relays, at line 8 and 13 the algorithm requests the probability value from the RelayProbTable.prob method with the parameters destNode, appDataRate, and chachRandNum. If the identified topology is categorized as a 2 or 3 common relay case, then at line 9 and 14 the algorithm requests the next hop relay from the RelayProbTable.probNxtHop method with the parameters destNode, appDataRate, and chachRandNum. The algorithm then proceeds to route the application data from the current node to the next hop relay node.
of common relays (2 or 3) the appropriate QLB probability which is extracted from the precomputed tables, will be used to find the next hop relay for routing of the application data packet. At line 9 and 14, the `probReq` method is called on the appropriate data set utilizing the objects `destNode`, `appDataRate` and `chachRandNum`; these three are the inputs required to match a next hop relay in the appropriate QLB probability table. Finally, at line 18, using the `routeAppData` method the `nxtHopRely` is used to route the current data packet from the `appDataReq` object. Going back to line 4, if there is no common relay between the destination and the current node, at line 21, the data packet would be routed using the normal OLSR routing algorithm. This assures that the QLB algorithm harmonizes with the normal route computation mechanism implemented in OLSR.

Based on QLB algorithm, when there are application layer data packets at source nodes, quantum entanglement and game theoretical strategies are used to manage the traffic relaying among the source/destination nodes which share 2 or 3 common relays. In case of high load traffic the algorithm tries to maximize even distribution of the traffic flow among relay nodes. In contrast, under a low load traffic, the flows are forced to utilize only one of the relays as the forwarding node and this results resource conservation. The fundamental logic behind QLB algorithm is that even load balancing is mainly beneficial to high load traffic than low load traffic flows.

5.7 QLB-OLSR Simulations and results

In this section, we used simulation study to analyze the performance of the proposed joint load balancing and routing algorithm (QLB-OLSR). The simulations were implemented using OMNET++ Discrete Event Simulator.

5.7.1 Simulation Setup and Assumptions

We consider two cases of load balancing for the simulation study, one with 2 relay nodes and the other with 3 relays. The network topology under study by this work is shown in Figure 5-5. The topology shown in this figure is the 3-Relay topology setup whereas in the case of 2-relay setup `node-7` is eliminated. We have setup `node-5` and `node-6` as source nodes with `node-1` and `node-2` as destination nodes. In the case of 2-relay study `node-3` and `node-4` are the only relays to forward the traffic flow to destination nodes. However, `node-7` is added to this setup as the third relay.

To be able to accurately evaluate the performance of QLB-OLSR, we have taken two traffic types into consideration i.e. high and low. The low load traffic produces Constant Bitrate (CBR) at the rate of 10Kbps and the high load traffic is generated at the rate of 1Mbps. Under a realistic network environment, source nodes may randomly change their traffic load. As the target of this work is to evaluate the performance of the protocol under the worst-case scenario, this behavior has been simulated.
using a random scheduling mechanism which randomly switches the traffic load between high and low states at every $\Delta t$ seconds. $\Delta t$ is set to be 10s in this work.

**Figure 5-5: 3-Relay Simulation Setup**

 Hence at every 10 seconds the source node-5 and node-6 would randomly and independently switch their application layer traffic between 10kbps and 1Mbps. This helps us evaluate the convergence time of the QLB-OLSR to changes in the network traffic and mainly its effect on the QoS factors.

The main simulation assumptions used in this work are listed below:

1) All nodes are equipped with IEEE 802.11 WLAN interfaces

2) Sender nodes have Quantum entanglement capability

3) Sender nodes have spin rotation and measurement capability

Points 2 and 3 of the simulation assumptions are necessitated by the quantum game theoretical approach used in this work.

### 5.7.2 Simulation Parameters

OMNET++ was used as the simulation platform to implement and evaluate the performance of QLB-OLSR. The simulation was left to run for a period of 9000 seconds. To improve confidence level in the results and reduce the error, the results are averaged over 50 simulation runs with different seed-sets. CBR traffic was used as the application layer traffic in this work. Furthermore, UDP was the choice of transport layer protocol in our work. The reason behind this decision was that UDP does not have an automatic re-transmission mechanism (as opposed to TCP) and this would help us highlight the actual
performance of our algorithm regardless of any data recovery mechanisms implemented in the transport protocols. Additionally, IPv4 was used as the network layer protocol.

As it was mentioned before, IEEE 802.11g was the choice of MAC and PHY layer protocol which is one of the most commonly used MAC protocols in simulations involving ad hoc networks. All the simulation parameters are listed in Table 5-2. It is very important to consider that for all the statistical averaging and error bar analysis performed on the simulation results presented in this chapter a confidence level of 95% has been used.

<table>
<thead>
<tr>
<th>Table 5-2: Simulation Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulation Parameters</td>
</tr>
<tr>
<td>Simulation Platform</td>
</tr>
<tr>
<td>Simulation Time</td>
</tr>
<tr>
<td>Number of runs</td>
</tr>
<tr>
<td>Total Nodes</td>
</tr>
<tr>
<td>(Sender/ Receiver)</td>
</tr>
<tr>
<td>(Relays)</td>
</tr>
<tr>
<td>Application</td>
</tr>
<tr>
<td>Application Packet Size</td>
</tr>
<tr>
<td>Transmission Interval</td>
</tr>
<tr>
<td>Low Load</td>
</tr>
<tr>
<td>High Load</td>
</tr>
<tr>
<td>Transport Protocol</td>
</tr>
<tr>
<td>Network Protocol</td>
</tr>
<tr>
<td>IP Fragmentation Unit</td>
</tr>
<tr>
<td>Mac Protocol</td>
</tr>
<tr>
<td>MAC Max Queue Size</td>
</tr>
<tr>
<td>Packet Retry Limit</td>
</tr>
<tr>
<td>Physical Layer Model</td>
</tr>
<tr>
<td>Wireless Frequency Band</td>
</tr>
<tr>
<td>Propagation Limit</td>
</tr>
<tr>
<td>Data Rate</td>
</tr>
<tr>
<td>Mode</td>
</tr>
<tr>
<td>Nodes Tx Power</td>
</tr>
<tr>
<td>Shadowing Model</td>
</tr>
<tr>
<td>Channel Model</td>
</tr>
<tr>
<td>Shadowing Mean</td>
</tr>
<tr>
<td>Receiver Sensitivity</td>
</tr>
<tr>
<td>Path Loss</td>
</tr>
<tr>
<td>Thermal Noise</td>
</tr>
</tbody>
</table>
5.8 Simulation Results

In this section, we have summarized the simulations results and discussed the performance of QLB-OLSR based on them. The results are reported in two sections of 2-Relay and 3-Relay study to be able to separately focus on the performance of QLB-OLSR under these two scenarios. The focus of our simulation result analysis is based on, Throughput Balance, Jitter and End-to-End delay which as it was elaborated in Section 5.4 are the most important QoS factors.

For simplicity, in the graphs provided in this section, the OLSR algorithm is referred to as Baseline and the QLB-OLSR algorithm is shortened to quantum. In this section, the term mean gain is used when referring to the average gain in measurements reported in the box plots. Another type of gain used in our analysis is the so-called stability gain, which is defined as the difference of upper quartile to the lower quartile in a box plot. This is also known as the 50% hinge spread of the data. When load balancing is the subject of our analysis, the mean gain represents the average gain achieved in the fair distribution of the traffic and the stability gain characterizes the deviation of the achieved gain from the mean based on the standard deviation of the results.

5.8.1 2-Relay Simulation Study

In this section, we compare the performance of the baseline OLSR routing algorithm denoted as “Baseline”, with the QLB-OLSR which implements the load balancing algorithm introduced in this chapter. The performance comparison is based on three metrics i.e. Throughput Unbalanced Factor, end-to-end delay and jitter.

5.8.1.1 Throughput Unbalanced Factor (TUF)

To be able to better analyze the balance distribution of throughput over the relay nodes we have defined a metric named Throughput Unbalance Factor (TUF). Throughput is measured at the relay nodes to represent the balance of load distribution among them. The measured throughput is a function of time which then by division of that with the total throughput that is being relayed by all relays we can calculate normalized throughput parametrized as $Th_N(t)$ in Eq. 5.15. $D$ is the distribution factor which, as shown in Eq. 5.16, has an inverse relationship with the number of relay nodes $n$ in our simulation setup. Based on the formula presented in Eq. 5.17, by subtracting $D$ from $Th_N(t)$ we can evaluate $TUF(t)$. Hence, $TUF$ is a measure of how unbalance the traffic load is distributed over the relays with taking the perfect case of load balancing $D$ as reference measure. Hence it can be concluded that a lower value of $TUF$ represents better load balancing and better performance.
Chapter 5: Quantum Load Balancing

\[ T_N(t) = \frac{\text{Throughput}(t)}{\text{Total Throughput}} \quad \text{Eq. 5.15} \]

\[ D = \frac{1}{n} \quad \text{Eq. 5.16} \]

\[ TUF(t) = T_N(t) - D \quad \text{Eq. 5.17} \]

Figure 5-6: Throughput Unbalanced Factor, 2-Relay (box-plot)

The box plot shown in Figure 5-6 compares the TUF performance of the QLB-OLSR with the baseline OLSR routing protocols. Based on the results reported in Figure 5-6, there is an 89\% mean gain reported in the proposed QLB-OLSR algorithm compared to the baseline OLSR.

Figure 5-7: Throughput Unbalanced Factor, 2-Relay (time-plot)

Additionally, the measured stability gain is nearly 90\% compared to the baseline algorithm. Both mean and stability gain shows a consistent load distribution resulted by our proposed algorithm.
5-7 visualizes the same result as Figure 5-6, using a time graph. The graph of TUF versus time also confirms a considerably fair load balancing achieved by the proposed QLB-OLSR compared to the baseline protocol. We can clearly conclude that in the 2-relay scenario, not only we have achieved significant gain in load balancing among the relay nodes but the statistical spread of TUF has improved considerably. It is important to note that the TUF reported here is averaged over the 2 relays.

5.8.1.2 End-to-End Delay

End-to-End delay for both baseline and quantum cases are visualized using the box plot in Figure 5-8. It can be observed that the end-to-end delay performance is nearly the same when comparing the case of quantum with baseline. Given that the number of hops between the source/destination pairs in the simulation study performed in this work is very limited, we do not expect to see any major improvements in terms of end-to-end delay in a the 2-relay scenario. It is only under a large network topology with multiple hops in the routing paths that QLB-OLSR could show performance improvement in terms of end-to-end delay.

![Figure 5-8: End to End Delay, 2-Relay](image)

5.8.1.3 Jitter

Jitter is considered as one of the most important QoS factors. Performance of the baseline protocol has been compared with the quantum case in Figure 5-9. QLB-OLSR has achieved 13% gain compared to the baseline OLSR. Additionally, there is a 14% stability gain achieved when comparing the proposed quantum case with the baseline. It must be noted that a 14% jitter gain in the small topology simulated in this work indicates that the load balancing performed by the QLB-OLSR algorithm can result a significant improvement in large network scenarios.
5.8.2 3-Relay Simulation Study

In this section, the performance of the QLB-OLSR routing algorithm is compared against the baseline OLSR routing algorithm under a 3-relay scenario. Similar to the 2-relay analysis the performance comparison is based on three metrics of TUF, end-to-end delay and jitter.

5.8.2.1 Throughput Unbalanced Factor

The box plot shown in Figure 5-10 compares the performance of QLB-OLSR against the baseline OLSR routing protocol under a 3-relay scenario. Based on our measurement there is a 50% mean gain when comparing the newly proposed algorithm against the baseline. The stability gain of the protocol is measured at 66% which shows great load balancing consistency under a 3-relay scenario as well. Under
this scenario, the theoretically calculated gain based on the quantum game strategies is lower than the 2-relay scenario. Hence it can be concluded that the simulation results are in line with the theoretical analysis on the quantum load balancing performed in Sections 5.3.3 and 5.3.4. The time graph shown in Figure 5-11, visualizes the performance comparison of the QLB-OLSR algorithm against baseline. This graph also confirms the performance gains achieved by the proposed protocol.

![Throughput Unbalance Factor (3-Relay)](image)

*Figure 5-11: Throughput Unbalanced Factor, 3-Relay (time-plot)*

### 5.8.2.2 End-to-End Delay

![End to End Delay (3-Relay)](image)

*Figure 5-12: End to End Delay, 3-Relay*

When we look at the end-to-end delay performance of the QLB-OLSR compared to baseline in Figure 5-12, we can see 21% improvement in the mean delay of the proposed algorithm compared to the baseline. We can also report a significant 29% stability gain in the proposed algorithm. As opposed to the 2-relay scenario where the performance of the proposed algorithm is similar to the baseline, in the
case of the 3-relay, we can observe a significant gain. Due to the lack of any load balancing mechanism in the baseline OLSR, the performance drops significantly under a 3-relay scenario. However, the QLB-OLSR algorithm performs a fair load distribution across all relays and reduces the probability of unbalanced queues at relays. This leads to a lower end-to-end delay as well as more stability in the measured performance. The stability of end-to-end delay has a direct impact on the QoS in the chosen paths hence it would directly impact the quality of routes computed in the network.

5.8.2.3 Jitter

The box plot shown in Figure 5-13 compares the performance of the QLB-OLSR algorithm against the baseline OLSR based on jitter delay. As it was mentioned before, jitter is one of the most important QoS factors in any type of network which affects the quality of services provided in the application layer.

![Figure 5-13: Jitter, 3-Relay](image)

The mean gain in the performance of quantum algorithm was measured at 29% compared to the baseline OLSR. Furthermore, the stability gain is at 26% which proofs that the load balancing algorithm has a better jitter consistency compared to the baseline protocol.

5.9 Summary

This this chapter, we have evaluated performance of quantum game based approach to load balancing. First, we formulated the problem of load balancing in ad hoc networks under the umbrella of quantum game theory. We then showed that the synchronization of entangled particles can be used to affect the decision-making process of distant players without transmission of any information. This enabled us to formulate the problem of load balancing in ad hoc networks using the novel concept of
quantum game theory. Hence the weakness of backpressure algorithm in performing fair load distribution was targeted by the synchronization properties of entangled particles. We also discussed that the behavioural states of a system need to be in violation of the bell inequalities for it to be considered under a quantum paradigm. The larger accessible space of states in a quantum system can be used to maximize a predefined utility function. We formulated the utility function of the quantum games based on the problem of load balancing in ad hoc networks. In order to be able to implement and analyse our theory, we harmonized the problem of load balancing with the quantum strategies resulted by the theoretical analysis of this work. Due to the research direction of this thesis, the proposed theory was implemented in the OLSR routing algorithm as the baseline of the implementation. The so called QLB-OLSR routing protocol was thoroughly analysed based on simulation studies and a significant performance gain was reported. The simulation study in this chapter confirms the expected performance gain in the theoretical analysis of the quantum load balancing. The 2-relay and 3-relay scenarios presented in this work stand as proof of concepts for the proposed theory which shows promise as a solution to the problem of load balancing in ad hoc networks. In a CR-MANETs environment with multiple SOPs, the load balancing can result even more gain due to the added capacity. The work presented in this chapter is to present a novel new perspective to target load balancing and stands as a proof of concept to our proposed theory. To apply QLB algorithm to larger more complex networks, a generalized quantum game theoretical analysis regardless of the number of relay nodes is required, which involves extremely complex quantum state calculations. Expansion of the QLB algorithm to larger network scenarios is considered as the future work of this research.
Chapter 6

6 Conclusion and Future Work

6.1 Conclusion

In Chapter 3 it was elaborated that with the idea of cognitive radio, nodes can utilize SOPs in CR-MANETs in order to improve the network capacity and to provide stable routes that support better QoS requirements. It was further discussed that the routing protocols in network layer can incorporate spectrum-aware route computation strategies to efficiently utilize the SOPs in CR-MANETs. The problem of multi-channel networks was modelled as multi-graphs and GDA was reported as the best candidate to find shortest weighted paths in such graphs. This solution was integrated in OLSR as the baseline protocol and the routing algorithm is so called Spectrum-aware OLSR. The performance gain of spectrum-aware OLSR in terms of PDR was confirmed but this gain comes at the cost of unstable end-to-end delay and higher signalling overhead. In Chapter 4 we proposed a technique to integrate backpressure algorithm into our implementation of spectrum-aware OLSR routing algorithm. Efficient utilization of network resources towards the aim of stable QoS provisioning is one of the main motivations for our interest in proposing load balancing under spectrum-aware routing paradigm. Our implementation of spectrum-aware backpressure OLSR not only balances the network load over the alternative routing paths but also takes SOPs into account in the route computation processes. With the aid of our proposed algorithm, the network load is distributed over all the SOPs via various routing paths in the network. It was confirmed that the load balancing algorithm proposed in Chapter 4 results better end-to-end delay and PDR compared to the baseline OLSR routing protocol as well as the spectrum-aware OLSR. However, this gain comes at the cost of excessive signalling load. The distributed infrastructure-less design of MANETs necessitates a distributed resource allocation mechanism. As it was detailed before, the contention based channel access method of the CSMA, IEEE802.11 has made it one of the most commonly used MAC layer protocols in all simulations involving all kinds of ad hoc networks. As it was discussed before, backpressure algorithm assumes a TDMA channel access method to guarantee an optimal throughput in the network. It was covered that the optimal throughput of backpressure algorithm comes at the cost of higher average end-to-end delay in the routing paths. One of the unrealistic assumptions in the original idea of backpressure algorithm is the instant access of nodes to the neighbouring nodes queue information which is infeasible under the distributed infrastructure-less design of MANETs. Under a realistic MANET environment distribution of any
signalling data including topology, queue information and etc. is subject to communication delays. As backpressure algorithm relies on the accuracy of the queue information and the topology graph, the inconsistency of delays in signalling such information can result unreliability in the computed routes. It was concluded that increasing the frequency of routing protocol’s signalling updates in the network, can minimize this inaccuracy to an acceptable level but to the best of our knowledge it cannot be eliminated. The lack of accuracy and the complexity of synchronization and stabilization of the signalling updates in the network was the motivation for the third contribution of this thesis. In Chapter 5, quantum game theory was firstly analysed on a theoretical level. It was shown that synchronization properties of entangled particles can be used to affect decision making process of distant players in a quantum game setup without the need for transmission of any information. Based on this concept, the problem of load balancing in ad hoc networks was modelled using the novel concept of quantum game theory. It was covered that when behavioural states of a system is in violation of the bell inequalities, it can be considered under a quantum paradigm. This results a larger accessible space of states which can be used to maximize a predefined utility function. The quantum game’s utility function defined in this work was specifically formulated to target load balancing in ad hoc networks and the quantum strategies have been tailored to accommodate these load balancing strategies. Furthermore, the proposed load balancing theory was implemented in OLSR routing algorithm as the baseline of our implementation. Finally, a simulation based performance analysis of QLB-OLSR routing protocol was covered and a significant gain was reported. The simulation studies was concentrated on the two main cases of 2-Relay and 3-Relay analysis which stands as a solid proof of concept for the proposed theory. Quantum load balancing presents a novel new perspective to target load balancing in ad hoc networks and is considered as a proof of concept for our proposed theory.

6.2 Future Work

The future work of this research can be summarized based on the three main contributions provided in this thesis.

- **Spectrum-aware routing:** In this work, we analysed the performance of three routing metrics in OLSR routing protocol as the base of our proposed spectrum-aware routing protocol. However, ETX was the routing metric of choice to perform link quality estimation in our proposed routing algorithm. Routing metrics are normally designed to target a specific utility in the networks and QoS provisioning may require maximization of various performance metrics. Hence, one of the future works of this research is to design the spectrum-aware routing protocol based on a multi-mode QoS provisioning. This can be enabled by classification of the application layer traffics based on their QoS requirement.
Furthermore, a quantitative measurement framework is required to target each class of application layer traffic based on one of the quality estimation metrics. A multi-mode utility based spectrum-aware routing can utilize the SOPs to maximize the quality of routes based on the categories of application layer traffic.

- **Backpressure spectrum-aware algorithm:** As it was detailed before, the second contribution of this work was based on integration of backpressure queue gradients into the route decision making processes. As backpressure algorithm requires full network connectivity graph to perform its load balancing mechanism, a proactive design was used as the baseline of the routing algorithm. However, proactive routing protocols rely on the periodic exchange of signalling messages containing topology, link quality and queue related information. Hence, the accuracy of signalling data depends on the frequency of these updates in the network. As signalling data occupies the available channels and consumes network resources, there is a limit to the level that we can increase the frequency of these updates. We consider inaccuracy of signalling data in backpressure algorithm as an open research problem which is considered to be one of the future works of this research. One of the methods that this inaccuracy can be targeted is by creating a time window (depending on the frequency of signalling updates) during which, when a link is utilized by the application layer traffic, based on the level of load in that traffic, a penalizing mechanism can avoid overutilization of the network resources for the duration of the proposed time window. Theoretically, this mechanism is expected to result load balancing stability in between the proactive signalling updates which does not depend on the queue information and hence is not subject to the limitations of signalling updates.

- **Quantum load balancing:** The quantum load balancing was the last contribution in this thesis which stands as a solid basis for proof of the proposed theory in this research. However, the 2-relay and 3-relay scenarios that were analysed in this work are mainly to proof the concept of quantum load balancing. In order to apply the QLB algorithm to larger more complex networks, a generalized quantum game setup needs to be analysed and theoretically formulated. Generalization of quantum game theory to larger networks is considered as the future work of this research. This involves formulation of the original game strategies based on n-relay nodes which requires complex quantum state calculations. Another approach in targeting this generalization problem is segmenting the network into smaller pieces where the proposed 2-relay and 3-relay load balancing mechanism introduced in this work can be applied.
Bibliography


