# NELS N M NDELA 

UNIVERSITY

# A Congestion Aware Ant Colony <br> Optimisation Based Routing and Wavelength Assignment Algorithm for Transparent Flexi-grid Optical Burst Switched Networks 

Joshua Femi Oladipo

Submitted in fulfilment of the requirements for the degree of Magister Scientiae<br>in the Faculty of Science<br>Nelson Mandela University

December 2018

## Declaration of Own Work


#### Abstract

I, Joshua Femi Oladipo (210036575), declare that the entirety of the work contained in this dissertation is my own original work. That I am the sole author thereof (save to the extent explicitly otherwise stated). That reproduction and publication thereof by Nelson Mandela University will not infringe on any third party rights. And that I have not previously in its entirety or in part submitted it for obtaining any qualification.


Signature:


Date: 11 October 2018

## Acknowledgements

This research would not have been possible without the support of certain key individuals. Firstly, I would like to thank God for keeping and guiding me throughout the duration of this research.

I would like to express my sincere gratitude to my supervisors Dr Mathys du Plessis, and Prof. Timothy Gibbon, for their technical, moral and material support throughout the course of the research. I am truly grateful.

I would like to thank the department of computing sciences for providing the resources required for this research, and for encouragement from various staff members.

I would like to thank Grant Woodford and Jean Rademakers, for assisting me with the technical hassles.

I would like to thank my family and friends for their support and understanding.
Finally, I would like to thank George Mujuru, for his words of encouragement.

## Abstract

Optical Burst Switching (OBS) over transparent flexi-grid optical networks, is considered a potential solution to the increasing pressure on backbone networks due to the increase in internet use and widespread adoption of various high bandwidth applications. Both technologies allow for more efficient usage of a networks resources. However, transmissions over flexi-grid networks are more susceptible to optical impairments than transmissions made over fixed-grid networks, and OBS suffers from high burst loss due to contention. These issues need to be solved in order to reap the full benefits of both technologies. An open issue for OBS whose solution would mitigate both issues is the Routing and Wavelength Assignment (RWA) algorithm. Ant Colony Optimisation (ACO) is a method of interest for solving the RWA problem on OBS networks. This study aims to improve on current dynamic ACO-based solutions to the Routing and Wavelength Assignment problem on transparent flexi-grid Optical Burst Switched networks.

In order to pursue the objective stated in the previous paragraph, an OBS simulator, which is capable of simulating the operation of OBS on transparent flexi-grid optical networks, was built and validated. A literature review of ACO based solution to the RWA problem on OBS networks, lead to the selection of two algorithms (ACOR and FSAC) as potential candidates for further development. A performance evaluation of both lead to the selection of FSAC as the algorithm of choice for future development. Based on the results of the performance evaluation, weak points of the FSAC algorithm were identified. Incorporating congestion information into the FSAC was identified as an option which might remedy the weak points of the algorithm. The modifications made to FSAC, was found to improve the performance of FSAC, and hence was given the name CM-FSAC.

## Contents

Declaration of Own Work ..... i
Acknowledgements ..... ii
Abstract ..... iii
Table of Contents ..... vii
List of Figures ..... ix
List of Tables ..... xi
List of Abbreviations ..... xii
1 Introduction ..... 1
1.1 Background ..... 1
1.2 Research Objective ..... 3
1.3 Context of Study ..... 4
1.4 Methodology ..... 5
1.5 Dissertation Structure ..... 7
2 Literature Review ..... 9
2.1 Introduction ..... 9
2.2 Wavelength Division Multiplexing ..... 9
2.2.1 Flexi-grid ..... 10
2.2.2 Transparent WDM Networks ..... 12
2.2.3 Enabling technologies ..... 14
2.2.4 Optical Impairments ..... 15
2.3 Optical Burst Switching ..... 18
2.3.1 Optical Switching ..... 19
2.3.2 Burst Assembly ..... 20
2.3.3 Routing and Wavelength Assignment ..... 20
2.3.4 Bandwidth Reservation and Switching ..... 23
2.4 Simulation Principles ..... 25
2.4.1 Discrete Event Simulation ..... 26
2.4.2 Simulator Frameworks ..... 27
2.4.3 Reduced Link Load Model ..... 28
2.5 Ant Colony Optimisation ..... 30
2.6 Ant Colony Optimisation RWA Algorithms ..... 32
2.6.1 ACOR ..... 32
2.6.2 DABR ..... 34
2.6.3 ACRWA ..... 36
2.6.4 UCBRWA ..... 38
2.6.5 FSAC ..... 40
2.6.6 Discussion ..... 41
2.7 Conclusion ..... 43
3 Simulator Design \& Validation ..... 44
3.1 Introduction ..... 44
3.2 Motivation ..... 44
3.3 Problem formulation ..... 45
3.4 Selection of Simulator Framework ..... 46
3.5 Omnet++ Architecture ..... 46
3.6 Simulator Design ..... 47
3.7 Simulator Implementation ..... 49
3.7.1 Edge Node ..... 50
3.7.2 Core Node ..... 52
3.7.3 Fibre ..... 52
3.7.4 Super Node ..... 53
3.8 OBS Validation ..... 55
3.8.1 Validation Procedure ..... 55
3.8.2 Validation Results ..... 56
3.9 Effects of Flexi-grid and Impairments ..... 57
3.9.1 Experimental Procedure ..... 57
3.9.2 Results and Analysis ..... 58
3.10 Conclusion ..... 60
4 Initial Investigations ..... 61
4.1 Introduction ..... 61
4.2 Adaptation \& Implementation of Algorithms ..... 62
4.3 Network Scenarios ..... 63
4.4 Parameter Study ..... 64
4.4.1 Experimental Procedure ..... 64
4.4.2 Results and Discussion ..... 66
4.5 Performance Comparisons ..... 73
4.5.1 Experimental Procedure ..... 73
4.5.2 Parameter Selection ..... 74
4.5.3 Overall Performance Comparison ..... 76
4.5.4 Timewise Performance Comparison ..... 78
4.6 Discussion ..... 82
4.7 Conclusion ..... 82
5 Proposed Improvements ..... 84
5.1 Introduction ..... 84
5.2 Incorporating Route Congestion Information ..... 84
5.2.1 Congestion Estimation ..... 85
5.2.2 Incorporating Congestion ..... 86
5.2.3 Other Modifications ..... 87
5.3 Information Deprived FSAC ..... 87
5.4 Parameter Study ..... 88
5.4.1 CM1 ..... 89
5.4.2 CM2 ..... 91
5.4.3 CM3 ..... 92
5.4.4 ID-FSAC ..... 92
5.5 Performance Comparison ..... 95
5.5.1 Parameter Selection ..... 95
5.5.2 Overall Performance Comparison ..... 96
5.5.3 Timewise Performance Comparison ..... 101
5.6 Conclusion ..... 104
6 Conclusion ..... 105
6.1 Introduction ..... 105
6.2 Research Contributions ..... 105
6.2.1 Development of a Flexi-grid OBS Simulator ..... 106
6.2.2 Improved ACO RWA Solution ..... 107
6.3 Limitations ..... 108
6.4 Recommendation for Future Investigations ..... 108
6.5 Summary ..... 109
A Results ..... 116
B Statistical Methods ..... 123
B. 1 Pearson Correlation Coefficient ..... 123
B. 2 Mann-Whitney U Test ..... 123
C CM-FSAC Pseudo-Code ..... 124
D Scatter Plots of Data Obtained in Parameter Studies ..... 127
D. 1 NSFNET ..... 127
D.1.1 High ..... 127
D.1.2 Medium ..... 130
D.1.3 Low ..... 133
D. 2 SIMPLE ..... 137
D.2.1 High ..... 137
D.2.2 Medium ..... 140
D.2.3 Low ..... 143
E IFIP PEMWN 2017 ..... 147

## List of Figures

1.1 Steps in conducting a simulation study (Law, 2003) ..... 5
2.1 Illustration of fixed and flexi-grid (Miroslaw Klinkowski et al., 2013) ..... 12
2.2 The wavelength continuity constraint on a path that consist of two links. Used wavelengths are indicated in colour. (Shen \& Yang, 2011) ..... 14
2.3 Schematic view of various optical switching elements ..... 16
2.4 Illustration of the operation of Liquid Crystal on Silicon (LCoS) (López \& Velasco, 2016) ..... 17
2.5 Elastic wavelength assignment (Miroslaw Klinkowski et al., 2013) ..... 24
2.6 Network timing diagrams, showing the operation of JIT and JET bandwidth reservation schemes. (Teng \& Rouskas, 2003) ..... 25
2.7 Illustration of the shortest path finding ability of ant colonies (Blum, 2005) ..... 31
3.1 Simple and compound modules ("OMNET++ Simulation Manual", 2018) ..... 46
3.2 Usage of simulator modules in building a network ..... 49
3.3 Edge Node components ..... 50
3.4 Core Node module layout ..... 51
3.5 Fibre module layout ..... 52
3.6 SuperNode module layout ..... 54
3.7 Network built using super nodes ..... 54
3.8 Topology used in validation, with routes used in validation ..... 55
3.9 Graphs showing the average burst loss probability vs load, for the simulations and approximations ..... 56
3.10 NSFNET Topology (Gravett, du Plessis, \& Gibbon, 2017). Link lengths in km. ..... 58
3.11 Graphs showing the average burst loss probability vs load, for various fixed and flexi-grid scenarios ..... 60
4.1 SIMPLE Topology (Gravett, du Plessis, \& Gibbon, 2017). Link lengths in km. ..... 63
4.2 Graphs showing the average Burst Loss Probability vs Spectrum Width, for various algorithms on the SIMPLE topology at various loads ..... 77
4.3 Graphs showing the average Burst Loss Probability vs Spectrum Width, for various algorithms on the NSFNET topology at various loads ..... 77
4.4 Time wise performance of algorithms on SIMPLE at various loads with 100 GHz spectrum width. ..... 80
4.5 Time wise performance of algorithms on SIMPLE at various loads with 200 GHz spectrum width. ..... 80
4.6 Time wise performance of algorithms on SIMPLE at various loads with 300 GHz spectrum width. ..... 80
4.7 Time wise performance of algorithms on NSFNET at various loads with 200 GHz spectrum width. ..... 81
4.8 Time wise performance of algorithms on NSFNET at various loads with 300 GHz spectrum width. ..... 81
4.9 Time wise performance of algorithms on NSFNET at various loads with 400 GHz spectrum width. ..... 81
5.1 Graphs showing the average burst loss probability vs Spectrum Width, for various algorithms on the SIMPLE topology at various loads. ..... 99
5.2 Graphs showing the average burst loss probability vs Spectrum Width, for various algorithms on the NSFNET topology at various loads. ..... 99
5.3 Time wise performance of algorithms on SIMPLE at various loads with 100 GHz spectrum width. ..... 102
5.4 Time wise performance of algorithms on SIMPLE at various loads with 200 GHz spectrum width. ..... 102
5.5 Time wise performance of algorithms on SIMPLE at various loads with 300 GHz spectrum width. ..... 102
5.6 Time wise performance of algorithms on NSFNET at various loads with 200 GHz spectrum width. ..... 103
5.7 Time wise performance of algorithms on NSFNET at various loads with 300 GHz spectrum width. ..... 103
5.8 Time wise performance of algorithms on NSFNET at various loads with 400 GHz spectrum width. ..... 103

## List of Tables

3.1 Burst Loss Probabilities obtained for various scenarios ..... 59
4.1 Parameter values used in parameter study of ACOR ..... 65
4.2 Parameter values used in parameter study of FSAC ..... 65
4.3 Mean inter-arrival time values used to achieve various loads in pa- rameter studies ..... 66
4.4 Correlation coefficients for ACOR at various simulation times on the SIMPLE topology. Significant values given in bold. ..... 68
4.5 Correlation coefficients for ACOR at various simulation times on the NSFNET topology. Significant values given in bold. ..... 68
4.6 Correlation coefficients for FSAC at various simulation times on the SIMPLE topology. Significant values given in bold. ..... 72
4.7 Correlation coefficients for FSAC at various simulation times on the NSFNET topology. Significant values given in bold ..... 72
4.8 Mean inter-arrival time values used to achieve various loads in per- formance comparisons ..... 74
4.9 Parameter values used for ACOR in performance comparisons ..... 75
4.10 Parameter values used for FSAC in performance comparisons ..... 75
4.11 Performance comparison of ACOR, FSAC and GSPR in cases with statistically significant results ..... 78
5.1 Parameter values used in parameter study of CM1, CM2 and CM3 ..... 89
5.2 Parameter values used in parameter study of ID-FSAC ..... 89
5.3 Correlation coefficients for CM1 at various simulation times on the SIMPLE topology. Significant values given in bold. ..... 90
5.4 Correlation coefficients for CM1 at various simulation times on the NSFNET topology. Significant values given in bold. ..... 90
5.5 Correlation coefficients for CM2 at various simulation times on the SIMPLE topology. Significant values given in bold. ..... 90
5.6 Correlation coefficients for CM2 at various simulation times on the NSFNET topology. Significant values given in bold. ..... 93
5.7 Correlation coefficients for CM3 at various simulation times on the SIMPLE topology. Significant values given in bold. ..... 93
5.8 Correlation coefficients for CM3 at various simulation times on the NSFNET topology. Significant values given in bold ..... 93
5.9 Correlation coefficients for ID-FSAC at various simulation times on the SIMPLE topology. Significant values given in bold. ..... 94
5.10 Correlation coefficients for ID-FSAC at various simulation times on the NSFNET topology. Significant values given in bold. ..... 94
5.11 Parameter values used for CM1, CM2 and CM3 in performance com- parisons ..... 96
5.12 Parameter values used for ID-FSAC in performance comparisons ..... 97
5.13 Performance comparison of CM1, CM2, CM3, FSAC, GSPR, and ID-FSAC, in cases with statistically significant results. ..... 100

## List of Abbreviations

| Abbreviation | Term |
| :---: | :---: |
| ACO | Ant Colony Optimisation |
| BVT | Bandwidth Variable Transponder |
| BLP | Burst Loss Probability |
| CWDM | Coarse Wavelength Division Multiplexing |
| DWDM | Dense Wavelength Division Multiplexing |
| JET | Just Enough Time |
| JIT | Just-In-Time |
| LCoS | Liquid Crystal on Silicon |
| OBS | Optical Burst Switching |
| OCS | Optical Circuit Switching |
| OEO | Optical-Electrical-Optical |
| OPS | Optical Packet Switching |
| OXC | Optical Cross-Connect |
| RAM | Random Access Memory |
| ROADM | Reconfigurable Optical Add/Drop Multiplexer |
| RWA | Routing and Wavelength Assignment |
| TTL | Time-To-Live |
| WC | Wavelength Continuity |
| WDM | Wavelength Division Multiplexing |
| WSS | Wavelength Selective Switch |

## Chapter 1

## Introduction

### 1.1 Background

An optical network is a communication network which uses light signals to transmit data over glass fibres (called optical fibres). The potential bandwidth that could be accessed over optical fibre vastly outstrips that available over any other communication medium currently in use; hence, they form a major part of telecommunication backbone networks in the form of Wide Area Networks (WANs) and Metropolitan Area Networks (MANs) (Maier, 2008; Ramaswami, Sivarajan, \& Sasaki, 2010). Optical fibre have also been increasingly deployed in Access Networks, where optical fibres are used up to the connections to homes and offices (Maier, 2008; Ramaswami et al., 2010; López \& Velasco, 2016).

Due to various technological limitations, the full potential of optical fibres are not currently being exploited. On the other hand, demand for network bandwidth is steadily and quickly increasing due to the increase in Internet access over fixed and mobile channels as well as the widespread adoption of various high bandwidth applications by end-users such as video streaming, cloud storage and cloud computing (López \& Velasco, 2016; Jinno et al., 2009). Network traffic patterns are also becoming more dynamic in time and direction; that is large changes in traffic magnitude as well as flow direction occur over the span of a day (López \& Velasco, 2016). These issues call for an increase in network capacity and flexibility.

Wavelength Division Multiplexing (WDM) is a method of increasing transmission capacity of an optical fibre by allowing the parallel transmission of multiple streams of data over the fibre (where each stream is sent on a different wavelength) (Maier, 2008; Ramaswami et al., 2010; Simmons, 2014). The two variations of WDM are

Coarse Wavelength Division Multiplexing (CWDM), where a maximum of 16 widely spaced wavelengths might be used; and Dense Wavelength Division Multiplexing (DWDM) where over 100 closely spaced wavelengths could be used (Stamatis V. Kartalopoulos, 2004; Transmode, 2013). The wavelengths which should be supported by equipment that supports DWDM are standardised by the International Telecommunication Union (ITU) in Recommendation G.694.1 (International Telecommunication Union - ITU-T, 2012); The document defines a fixed grid scheme of various bandwidth spacing; as well as a flexible grid scheme (also known as flex spectrum) which allows the use of arbitrarily positioned and sized bandwidth channels, made up of smaller concatenated channels (whose size are multiples of 12.5 GHz ). The flexible grid scheme allows for more efficient usage of the bandwidth available over a fibre, however, it comes with a penalty in terms of increased inter-channel crosstalk due to the reduced guard bands between transmission channels on the fibre.

The use of Wavelength Division Multiplexing (WDM) and it's corresponding optical transmission and switching technologies (such as Bandwidth Variable Transponders (BVT's), Reconfigurable Optical Add Drop Multiplexers (ROADM's) and Optical Cross Connects (OXC's)) allow for reconfigurable optical networks; hence improving the flexibility of these networks. Researchers are presented with the opportunity to develop advanced control plane algorithms to derive maximum benefit from current optical networks, hence, reducing the need for capital expenditure spent to improve network capacity and performance. The availability of the above mentioned technologies (BVT's, ROADM's and OXC's) lead to the proposal of new optical switching technologies which such as Optical Burst Switching and Optical Packet Switching which would improve the efficiency of optical networks.

Current optical networks use a switching paradigm known as Optical Circuit Switching (OCS), where all the bandwidth on a channel (where a channel could be a fibre, a wavelength, or a time slot in a Time Division Multiplexed (TDM) system) is fully dedicated to transmission between a pair of nodes (Maier, 2008; Chen, Qiao, \& Yu, 2004). The channels may be assigned statically or as required for a given amount of time. The disadvantage of this is that for bursty (irregular) data the bandwidth of a channel is reserved for a given node irrespective of if the node has data to transmit. A node might not need to send data or might only need to send data across a channel for a short period of time, leaving the remaining channel slot unused; while another application might currently need the bandwidth. This is inefficient. A better way would be to use Optical Packet Switching (OPS), where packets sent over a given wavelength channel are independently switched towards their destination (Maier \& Reisslein, 2008; Rouskas \& Xu, 2004). However, true OPS is not currently possible
due to the absence of cost effective optical buffers, which would allow the buffering of packets to enable contention resolution at core nodes, as well as efficient methods for processing each packet within the optical domain similar to what occurs on electronic packet switched networks (Maier \& Reisslein, 2008; Rouskas \& Xu, 2004).

Optical Burst Switching (OBS) could be considered a compromise to improve the efficiency of optical networks using currently available technology (Chen et al., 2004; Bjornstad et al., 2003; Verma, Chaskar, \& Ravikanth, 2000). In an OBS network, packets are buffered and assembled into bursts at network edge nodes according to various parameters. When a burst is ready to be sent, the node sends a packet known as a Burst Control Packet (BCP), on a reserved channel called the control channel, in order to reserve the required bandwidth and configure the switches at intermediate nodes. The burst is then sent after a certain offset time, in order to give the BCP time to finish setting up the switches at the intermediate nodes. The reserved bandwidth for a transmission is released after the burst has traversed the network, in order to allow its use by another burst.

A major issue that needs to be solved in OBS networks is the Routing and Wavelength Assignment (RWA) problem. RWA deals with the routing and assignment of wavelengths in order to ensure that the current request is fulfilled, the Burst Loss Probability (BLP) of future requests is minimised and ensure that available network resources are used in an efficient manner (Gravett, du Plessis, \& Gibbon, 2017). The RWA problem consist of two parts: computing a path from source to destination and then assigning a wavelength to the selected path. The RWA problem could be solved in the presence of the Wavelength Continuity (WC) constraint, where a single wavelength must be used to traverse the path from the source to the destination node or without it (Simmons, 2014; Rouskas, 1999). Enforcing the WC constraint offers the benefit of avoiding the costly optical to electrical conversion (and possibly storage) which would be required at each node to deal with any wavelength contention that might occur.

### 1.2 Research Objective

Currently used technologies in optical networks are not sufficient to deal with the increasing demand for bandwidth and the new dynamic dimensions of this demand. Hence there is a need for new methods of data transmission in order to exploit improved technology and handle the demand. As presented in Section 1.1, OBS on
flexi-grid networks is the most viable technology for the foreseeable future. However, both technologies have issues that require solutions to manage them; that is, increased transmission impairments on flexi-grid and the Routing and wavelength assignment (RWA) problem for OBS.

A method of interest for solving the RWA problem is Ant Colony Optimisation (ACO). ACO is a set of algorithms that aim to emulate the emergent behaviour of ants foraging for food. It has been applied to various graph-theoretic problems, and is of interest due to its ability to dynamically route data while improving network performance, and to continuously adapt to changes on the network (such as changes in the number of available wavelengths, removal or addition of a node and increase or decrease in the amount of traffic). Hence the objective of this research is to improve on current dynamic ACO-based solutions to the Routing and Wavelength Assignment problem on flexi-grid optical burst switching networks. The improved solution should reduce the Burst Loss Probability (BLP) and effectively utilise and manage spectrum resources while also considering optical impairments, under the wavelength continuity constraint.

Pursuing the objective stated above will require the use of a discrete event simulator which simulates the operation of OBS over flexi-grid networks, while incorporating impairment modelling. However, in the course of preliminary research, no simulator which could fulfil the needs of the research could be found. Hence, a secondary objective of this research is to design and implement a flexible simulator which is capable of meeting the needs of OBS researchers.

### 1.3 Context of Study

This research will be conducted within the Centre for Broadband Communication (CBC) and falls under the "Next generation Dense Wavelength Division Multiplexing (DWDM) systems research" focus area of the centre ${ }^{1}$. The CBC was created with the support of the Department of Science and Technology (DST), the National Research Foundation (NRF), the Council for Scientific and Industrial Research (CSIR), the Square Kilometre Array (SKA) and Cisco, and is tasked with developing resources necessary to ensure that all South Africans have access to the Internet by 2020 .

[^0]

Figure 1.1: Steps in conducting a simulation study (Law, 2003)

### 1.4 Methodology

Theory related to WDM (focussing on flexi-grid, transparent networks, enabling technologies, and optical impairments), OBS, simulation and ACO are presented in Chapter 2. A literature review of existing ACO RWA algorithms for OBS networks is also performed in Chapter 2, in order to gain an understanding of the ways ACO has been applied to OBS networks.

The research will be performed using the seven step approach for successfully conducting a simulation study prescribed by Law (2003). The steps are presented in Figure 1.1 and explained below

1. Formulate the problem - determine the objectives of the study, the scope
of the model, and the performance measures that need to be collected.
2. Collect information and construct conceptual model - collect information from real systems if possible; for use in constructing a conceptual model and validating the final system. The documentation of the conceptual model should include a detailed description of the operations of various subsystems as well as how they interact and the various assumptions made.
3. Check for conceptual model validity - discuss the conceptual model with stakeholders in a structured manner to determine any errors and omissions, and ensure that any assumptions made are correct.
4. Implement the model - implement the conceptual model either from scratch or using an appropriate simulation tool. Verify the program by debugging.
5. Check for implementation validity - compare the results obtained from the tool against results from real world systems if available, or analytical results. Results should be scrutinised to ensure they are reasonable, and analysis should be performed to determine the extent to which various factors affect the obtained results, to allow for careful implementation of the more influential factors.
6. Design, conduct and analyse experiments - determine configurations of interest and related factors such as the run length of experiments, and number of times to replicate them. Analyse the results to determine if more experiments are needed.
7. Document and present the simulation results - document conceptual models and implementation details as well as results from experiments. Presentation of results should also clearly state their limitations.

Consequently, chapter 3 details the design, implementation and validation of an OBS simulator using a selected simulation frame work, hence, fulfilling steps one to five of the seven step approach.

Chapters 4 and 5 present the experimental design as well as results obtained from the experiments towards the main objective of this research as specified in Section 1.2. In Chapter 4, Two ACO RWA algorithms, which might undergo further development were selected from those previously discussed in Chapter 2. Parameter studies and performance evaluations were performed on both algorithms, in order to determine which performs best, and hence, which might be selected for further development. The algorithms were evaluated on two topologies with three load conditions. In Chapter 5, modifications which might improve the operation of the algorithm were
proposed and implemented based on observations made about the best performing algorithm in the performance evaluations of Chapter 4. The proposed modifications were then evaluated against each other and the original algorithm, to determine, if an improvement had been achieved. Chapters 4 and 5 fulfil steps six and seven of the seven step approach.

### 1.5 Dissertation Structure

An outline of the contents of the various chapters of this dissertation is given below

## Chapter 2: Literature Review

Chapter 2 presents the applicable background theory of optical networks, OBS, simulator development and ACO. Various applications of ACO to RWA on OBS networks are reviewed.

## Chapter 3 : Simulation Design \& Validation

Chapter 3 details the process of designing, implementing and validating the simulator that was used in this research.

## Chapter 4 : Initial Investigations

Chapter 4 details the initial evaluations of select ACO algorithms which were reviewed in Chapter 2. They were evaluated in order to determine their performance against each other on flexi-grid OBS networks, and what improvements may be made to them. The algorithm which performs best in the evaluations is selected for future development.

## Chapter 5 : Proposed Changes

In Chapter 5, various modifications which might improve the performance of the algorithm selected in Chapter 4 are made. The modified algorithm is evaluated against the original, in order to determine if there is a performance improvement.

## Chapter 6 : Conclusions

In chapter 6, an overview of the research will be given. The contributions and limitations of this work will be stated. Finally, recommendations for future research are also proposed.

## Appendix A: Results

Appendix A presents the comparison results collected for the various algorithms in tabular form.

## Appendix B : Statistical methods

Appendix B presents a brief overview of the two major statistical techniques used in the course of the research.

## Appendix C : CM-FSAC Pseudo-code

Appendix C presents the pseudo-code of the CM-FSAC algorithm generated by this research.

Appendix D : Scatter Plots of Data Obtained in Parameter Studies
Appendix D presents scatter plots of data obtained during parameter studies.

## Appendix E : IFIP PEMWN 2017

Appendix E presents the conference paper that was accepted and presented at the 6th IFIP International Conference on Performance Evaluation and Modeling in Wired and Wireless Networks.

## Chapter 2

## Literature Review

### 2.1 Introduction

In this chapter, various aspects of the research are discussed. Wavelength Division Multiplexing is discussed in Section 2.2; focus is placed on its evolution, enabling technologies, and the various optical impairments which may occur. Next, the operation of Optical Burst Switching is described in Section 2.3; paying particular attention to the burst assembly, routing and wavelength assignment and switching aspects. Section 2.4 presents theory relating to simulations and an introduction to Discrete Event Simulation (Section 2.4.1). It was decided that a simulator framework would be used as opposed to building a simulator from scratch; in order to increase the speed of development, and obtain the credibility, usability and extensibility advantages which may be obtained from a framework. Hence, a survey of potential simulator frameworks which might be used in building the simulator required for the research is given (Section 2.4.2); followed by an introduction to the method which will be used to validate the operation of the built simulator (Section 2.4.3). An introduction to Ant Colony Optimisation is given in Section 2.5. And lastly, a survey of Ant Colony Optimisation methods applied to Optical Burst Switched networks is performed in Section 2.6.

### 2.2 Wavelength Division Multiplexing

Wavelength Division Multiplexing (WDM) is a method of increasing the transmission capacity of an optical fibre, by allowing the parallel transmission of multiple streams of data, with potentially different transmission speeds over different wave-
lengths across an optical fibre (Maier, 2008; Ramaswami et al., 2010; Simmons, 2014). This vastly increases the transmission capacity of optical networks.

In this section, a discussion of flexi-grid and transparent WDM networks is given. Followed by a brief overview of their enabling technologies and the optical impairments which may occur over optical networks.

### 2.2.1 Flexi-grid

WDM systems may adhere to either the Coarse Wavelength Division Multiplexing (CWDM) standard or Dense Wavelength Division Multiplexing (DWDM) standard of the ITU-T (Stamatios V. Kartalopoulos, 1999; Transmode, 2013). The standards prescribe what wavelengths WDM hardware should support, in order to foster interoperability of hardware by different manufacturers. CWDM and DWDM are both fixed grid schemes; that is, the frequency spacing between the central frequency of adjacent wavelengths is kept constant (this spacing is called the channel spacing). CWDM is the earlier standard(International Telecommunication Union - ITU-T, 2003). CWDM requires that the wavelengths are widely spaced in order to reduce the impact of impairments on the transmissions; this was necessary due to the expense and immaturity of transponder and fibre technology at the time. Hence CWDM systems may only support a maximum of 18 wavelengths. As technology improved, the wavelengths could be moved closer together leading to DWDM (illustrated in Figure 2.1a) which allows for equipment that supports $12.5 \mathrm{GHz}, 25 \mathrm{Ghz}, 50 \mathrm{Ghz}$ and 100 GHz spacings (Stamatios V. Kartalopoulos, 1999; International Telecommunication Union - ITU-T, 2012). However, in practice 50 GHz spacing (leading to networks with over a 100 wavelengths) has been favoured, as it allows for the transmission of signals with a transmission rate of $10-100 \mathrm{~Gb} / \mathrm{s}$. However, the fixed grid spectrum is considered wasteful, due to the fact that transmissions often do not require the whole wavelength, leading to the presence of large guard bands (unused spectrum) (Wright, Lord, \& Velasco, 2013; Xia, Gringeri, \& Tomizawa, 2012). For example, a $10 \mathrm{~Gb} / \mathrm{s}$ transmission requires only 15 GHz of spectrum, however, it has to be assigned the whole 50 GHz (Xia et al., 2012).

Due to the increasing pressure on data networks, network operators are seeking ways of increasing their transmission capacity (Wright et al., 2013; Gerstel, Jinno, Lord, \& Yoo, 2012; Xia et al., 2012). Hence, network operators are deploying higher transmission rates of $400 \mathrm{~Gb} / \mathrm{s}$ and higher, to handle high-bandwidth applications (Wright et al., 2013; Gerstel et al., 2012; Xia et al., 2012). However, the ITU-T fixed grid does not allow efficient transmission of high bit rate signals, as they do not fit
into the prescribed 50 GHz spectrum. For example, in order to transmit a $400 \mathrm{~Gb} / \mathrm{s}$ signal on a fixed grid, the signal could be split into four $100 \mathrm{~Gb} / \mathrm{s}$ signals, which would fit into 50 GHz channels; however, this would use up 125 GHz of spectrum more than if the signal had been transmitted using a single $400 \mathrm{~Gb} / \mathrm{s}$ signal (Wright et al., 2013). In order to be cost-effective, optical networks need to be multi-line rate networks, due to diverse nature of demands (diverse transmission rates, and hence, spectrum requirement) they have to serve and the trade-off between transmission rate and reach (Nag, Tornatore, \& Mukherjee, 2010; Amaya et al., 2011; Wright et al., 2013). The trade-off between transmission rate and reach the distance a signal will travel without having to re-generated is dependent on its transmission rate; higher transmission rate signals are capable of travelling less distance than lower transmission rate signals (Nag et al., 2010; Amaya et al., 2011; Wright et al., 2013). It is important to limit the amount of signal re-generation performed due to the equipment cost and to minimise energy consumption of the network (Nag et al., 2010; Amaya et al., 2011; Xia et al., 2012).

Flexi-grid was specified in (International Telecommunication Union - ITU-T, 2012), in order to cater for the issues presented in the previous paragraphs (wastefulness in current networks due to large guard bands, deployment of increased transmission rates, and need to support multi-line rate networks). Flexi-grid finely specifies the wavelengths and allows the concatenation of adjacent wavelengths to make bigger channels (International Telecommunication Union - ITU-T, 2012). Figure 2.1 illustrates the structure of the bandwidth over a link, on a fixed grid (Figure 2.1a) and flexi-grid (Figure 2.1b) network. Potential central frequencies are spaced out at 6.25 GHz ; however, the channel spacing (channel segment in Figure 2.1b) is 12.5 GHz . This difference in central frequency spacing and channel spacing is in order to allow the placing of channels whose width is an even multiple of 12.5 GHz , next to one whose width is an odd multiple of 12.5 GHz , without a gap. Any number of adjacent channels can be assigned to a transmission as long as they do not overlap with another already assigned channel. The number of channels required for a transmission is dependent on the transmission speed and modulation format.

Flexi-grid leads to more efficient usage of the available bandwidth, due to the fact that transmissions with large bandwidth requirements can grow as much, to use whatever they need (faster transmissions require more spectrum). While transmissions with small spectrum requirements only use the minimum spectrum they require (Wright et al., 2013; Gerstel et al., 2012). Flexi-grid also makes the network more flexible and responsive to changes in the network hardware, evolving demands and improvements in technology (Wright et al., 2013; Gerstel et al., 2012). The

Rigid optical channel with implicitly assigned fixed 50 GHz spectrum width

(a) Fixed DWDM Grid

(b) Flexi-grid

Figure 2.1: Illustration of fixed and flexi-grid (Miroslaw Klinkowski et al., 2013)
channel spacings of flexi-grid also allow it to be compatible with DWDM systems; hence easing the adoption process. However, due to the lack of large guard bands as are present in fixed grid systems, the impact of impairments (particularly interchannel cross-talk) which might prevent signals from reaching their destination is more prominent than in flexi-grid systems (Wright et al., 2013; Gerstel et al., 2012).

Flexi-grid is an example of flex-spectrum transmission. Where flex-spectrum is a generic term used in the literature for optical transmission systems where the spectrum may be assigned arbitrarily with or without the guidance of an ITU-T grid. Another type of flex-spectrum is grid-less (Shen \& Yang, 2011), where arbitrarily sized and positioned spectrum can be assigned to transmissions on the network. Grid-less has all the advantages and disadvantages of flexi-grid, except that it isn't supported by the ITU-T and is not compatible with the DWDM systems.

### 2.2.2 Transparent WDM Networks

WDM networks can be classified as opaque, transparent or translucent (Maier, 2008; Saleh \& Simmons, 2012; Simmons, 2014). In opaque networks, Optical-ElectricalOptical (OEO) conversion occurs at each node; all incoming optical wavelength channels are converted to electrical signals before being retransmitted across another link if they have not arrived at their destination. This allows for wavelength
conversion at intermediate nodes (that is switching a transmission from its current wavelength to a free one, if it current wavelength is being occupied by another transmission) and retransmission at intermediate nodes, hence improving the reach of transmissions. On the other hand, in transparent networks (also called All-Optical Networks), intermediate nodes on the way to the destination can be optically bypassed by optical wavelengths which have not arrived at their destination, while transmissions terminating at that node are converted to electrical signals. Thus a transmission between two nodes can traverse the network without undergoing OEO transmission. Translucent networks are simply WDM networks which have a mix of transparent and opaque nodes.

Transparent networks are more scalable and reliable than opaque networks due to large reduction in required electronics (particularly transponders) yielding reduced cost, space, heat dissipation and power requirements (Maier, 2008; Saleh \& Simmons, 2012; Simmons, 2014; Jinno et al., 2009). They are also much easier to extend, as extending a node only requires the addition of more optical transponders (Maier, 2008; Saleh \& Simmons, 2012; Simmons, 2014; Jinno et al., 2009). Finally, and most importantly, they are agnostic to the wavelength, modulation scheme, line rate and protocol being used; that is in an opaque network every node needs to support all transmission line rates and modulation format (Maier, 2008; Saleh \& Simmons, 2012; Simmons, 2014; Jinno et al., 2009). However, in a transparent optical network, only the source and destination nodes need to support the formats being used for transmission. All the above advantages make transparent networks an attractive proposal to network operators, as they are more flexible and cheaper to operate than opaque networks (Maier, 2008; Saleh \& Simmons, 2012; Simmons, 2014; Jinno et al., 2009).

Disadvantages of transparent networks are: transmissions are not regenerated due to absence of OEO conversion at each node, hence increasing the effects of impairments on transmissions (potentially reducing the reach of transmissions) and reducing the performance monitoring ability of the network (Maier, 2008; Saleh \& Simmons, 2012; Simmons, 2014). An incoming transmission into a network would tie up a wavelength across multiple links, preventing the wavelength from being used by subsequent transmissions which arrive within the duration of the first transmission and have intersecting paths on the network. This is called the Wavelength Continuity (WC) constraint (Maier, 2008; Saleh \& Simmons, 2012; Simmons, 2014). In Figure 2.2 , the WC constraint on a path with two links is illustrated. It can be observed that the actual amount of spectrum which may be assigned to a transmission which would travel across Link 1 and Link 2 (indicated as "Lightpath") is less than the


Figure 2.2: The wavelength continuity constraint on a path that consist of two links. Used wavelengths are indicated in colour. (Shen \& Yang, 2011)
free amount of spectrum on Link 1 and Link 2; the spectrum that may be assigned to a new transmission, excludes those already assigned to other transmissions on either of the links.

The wavelength continuity constraint has the implication that transmissions might be lost even though there are enough wavelengths available on the required links; due to the necessary wavelength being occupied along the route of the transmission. In flexi-grid networks, the WC constraint also leads to fragmentation, where due to the mixture of transmissions with various transmission speeds on the network (faster transmissions require more spectrum), free spectrum along a route can become isolated, hence preventing the assignment of such spectrum to new transmissions due to the isolated wavelengths not being large enough to service new requests (Wright et al., 2013; Simmons, 2014).

The wavelength continuity constraint can only be eliminated by wavelength conversion of blocked transmissions at intermediate nodes (Maier, 2008; Saleh \& Simmons, 2012; Simmons, 2014). Sadly, all-optical wavelength converters are still immature and expensive technology, hence the only practical way of achieving this is by introducing OEO conversion at intermediate nodes to service blocked transmissions (Maier, 2008; Saleh \& Simmons, 2012; Simmons, 2014). However, this has the disadvantage of reducing the benefits obtained from transparent networks, as stated above.

### 2.2.3 Enabling technologies

The dynamically configurable technologies enabling All-optical flexi-grid networks are Bandwidth Variable Transponders (BVT's), Reconfigurable Optical Add Drop Multiplexers (ROADM's) and Optical Cross-connects (OXC's) (Maier, 2008; Ra-
maswami et al., 2010; Simmons, 2014). BVT's allow for the transmission and receiving of data on arbitrarily sized and positioned wavelengths within the supported range, using various modulation formats (Maier, 2008; Ramaswami et al., 2010; Simmons, 2014). This allows the delivery of data at multiple data rates. ROADM's and OXC's allow for the switching of arbitrary single wavelengths at a node from link to link, and allow the addition and removal of a transmission into and out of the network, respectively (Maier, 2008; Ramaswami et al., 2010; Simmons, 2014). Figure 2.3a and 2.3b show a schematic view of an opaque and transparent OXC, respectively. The difference between both is that in an opaque OXC, the incoming transmission is converted to electrical signals before being retransmitted on the appropriate link, transmissions that should be taken off the network are removed in the electronic domain; while in the transparent OXC all switching is performed in the optical domain with transmissions that should be taken off the network, directed optically towards an output transponder. The difference between ROADM's and OXC's is that ROADM's are only degree two (where the degree of a node is the number of links that leave a node); hence they are used in linear scenarios such as point to point links and ring networks, while OXC's are multi-degree allowing for use in mesh networks (Maier, 2008; Ramaswami et al., 2010; Simmons, 2014). A schematic of the operation of transparent ROADM's is shown in Figure 2.3c. The flexibility offered by ROADM's and OXC's also enables new switching technologies such as Optical Burst Switching (OBS) and Optical Packet Switching (OPS).

The key technology that enable transparent ROADM's and OXC's is the Wavelength Selective Switches(WSS); in the case of flexi-grid, specifically the Liquid Crystal on Silicon (LCoS) based WSS (López \& Velasco, 2016; Tomkos, Azodolmolky, SoléPareta, Careglio, \& Palkopoulou, 2014; Fernandez-Palacios, López, Cruz, \& De Dios, 2014). A representation of the operation of LCoS Switches is given in Figure 2.4. Light to be switched from an input port is directed onto a diffraction grating which splits the light into its constituent wavelengths; the resulting wavelengths are then directed and spatially spread out onto the LCoS, where they are reflected into the appropriate output fibre (López \& Velasco, 2016). The angle at which a wavelength is reflected from the LCoS determines what fibre it will be sent into. This angle is determined by the pattern displayed on the LCoS.

### 2.2.4 Optical Impairments

When designing optical WDM networks, certain physical effects which may prevent transmissions from being received at their destinations occur as a result of the

(a) Opaque Optical Cross-connect (Luke, Larsen, Ehrhardt, \& Jansen, 2005)

(b) Transparent Optical Cross-connect (Luke, Larsen, Ehrhardt, \& Jansen, 2005)

(c) Reconfigurable Optical Add Drop Multiplexer (Ramaswami, Sivarajan, \& Sasaki, 2010)

Figure 2.3: Schematic view of various optical switching elements


Figure 2.4: Illustration of the operation of Liquid Crystal on Silicon (LCoS) (López \& Velasco, 2016)
interaction between the light signals and the optical fibre. They may be classified as either Linear or Non-linear (Shaw, 2004; Maier, 2008; Azodolmolky et al., 2009). Linear impairments are independent of the power of a transmission and are proportional to the length of the fibre. Non-linear impairments are dependent on the number and power of the various transmissions on a fibre, as well as details such as the modulation format and the speed of each transmission. Some optical impairments which may occur on optical fibres are (Shaw, 2004; Maier, 2008; Azodolmolky et al., 2009)

- Attenuation where the received signal strength is weakened over a length of fibre compared to its strength when sent, due to slight manufacturing and splicing defects.
- Dispersion where different components of the transmitted signal travel at different speeds within the fibre and arrive out of phase with each other due to the various optical characteristics of the fibre. This causes the transmitted pulse to widen leading to inter-symbol interference which may prevent the transmitted data from being readable at the receiver. This limits the maximum transmission speed and is dependent on the length of the fibre.
- Cross-talk where signals from different channels interfere with each other.
- Non-linearities are effects that occur at high power levels due to scattering effects caused by molecular vibrations and refractive properties of the fibre.
- Noise introduced by processes in various electrical components.

In this research, the optical impairments which are going to be considered are Attenuation and Cross-talk. Noise within electrical components will not be considered in the simulations as it is not of interest to solving the RWA problem. Dispersion will also not be considered, due to the fact that there are well established methods to deal with dispersion (such as dispersion compensating fibre) in currently deployed networks (Stamatis V. Kartalopoulos, 2004). In order to model the power penalty (in decibels (dB)) experienced by a transmission across a fibre link due to attenuation and cross-talk; the following equation will be used (Boiyo et al., 2015)

$$
\begin{equation*}
\operatorname{Penalty}(d B)=A L+c L \sum_{i \in \mathcal{T}_{f} \backslash\{s\}} \frac{b_{s} 10^{\frac{P_{i}}{10}}}{b_{i} 10^{P_{s}}\left(f_{i}-f_{s}\right)} \tag{2.1}
\end{equation*}
$$

where the constant $c=4.78$ (Assuming On-Off keying modulation), $A$ is the attenuation constant in $\mathrm{dB} / \mathrm{km}, \mathrm{L}$ is the fibre length in $\mathrm{km}, \mathcal{T}_{f}$ is the set of signals traversing the fibre between the start and end time of the signal, $b_{s}$ is the bit rate of the signal, $b_{i}$ is the bit rate of the signal causing the interference, $P_{i}$ is the power of the signal causing the interference in $\mathrm{dBm}, P_{s}$ is the power of the signal in decibelmilliwatts $(\mathrm{dBm}),\left(f_{i}-f_{s}\right)$ is the difference between the central frequencies of the interfering signal and the signal in GHz.

The penalty on the power of a transmission over a length of fibre, due to the effects of other simultaneous transmissions on the length of fibre, as well as due to the effects of attenuation can be obtained by using equation (2.1). The obtained penalty can then be subtracted from the power of the transmission at its source, to obtain the power that will be sensed by a receiver at its destination. If the power of the transmission at the destination is less than the receiver sensitivity at the destination, then the transmission is considered lost.

### 2.3 Optical Burst Switching

In this section, a review of the various optical switching paradigms is given, followed by an in depth description of the operation of burst assembly, routing and wavelength assignment and bandwidth reservation and burst switching.

### 2.3.1 Optical Switching

Current optical networks use a switching paradigm known as Optical Circuit Switching (OCS), where all the bandwidth on a channel (where a channel could be a fibre, a wavelength, or a time slot in a Time Division Multiplexed (TDM) system) is fully dedicated to transmission between a pair of nodes (Maier, 2008; Chen et al., 2004). The channels may be assigned statically or as required for a given amount of time. The disadvantage of this is that, for bursty (irregular) data, the bandwidth of a channel is reserved for a given node, irrespective of if the node has data to transmit. A node might not need to send data, or might only need to send data across a channel for a short period of time, leaving the remaining channel slot unused, while another application might currently need the bandwidth. This is inefficient. A better way would be to use Optical Packet Switching (OPS), where packets sent over a given wavelength channel are independently switched towards their destination (this is known as statistical multiplexing) (Maier \& Reisslein, 2008; Rouskas \& Xu, 2004). However, true optical packet switching is not currently possible due to the absence of cost effective optical buffers which would allow the buffering of packets to enable contention resolution at core nodes, as well as efficient methods for processing each packet within the optical domain (Maier \& Reisslein, 2008; Rouskas \& Xu, 2004).

Optical Burst Switching (OBS) could be considered as a compromise to improve the efficiency of optical networks using currently available technology (Maier \& Reisslein, 2008; Bjornstad et al., 2003; Battestilli, Perros, \& Carolina, 2003; Verma et al., 2000). In an OBS network, packets are buffered and assembled into bursts, at network edge nodes, according to various parameters. When a burst is ready to be sent, the node sends a packet, known as a Burst Control Packet (BCP) on a reserved channel called the control channel, in order to reserve the required bandwidth and setup the switching at intermediate core nodes. The burst is then sent after a certain offset time, in order to give the BCP time to finish setting up the switching at the intermediate nodes. The reserved bandwidth for a transmission is released after the burst has traversed the network, in order to allow its use by another burst. OBS leads to more efficient usage of network resources, since bandwidth on the network is not tied up at any node for a long period, like in OCS. OBS has been shown to offer higher data throughput and lower blocking rates compared to OCS in simulation studies performed by Fei, Yoo, Yokoyama, and Horiuchi (2005) and Liu, Qiao, Yu, and Gong (2006)

### 2.3.2 Burst Assembly

Burst assembly deals with the aggregation of packets into data bursts at the edge nodes (Maier \& Reisslein, 2008; Chen et al., 2004; Battestilli et al., 2003). Burst assembly algorithms could be time based; where only packets that arrive within a predefined period are sent. Burst size based; where the burst is sent once it reaches a predefined minimum size; a mix of both time based and burst size based, where a burst is sent depending on which parameter is satisfied first; or dynamic where either the burst assembly time or burst size is determined dynamically based on network conditions. Burst assembly is important because long bursts hold network resources for long time periods and may therefore cause higher burst losses due to contention, while short bursts increase network overhead because of the increased number of control packets that need to be processed.

### 2.3.3 Routing and Wavelength Assignment

Burst loss in an OBS network can occur due to two factors, wavelength contention between two or more burst reservations on a route (leading to the loss of at least one burst), and linear and non-linear optical impairments that could render the data sent in a burst unreadable at the destination node (Miroslaw Klinkowski et al., 2010; Azodolmolky et al., 2009). Routing and wavelength assignment deals with selecting a route-wavelength combination that will ensure that the current request is fulfilled, the blocking probability of future requests is minimised, that available network resources are used in an efficient manner (Gravett et al., 2017). To that end, routes and wavelengths for a request should be assigned so that they do not contend with existing transmissions on the network; they go through the smallest number of hops and use the least amount of bandwidth possible, to reduce the probability of blocking subsequent transmissions; and they use the shortest paths, in order to minimise the effects of impairments and reduce the latency of the network.

Routing and wavelength assignment on an OBS network could either be static or dynamic (Miroslaw Klinkowski et al., 2010). If it is static then predefined routes and wavelengths for each source-destination pair are pre-computed and bursts between two nodes are sent along their predefined routes. If routing is dynamic, a route will have to be computed or selected for each bursts; this allows the network to automatically respond to varying loads (by allowing the reassignment of unused resources to transmissions which require it) and detrimental events (for instance, the failure of a node might lead the network to use a different route for transmission
which go through the failed node). Routing might be performed at in a distributed manner, at each intermediate node (hereby referred to as node-by-node) or at the source edge node (hereby referred to as source routing) (Miroslaw Klinkowski et al., 2010). In a centralised routing protocol the BCP is sent to a central control node which keeps track of the resources available at each node as well as the network topology. The central control node determines the route that the burst will be sent over and reserves the bandwidth resources at each intermediate node. A centralised protocol is potentially more effective at minimising the Burst Loss Probability (BLP) as the central node has near perfect knowledge of the network state. However, the disadvantages of this is the high control overhead due to the high number of transmissions between the central node and every other node on the network (this also means that the central node is going to have to be equipped with high processing capabilities as it is going to need to process request for every node in the network); there will also be an increase in the latency of the network, due to the requirement that nodes wait for the response of the central node.

Routing and wavelength assignment algorithms may be classified based on the strategy they use in order to improve the performance of a network. RWA algorithms may be considered to be (Miroslaw Klinkowski et al., 2010)

- With single path routing algorithms, the algorithm uses a single path for all transmissions between a source and a destination. This path may be statically set or may be dynamically selected as network conditions change.

The primary example of static single path routing is Shortest Path Routing (SPR). Transmissions between two nodes are sent along the shortest path which was pre-computed offline by an algorithm like Djikstra's.

- With multi-path routing, a set of paths are used to send information between a source and destination in order to try and balance the load across the links of the network. In static routing, preset routes are determined, and assigned to transfer a certain portion of the traffic. In dynamic scenarios the path to be used is selected for each burst.
- Deflection routing can be used in conjunction with single and multi-path algorithms. However, when a wavelength contention occurs, the node at which contention occurs seeks to reschedule the burst on another link which is currently free. The links to which the burst may be rescheduled may be determined statically or dynamically. If static (commonly know as Fixed Alternate (FA) routing), the possible alternative links are pre-determined and fixed. To select an alternative link, the first alternative link found with the required free
wavelength is selected or a link is randomly selected from the set of alternate links on which the required wavelength is available. Alternative links with the required wavelengths available on them could also be chosen based on the shortest path to the destination which may be achieved through them.

If dynamic then the node uses knowledge of the wider network to compute a path towards a burst destination and push the burst towards its destination. An example of dynamic deflection routing is Fixed Alternate (FA). In FA, if an available wavelength can't be found on the shortest path, the shortest available path with a free wavelength is used.

Assigning wavelengths in fixed grid networks is relatively straight-forward, as a free wavelength slot is simply assigned to a transmission. However, in a flexi-grid network, the amount of spectrum assigned needs to be variable in order to cater for multiple transmission rates. When assigning wavelengths to transmissions, the assigned wavelengths must be contiguous and should not overlap with those assigned to other transmissions. Figure 2.5 shows the various methods of catering for the varying spectrum requirements of transmissions presented by (Miroslaw Klinkowski \& Walkowiak, 2011). Each method is described below

- Using fixed wavelength assignment, the spectrum is split into chunks large enough for the largest demands, and a transmission is simply assigned to one of these chunks. This is equivalent to fixed-grid transmission and negates the benefits of flexi-grid.
- Using semi-elastic wavelength assignment, the central frequencies are preselected; however, the assigned spectrum is allowed to grow symmetrically around this central frequency. This is better as it allows for varying size transmission wavelengths; however, it still doesn't fully exploit the benefits of flexi-grid.
- The elastic wavelength assignment approach allows for free selection of the central frequency and channel size. In practice this is performed by selecting a potential centre and expanding the channel as required upwards or downwards if possible, then finally, recalculating the centre of the newly selected wavelength. If there is no room for expansion, the centre frequency may be chosen at another location on the spectrum.

Four basic heuristics, which may be used to select a wavelength when performing fixed wavelength assignment, or the wavelength centre when performing semi-elastic and elastic wavelength assignment are (Zang, Jue, Mukherjee, et al., 2000)

- Random - a free wavelength is selected randomly with uniform probability.
- First-Fit - wavelengths are numbered sequentially, from the lower limit of the available spectrum to the upper limit. When searching for a free wavelength, the wavelengths are considered in ascending order, with the first free one being selected. The idea behind first-fit is to pack all current transmissions towards the beginning of the spectrum range, hence increasing the probability that more wavelengths will be available at the higher end of the spectrum to be used for longer transmissions.
- Least-Used - this heuristic determines which of the currently available wavelengths is the least used wavelength on the network, and assigns this to new transmissions in an attempt to balance transmissions among all wavelengths. It requires global knowledge of how many times each wavelength has been used.
- Most-Used - this heuristic determines which of the currently available wavelengths is the most used wavelength on the network, and assigns this to new transmissions. Its goal similar to first fit is to pack all transmissions within a small region of the spectrum, in order to allow more wavelength for longer transmissions. It requires global knowledge of how many times each wavelength has been used.

Two simple greedy algorithms for RWA is the combination of SPR with random wavelength selection or First-Fit wavelength selection.

### 2.3.4 Bandwidth Reservation and Switching

The protocol to reserve the network bandwidth could be either distributed or centralised. In a distributed protocol, the edge node sends the BCP to the first node on the way towards the destination address, the first node then passes on the BCP to the next node and so on until the BCP reaches the destination address.

The bandwidth reservation and switching in intermediate nodes can be performed in two ways; Just-In-Time (JIT) and Just Enough Time (JET) (Chen et al., 2004; Teng \& Rouskas, 2003; Kirci \& Zaim, 2006). In JIT (Figure 2.6a), the intermediate nodes perform the bandwidth reservation and switching immediately after they receive the BCP. The bandwidth is reserved for the whole period of the transmission of the burst, and any request that come in for that wavelength during that period is discarded.

In JET (Figure 2.6b), the intermediate node use the offset time carried by the


Figure 2.5: Elastic wavelength assignment (Miroslaw Klinkowski et al., 2013)

BCP to perform the switching operation at the last possible moment. A burst is successfully scheduled using JET if the reservation request is for a time period after the last use of the wavelength or if the reservation request will fit into a time gap (void) between the end of a burst transmission and the start of the next one. There are various scheduling algorithms for JET such as Horizon and LAUC-VF; each with time complexity and bandwidth utilisation benefits (Nleya \& Mutsvangwa, 2014).

Teng and Rouskas (2003) and Kirci and Zaim (2006), recommend the use of JIT over JET due to its simplicity of implementation and similarity in performance over current hardware which have switching speeds in the millisecond range. Teng and Rouskas (2003) performs an analytical study of JIT and JET to determine in what range their performance are equivalent; they then perform computer simulations to determine the BLP over a variety of simulations and find the performance of JIT and JET to be comparable, particularly in scenarios with a small number of wavelengths. Kirci and Zaim (2006) performs simulations to compare JIT and JET over various network scenarios, and finds their performance to be comparable and the difference in performance to be nearly constant across all scenarios tested.

The offset time between the BCP and burst transmissions is (Teng \& Rouskas, 2003)

$$
T_{\text {offset }}=n T_{\text {setup }}+T_{O X C}
$$



Figure 2.6: Network timing diagrams, showing the operation of JIT and JET bandwidth reservation schemes. (Teng \& Rouskas, 2003)

Where $n$ is the number of core nodes on the selected route, $T_{\text {setup }}$ is the amount of time required for BCP processing at each core node, and $T_{O X C}$ is the switching time required by the last core node.

Bandwidth reservations may be released at each node in two ways; either the edge node sends a trailing release control message after the burst transmission, or the core nodes use their knowledge of the offset time and the length of the burst to estimate the time after which they may safely release the reserved bandwidth (Battestilli et al., 2003; Nleya \& Mutsvangwa, 2014).

### 2.4 Simulation Principles

A simulator is an imitation of a complex entity, system, phenomena, or process, in order to study its behaviour (Guizani, Rayes, Khan, \& Al-fuqaha, 2010; Banks, Nelson, Carson, \& Nicol, 2010). A simulator typically employs one or more models in order to functionally represent a real world system to some degree of accuracy; where a model is a representation of a system for the purpose of studying that system (Guizani et al., 2010; Banks et al., 2010; Sokolowski \& Banks, 2009). Simulators are often used to study systems which are too complex to study analytically; and too expensive to study in the real-world. They allow observers to obtain insight into how the various aspects of a system affect each other and their relative importance to
the functioning of the system (Guizani et al., 2010; Banks et al., 2010; Sokolowski \& Banks, 2009). Simulators also allow designers test the performance of new system designs (Guizani et al., 2010; Banks et al., 2010; Sokolowski \& Banks, 2009), hence aiding the design process.

In this section, a brief introduction to Discrete Event Simulation (DES) is presented, followed by a brief introduction to the various simulator frameworks which might be used to build the OBS simulator. Finally, the Reduced Link Load model which will be used to validate the OBS simulator is described.

### 2.4.1 Discrete Event Simulation

Simulation models can either be discrete or continuous (Guizani et al., 2010; Banks et al., 2010; Sokolowski \& Banks, 2009). A model is continuous if the state variable change continuously over time while a system is considered to be discrete only if the state variables change at a discrete, finite set of points in time. Computing and computer network systems are often modelled discretely, while biological and ecological systems are often modelled continuously. No real life system can be fully modelled exactly discretely or continuously; however, what determines what kind of model is used, is what behaviour the researcher is most interested in.

DES involves simulating a finite sequence of events, where each event is atomic (that is it leaves the system in a consistent state) and has a specific start and end time. The major components of a DES are an event queue, which keeps track of all events waiting to happen in the future; State variables which together completely describe the state of the system, and a simulation clock which keeps track of the global simulated time. Events are instantaneous occurrences which might change the state of the system. Events may create other events. The simulation clock can be advanced in an event driven manner or in a fixed increment manner. If done in a fixed increment manner; time is divided into small, fixed increments and all events occurring within each increment are processed before moving the simulation time forward. If done in an event driven manner; time is incremented to the start time of the next event in the event queue, after this event is processed, the clock is set to the time of the next event. The event driven manner is more efficient, particularly in scenarios where the time between inactive periods is large.

### 2.4.2 Simulator Frameworks

In this section, three free, open-source simulator frameworks which may be used to develop an OBS simulator are described. These three were chosen, due to the fact that they are actively being developed, they already have pre-built libraries simulating real world protocols, and they are conducive to large scale simulation studies (that is, simulation studies with lots of runs).

### 2.4.2.1 Omnet++

The Omnet++ framework is an open-source, Discrete Event Simulator framework that can be used to build simulations for different scenarios (not just communication networks) ("Omnet++ Discrete Event Simulator - Home", 2016; András Varga \& Hornig, 2008; Andras Varga, 2010). Omnet++ uses a component based architecture, which allows for easier development of a simulation as key aspects of a model can be identified and implemented in separate modules. Modules may be nested in other modules to aggregate their behaviours and communication between modules is done via messages passed between modules. Development of modules is done using C++ and a module description language call NED, which is used to define the structure, parameters and connections of modules. NED is also used to define topologies, data collection and input parameters to simulations.

Omnet++ is well documented which makes the API easy to learn. It also has a sizeable active community as well as multiple open-source, community written simulation models (most notably the INET framework ("INET Framework", 2018) which models various network protocols). It provides an Eclipse based IDE which allow for developing of modules in C++ and NED, as well as data analysis of result files obtained from simulations. Data analysis may also be done in R using the plugin provided by the developers. The framework provides a graphical user interface for simulations, which may be used to trace the operation of simulations and find errors while the simulation is under-way, hence, easing debugging.

### 2.4.2.2 IKR

IKR is an open-source, Discrete Event Simulator framework, which uses a component based architecture ("Institute of Communication Networks and Computer Engineering (IKR) - IKR Simulation and Emulation Library", 2017; Sommer \& Scharf, 2010). The framework provides a C++ and Java version. The framework
comes with a number of pre-written components and allows on-line data processing. However, topologies and simulations have to be created in code, increasing the difficulty of putting simulations together. The C++ version has a TCP library which may be used in simulations; and both versions have an emulation library which may be used in conjunction with the simulation library to simulate the operation of real nodes.

### 2.4.2.3 NS3

NS3 is an open-source, Discrete Event Simulator framework primarily built to simulate various real world network protocols ("ns-3", 2018; Riley \& Henderson, 2010). Hence, they are tightly coupled into it design of the simulator. This tight coupling makes it difficult to understand and modify the simulator for own use, especially as the documentation is not robust. Nodes which do not allow nesting of other modules are its lowest abstraction. Topologies and simulations have to be created using $\mathrm{C}++$, hence increasing the difficulty of putting simulations together. While it offers result recording in the form of logging and traces, it does not provide any tools for post-processing (external software has to be used).

### 2.4.3 Reduced Link Load Model

A method which may be used to validate an OBS network simulation is the Reduced Link Load Model by Rosberg, Zukerman, and White (2003). It was adapted from the work done by Kelly (1986) on estimating blocking probabilities in OCS networks. Given a network with specified routes for the transmission of bursts, this model allows the steady state blocking probability of each link in a network to be approximated by considering only the reduced offered load caused by the blocking on other links in the network. This knowledge is then used to determine the blocking along a route and hence in the entire network.

Assuming that burst are offered to route $r$ according to a poisson process with rate $\lambda_{r}$ and that blocking events in a link occur independently from blocking events in other links (that is all wavelengths on a link are considered equally likely to be in use, hence the blocking on a link is dependent only on the load offered to the link); then the blocking probability of a link $j$, with $W_{j}$ wavelength channels and offered load $\rho_{j}$, on a network with full wavelength conversion provided at each link, is given by the Erlang-B formula

$$
\begin{equation*}
B_{j}=H\left(\rho_{j}, W_{j}\right)=\frac{\rho_{j}^{W_{j}} / W_{j}!}{\sum_{k=0}^{W_{j}} \rho_{j}^{k} / k!} \tag{2.2}
\end{equation*}
$$

However, this can be simplified for a network which maintains the wavelength continuity constraint to (Luo, $\mathrm{Hu}, \& \mathrm{Li}, 2004$ )

$$
\begin{equation*}
B_{j}=H\left(\rho_{j}, W_{j}\right)=\frac{\rho_{j}}{1+\rho_{j}} \tag{2.3}
\end{equation*}
$$

$\rho_{j}$ is given by

$$
\rho_{j}=\mu_{j}^{-1} \sum_{r \in \mathcal{R}} \lambda_{r} \prod_{i=1}^{J}\left(1-I(i, j, r) \cdot B_{i}\right)
$$

where $\mathcal{R}$ is the set of all routes on the network, $\mu_{j}$ is the service rate of burst in link $j$, measured in bursts per time unit, $J$ is the number of links on the network and $I(i, j, r)$ equals one if $i, j \in r$ and link $i$ strictly, but not necessarily immediately precedes link $j$ on route $r$; or zero otherwise. The link blocking probabilities can then be obtained by the method of successive substitution. For any given vector of blocking probabilities $\mathbf{B}$, define the transformation vector $V(\mathbf{B})=\left(V_{1}(\mathbf{B}), V_{2}(\mathbf{B}), \ldots, V_{J}(\mathbf{B})\right)$ by

$$
V_{j}(\mathbf{B})=H\left(\mu_{j}^{-1} \sum_{r \in \mathcal{R}: j \in r} \lambda_{r} \prod_{i=1}^{J}\left(1-I(i, j, r) \cdot B_{i}\right), W_{j}\right)
$$

The successive substitution method begins with an initial blocking probability vector $\mathbf{B}^{(0)}=\mathbf{1}$, and repeatedly computes $\mathbf{B}^{(n)}=V\left(\mathbf{B}^{(n-1)}\right)$ for $n=1,2,3, .$. , until $\mathbf{B}^{(n)}$ is sufficiently close to $\mathbf{B}^{(n-1)}$. The approximate blocking probability of bursts offered to route $r$ may then be obtained using the following expression

$$
B(r)=1-\prod_{i \in r}\left(1-B_{i}\right)
$$

and the blocking probability of an arbitrary burst is given by

$$
B=\frac{1}{\Lambda} \sum_{r \in \mathcal{R}} \lambda_{r} . B(r)
$$

where $\Lambda=\sum_{r \in \mathcal{R}} \lambda_{r}$
The value of $\mu_{j}$ is dependent on the reservation scheme being considered on the system. For JIT, $\mu_{j}=1 /\left(E\left(S_{j}\right)+E\left(D_{j}\right)\right)$, while for JET $\mu_{j}=1 / E\left(S_{j}\right)$. Where
$E\left(S_{j}\right)$ is the mean burst service time in link $j$ and $E\left(D_{j}\right)$ is the mean offset time between BCPs and bursts in link $j$.

### 2.5 Ant Colony Optimisation

Swarm Intelligence (SI) is the term given to the emergent problem solving behaviour that occurs over time when groups of independent agents, following simple rules, communicate with each other either directly or indirectly by acting on their local environments (Engelbrecht, 2007). Ant Colony Optimisation (ACO) is a biologically inspired SI algorithm, which attempts to emulate the behaviour of a colony of ants foraging for food (M. Dorigo \& Di Caro, 1999); and is one of multiple approaches of interest in flex-spectrum optical networks particularly with OBS, as a way of solving the Routing and Wavelength Assignment (RWA) problem in such networks.

In nature, ants accomplish the task of foraging for food by depositing small amounts of pheromones on their path, when returning to their nest from a food source. When an ant discovers a food source, it carries some food to the nest while depositing pheromones along its path. Other ants that come across the path might then probabilistically select the path based on the strength of its pheromone concentration. The more ants select a given path, the stronger its pheromone concentration will be and the greater the probability of other ants selecting it becomes; while the pheromone concentration of less used paths become weaker over time as the pheromones deposited on them evaporate. A factor that influences the pheromone concentration of a path is the length of the path; this is due to the fact that ants will return quicker along a shorter path than along a longer one. Hence, reinforcing the pheromones along that path faster and making the pheromone concentration higher than other paths. In this way the shortest path is chosen by the ants. An illustration of this is given in Figure 2.7.

An example of how this can be applied to a simple network, made of nodes connected by single bi-directional links, is the Simple Ant Colony Optimisation (SACO) (Marco Dorigo \& Stützle, 2001), which is the simplest ACO algorithm. In SACO, the aim is to find the shortest path between two nodes. Initially, each edge is assigned a small random value which represents the initial pheromone concentration of the edge, and $k$ ants are placed on the source vertex. For each iteration, each ant has to construct a path between the source node and the destination node by making a decision of

(a) All ants are in the nest. There is no pheromone in the environment.
(b) The foraging starts. In probability, $50 \%$ of the ants take the short path (symbolized by circles), and $50 \%$ take the long path to the food source (symbolized by rhombs).

(c) The ants that have taken the short path have arrived earlier at the food source. Therefore, when returning, the probability to take again the short path is higher.
(d) The pheromone trail on the short path receives, in probability, a stronger reinforcement, and the probability to take this path grows. Finally, due to the evaporation of the pheromone on the long path, the whole colony will, in probability, use the short path.

Figure 2.7: Illustration of the shortest path finding ability of ant colonies (Blum, 2005)
which link to follow according to the following transition probability

$$
p_{i j}= \begin{cases}\frac{\left(\tau_{i j}\right)^{\gamma}}{\sum_{j \in \mathcal{N}_{i}}\left(\tau_{i j}\right)^{\gamma}} & \text { if } j \in \mathcal{N}_{i} \\ 0 & \text { if } j \notin \mathcal{N}_{i}\end{cases}
$$

Where $\tau_{i j}$ is the amount of pheromones on the link from node $i$ to node $j, \mathcal{N}_{i}$ is the set of feasible nodes connected to node $i$ (that is, every adjacent node except the preceding node), and $\gamma$ is a constant which amplifies or weakens the influence of the pheromone concentrations. When all ants reach the destination node, each ant then retraces its route back to the source node, while modifying the pheromone concentration on each link along its path. The change in pheromones $\Delta \tau_{i j}$ is calculated based on the length of the link such that

$$
\Delta \tau_{i j} \propto \frac{1}{L}
$$

Where $L$ is the length of the path constructed by the ant. The pheromone concentration on each link is also reduced at each iteration, in order to emulate pheromone
evaporation, according to the following expression

$$
\tau_{i j}(t+1)=(1-\alpha) \tau_{i j}(t)
$$

Where $\alpha \in[0,1]$ is a pre-set constant which specifies the rate at which pheromones evaporate. The algorithm is executed until a given termination condition is met. Examples of termination conditions could be when a maximum number of iterations has been exceeded, when an acceptable solution has been found or when all or most ants follow the same path.

There are multiple variations of the simple algorithm presented above, which differ based on the transition probability used, how pheromone deposition and evaporation work, and the number of ants used. Such as the Ant System (AS) (Marco Dorigo, Maniezzo, \& Colorni, 1996), Fast Ant (FANT) (Taillard, 1998), Ant Colony System (ACS) (Marco Dorigo \& Gambardella, 1997) and Max-Min Ant System (MMAS) (Stützle \& Hoos, 2000).

### 2.6 Ant Colony Optimisation RWA Algorithms

In this section, various ACO based RWA algorithms designed to improve the performance of OBS networks are presented in order of publication. After an introduction to all the algorithms, this section is ended with a discussion of various common traits and shortcomings of the algorithms.

While most of the discussed algorithms cannot be applied in their described forms to flex spectrum scenarios; an understanding of how ACO might be applied to OBS networks is obtained by studying these algorithms. During this literature review, it was observed that the algorithms were not tested against the same algorithms and in comparable network conditions, hence no conclusions can be made about the effectiveness of one over another.

### 2.6.1 ACOR

Ngo, Jiang, and Horiguchi (2006) present a solution to the RWA problem which they state could be applied to OCS and OBS networks. However, the authors evaluated the algorithm only on an OCS network; hence nothing can be said about its performance on an OBS network. The WC constraint is considered in the design; however,
wavelength assignment is not performed using ACO, but is instead performed in a first fit manner. Key features of the algorithm are

- Each node in the network holds a routing table where the entries of the table hold a pheromone value $\tau_{i j}^{i d}$ which indicates the probability of an ant using the link between the current node $i$ and an adjacent node $j$ to get to a destination node $d$. The sum of the pheromone values of all links for a given destination must equal 1.
- Ants are continuously sent out to random destinations at pre-set time intervals, and based on a pre-set probability to random destinations, in order to allow continuous monitoring of the network state. An ant only moves in a one way direction and is killed on reaching its destination, when its Time-To-Live (TTL) is exceeded, or a circular path is discovered.
- Ants decide what link to follow towards their destination based on the pheromone concentration of the entry for that link. Ants may exploit the entry with the highest pheromone value in the table with probability $(1-g)$; and may explore new paths with probability $g$, in order to prevent stagnation due to constant usage of a good path).
- Ants modify the pheromone value at every node they pass. Updates to the entry of the routing table corresponding to the source node $s$ of the ant, at a node $i$, by an ant coming from node $j$ are made in the following manner

$$
\begin{gathered}
\tau_{i j}^{i s}(t+1)=\frac{\tau_{i j}^{i s}(t)+\Psi}{1+\Psi} \\
\text { where } \Psi=v \Lambda+(1-v) X \\
\Lambda=e^{-\delta l} \\
X=e^{\epsilon w}-1
\end{gathered}
$$

with $0 \leq v \leq 1$ being a constant which determines the influence of $\Lambda$ and $X$ on the change in the pheromone value, $l$ is the difference between the current path of the ant and the shortest path to the destination of the ant, $\delta$ and $\epsilon$ are constants, and $w$ is the ratio of currently unused wavelengths on the path the ant has moved along.

In order to maintain the constraint that the sum of the pheromone values of all links for a given destination must equal 1 , the entries corresponding to all
other links $k \neq j$ connected to node $i$ are updated as follows

$$
\tau_{i k}^{i s}(t+1)=\frac{\tau_{i k}^{i s}(t)}{1+\Psi}, \text { where } k \neq j
$$

With this algorithm, a wavelength and route can be immediately chosen by the source node on receiving a request. Routing of a request is performed at each node using the link with the highest pheromone value, for the stated destination of the request. A free wavelength is chosen at the source node and used for the whole transmission. If the chosen wavelength is unavailable at an intermediate node then the request is blocked. The authors found that the algorithm offered a better blocking probability than SPR and FA across all network configurations they considered.

### 2.6.2 DABR

Pedro, Pires, and Carvalho (2009) describe a distributed ACO based algorithm applied to OBS networks. The algorithm is evaluated against a centralised routing algorithm and the shortest path routing algorithm. It is found to perform significantly better than the shortest path routing algorithm, but worse than the centralised routing algorithm. It is never stated if the simulations were performed under the WC constraint, the design of the algorithm suggest it wasn't. Key features of the algorithm are

- Each node maintains two tables; a pheromone table which is used by ACO to keep track of the pheromone values assigned to an adjacent node based on the source-destination pair; and a routing table which is used to route burst and avoid circular path formation. Each type of table is maintained by two different kind of ants, the Explorer Ant Packet (EAP) and Referee Ant Packet (RAP) respectively. Each table consist of pheromone values $\tau$ for each source-destination ( $s, d$ ) pair on the network, and the output ports of the node that will push the burst towards the destination. A constraint is placed on the pheromones of the source-destination pairs, such that the sum of pheromones related to a source-destination pair should equal 1 . The values in the pheromone tables of a node $i$, are initially set to be biased towards paths with least number of hops in the following manner

$$
\tau_{i j}^{s d}= \begin{cases}\frac{\left(1+h^{j d}\right)^{-1}}{\sum_{k \in \mathcal{A}_{i}}\left(1+h^{k d}\right)^{-1}} & j \in \mathcal{A}_{i}, i \neq d, s \neq d \\ 0 & \text { otherwise }\end{cases}
$$

Where $\mathcal{A}_{i}$ is the set of nodes adjacent to $i, h^{j d}$ and $h^{k d}$ is the number of links on the shortest path between nodes $j$ and $d$ and nodes $k$ and $d$, respectively.

- Whenever a burst is ready to be sent, an EAP with a resource reservation request is sent out with a preset probability from the source node, else the burst is normally routed as stated below. The EAP that is sent is known as a Forward EAP (F-EAP). At each node $i$, the EAP decides what unvisited adjacent node to move to, according to the following probability

$$
p_{i j}^{s d}= \begin{cases}\frac{\tau_{i j}^{s d}}{(1+v) \sum_{k \in \mathcal{A}_{i}, k \notin \mathcal{V}_{E A P}} \tau_{i k}^{s d}}+v \frac{w_{i j}}{(1+v) \sum_{k \in \mathcal{A}_{i}, k \notin \mathcal{V}_{E A P}} w_{i k}} & j \in \mathcal{A}_{i}, j \notin \mathcal{V}_{E A P} \\ 0 & \text { otherwise }\end{cases}
$$

Where $v$ is a preset constant, and $\mathcal{V}_{E A P}$ is the set of nodes previously visited by the EAP, and $w_{i j}$ is the ratio of available wavelengths over link $i j$ to the total number of wavelengths on the network.

- When the F-EAP reaches the destination node, it sends a Back EAP (B-EAP) to the source node, which updates the pheromone table, at each node it passes in the following manner

$$
\begin{gathered}
\tau_{i j}^{s d}(t+1)=\tau_{i j}^{s d}(t)+G\left(1-\tau_{i j}^{s d}-u \frac{\left|\mathcal{A}_{i}\right|-1}{\left|\mathcal{A}_{i}\right|}\right) \\
\tau_{i k}^{s d}(t+1)=\tau_{i k}^{s d}(t)-G\left(\tau_{i k}^{s d}-\frac{u}{\left|\mathcal{A}_{i}\right|}\right), k \in \mathcal{A}_{i}, k \neq j .
\end{gathered}
$$

Where $u$ is the minimum amount of pheromone that must be split between the adjacent nodes, and $G$ is calculated as follows

$$
G=a \frac{Y(Q)}{Y(1)}
$$

with $a$ being the maximum pheromone that may be deposited by a B-EAP, and

$$
\begin{gathered}
Y(x)=\left(1+\left(e^{\left.\frac{1}{x \mid V_{E A P}}\right)}\right)\right)^{-1} \\
Q= \begin{cases}q^{s d} / \hat{q}^{s d} & q^{s d}<\hat{q}^{s d} \\
1 & \text { otherwise }\end{cases} \\
q^{s d}=\frac{1}{B^{s d}} \\
B^{s d}=1-\prod_{k=1}^{\left|V_{E A P}\right|-1}\left(1-B_{v_{k} v_{k+1}}\right)
\end{gathered}
$$

where $B^{\text {sd }}$ is the average burst blocking probability of the routing path between a source $s$ and destination $d, q^{s d}$ is the currently estimated fitness of a routing path, and $\hat{q}^{s d}$ is the best fitness value estimated by $\Upsilon$ previous ants which is maintained at each node. $B_{v_{k} v_{k+1}}$ is the estimated burst blocking probability in the fibre link between $v_{k}$ and $v_{k+1}$. Where $v_{k}$ is a node in the ordered set of nodes visited by F-EAP.
$B_{v_{k} v_{k+1}}$ is estimated as (assuming a network with full wavelength conversion)

$$
\begin{equation*}
B_{v_{k} v_{k+1}}(M, W)=\frac{M^{W} / W!}{\sum_{z=0}^{W} M^{z} / z!} \tag{2.4}
\end{equation*}
$$

where $W$ is the total number of wavelengths and $M=W-U_{v_{k} v_{k+1}}$, with $U_{v_{k} v_{k+1}}$ being the number of available wavelengths on the fibre link between $v_{k}$ and $v_{k+1}$. Equation (2.4) is an adaptation of the Erlang-B formula given in Equation 2.2.

- The routing table is modified whenever a B-EAP arrives at a source node, informing the node of a new best adjacent node to a destination. The source node then sends an F-RAP through the new best adjacent node. At each node the F-RAP encounters, it compares the next best node against the last nodes visited to determine if there are any circles. if any nodes are the same, the F-RAP is discarded and leaves the routing tables unchanged, else if it reaches the destination node, a B-RAP is sent back along the path to update the routing tables, such that the path used to route data burst between the source and destination, is the current path.

When a burst request comes in, the algorithm determines what path the burst should take at each node, based on the dominant route for the specified source-destination pair in the routing tables. A wavelength is randomly selected from wavelengths at the source node which are locally available.

### 2.6.3 ACRWA

In (Triay \& Cervelló-Pastor, 2010), the authors describe an ACO based solution to the RWA problem on OBS networks in the presence of the WC constraint. The algorithm devised by the authors is based on the Ant Colony System (ACS)(Marco Dorigo \& Gambardella, 1997) ACO algorithm. Key features of this algorithm are

- An initialisation process is performed at each node, in order to populate lists of neighbouring nodes and available wavelengths, and determine the shortest
route from a node to every other node in the network using a deterministic algorithm.
- A pheromone table which consist of pheromone values $\tau_{i j k}$ assigned to switching configurations is used at each node; where each switching configuration consist of an input link $i$, an output link $j$ and a wavelength available on both links $k$. The output link $\Omega$ and wavelength $\Theta$ from source node $s$ to destination node $d$ is initially chosen at $s$ in the following manner

$$
(\Omega, \Theta)=\arg \max _{j \in \mathcal{N}_{s d}, k \in \mathcal{W}_{s d}}\left\{\tau_{s j k}\left(\eta_{s j}^{s d}\right)^{\beta}\right\}
$$

Where $\mathcal{N}^{\text {sd }}$ and $\mathcal{W}^{\text {sd }}$ are the sets of all currently available, feasible nodes and wavelengths respectively, that would allow for data transmission from the source node $s$ to destination node $d . \eta_{s j}^{s d}$ is equal to the reciprocal of the shortest path from the source node $s$ to the destination node $d$ through output port $j$, and $\beta$ is a preset parameter, which can be changed to make the algorithm favour shorter paths.

- After the initial selection at the source node, the method used to select the output link $\Omega$ at subsequent nodes is decided by generating a random value $g \sim U(0,1)$. If $g \leq g_{0}$ then

$$
\Omega=\arg \max _{j \in \mathcal{N}_{i d}} \tau_{i j k}\left(\eta_{i j}^{i d}\right)^{\beta}
$$

else if $g>g_{0}$, then the probability of an output link $j$ being selected is

$$
p_{i j k}=\frac{\tau_{i j k}\left(\eta_{i j}^{i d}\right)^{\beta}}{\sum_{m \in \mathcal{N}_{s d}} \tau_{i m k}\left(\eta_{i m}^{i d}\right)^{\beta}}
$$

Where $g_{0} \in[0,1]$ is a user specified parameter.

- Ants are implemented as BCPs, with forward BCPs used to perform RWA, and feedback BCPs (called BCP-ACKs) which have knowledge of the outcome of the reservation process, and move in reverse along the path. The pheromone concentrations of the various switching configurations at each node is updated on the forward pass of the BCP, on successful switching reservation at the current node according to the following expression

$$
\tau_{i j k}(t+1)=\tau_{i j k}(t)+\xi e^{-\delta l}
$$

Where $\xi$ and $\delta$ are user specified parameters, and $l$ is the difference between
the length of the ant's current path and the global shortest path between the source and destination. Pheromone concentrations are also updated on the backward pass of the BCP, after the whole path to the destination has been successful reserved.

$$
\tau_{i j k}(t+1)=(1-v) \tau_{i j k}(t)+v \gamma_{i j} \Delta \tau_{i j k}
$$

Where $v$ is a pre-set constant and $\Delta \tau_{i j k}=e^{-\phi l}$ with $\phi$ a preset constant, and

$$
\gamma_{i j}= \begin{cases}1 & \text { if successful reservation } \\ -1 & \text { if unsuscessful reservation }\end{cases}
$$

note that the update on the backward pass also handles pheromone evaporation. By using the pheromone updates as described above, congestion levels at each node is monitored.

The algorithm is compared against random routing and wavelength assignment, SPR with random wavelength assignment and SPR with first fit wavelength assignment, across various load and network configuration and was found to outperform each of the benchmarks. A shortcoming of this algorithm is that the authors did not present a method of offset calculation to be used in conjunction with the algorithm. This is necessary as the length of the potential route is not known at the edge node making the transmission. To get around the need to calculate an offset the authors evaluated the algorithms on an alternative OBS architecture called Offset Time Emulated OBS (E-OBS) (Mirosław Klinkowski, Careglio, Solé-pareta, Marciniak, \& Member, 2009). E-OBS uses long stretches of fibre (called Fibre Delay Lines), as buffers at every core node, in order to keep the offset time constant for all transmissions. The disadvantage of E-OBS is the added complexity and bulkiness of the Fiber Delay Lines added at every core node.

### 2.6.4 UCBRWA

UCBRWA is an ACO RWA algorithm for OBS networks which aims to find a balance between exploration and exploitation, by applying the Upper Confidence Bound 1 (UCB1) formula in the algorithm (Gravett, du Plessis, \& Gibbon, 2016). The UCB1 formula was presented in (Auer, Cesa-Bianchi, \& Fischer, 2002), as part of a solution to the multi-armed bandit problem (Katehakis \& Veinott, 1987).

Each edge node in a network which uses UCBRWA, holds a pheromone table $\Xi$,
which contains $n$ entries for each possible destination $d$. Each entry $\zeta_{m}$ is a tuple $<r_{m}^{s d}, \varpi_{m}, \sigma_{m}, \chi_{m}>$ consisting of a route $r_{m}^{s d}$ between the current edge node $s$ and a destination $d$, a wavelength $\varpi_{m}$, a success counter $\sigma_{m}$ and a fail counter $\chi_{m}$. The entries are created by randomly selecting a route from a candidate list of $K$-shortest paths, randomly selecting a wavelength from the set of available wavelengths, and finally initialising all counters to zero.

When a burst request is made, the source node $s$ determines which entry to use for the transmission by generating a random value $g \sim U(0,1)$, which determines what selection process to follow

- If $g<\alpha_{1}$ (where $\alpha_{1} \in(0,1]$ ), then an entry $\hat{\zeta}$ is chosen according to the following expression

$$
\begin{equation*}
\hat{\zeta}=\arg \max _{\zeta_{m} \in \Xi_{s d}}\left(\frac{\sigma_{m}}{\sigma_{m}+\chi_{m}}\right)+\left(C \sqrt{\frac{2 \ln \left(\sigma_{s d}+\chi_{s d}\right)}{\sigma_{m}}}\right) \tag{2.5}
\end{equation*}
$$

where $C$ is a pre-set constant which determines the probability of exploration, $\Xi_{s d}$ is the set of entries in the pheromone table which are valid for routing between $s$ and $d, \sigma_{m}$ and $\chi_{m}$ are the number of successful and failed transmissions, respectively, for entry $\zeta_{m}, \sigma_{s d}$ and $\chi_{s d}$ are the total number of successes and failures, respectively, for all $\zeta_{m} \in \Xi_{s d}$.

The two terms of equation (2.5), encourage exploitation, and exploration, respectively. The first term encourages exploitation of the entry with the highest success ratio, while the second term is a penalty term, to encourage exploration of less explored entries

- If $g>\alpha_{1}$, then a new entry is generated by randomly selecting a wavelength and a path (with source $s$ and destination $d$ ) from the candidate list of $K$ shortest paths. The new entry is used to route the current request, and replace the entry for routing between $s$ and $d$ with the lowest percentage success value in the pheromone table.

The forward ant is then sent in the form of the BCP along the chosen route and wavelength, in order to set up the switching configuration at each intermediate node. The burst is sent shortly after sending the BCP. After a burst has been successfully delivered, a feedback ant is sent in the form of an ACK message to inform the source node of a successful or failed delivery, to allow it update the counters of a pheromone concentration. The algorithm is evaluated against ACRWA in fixed grid scenarios, and is found to out-perform ACRWA, in all configurations considered.

### 2.6.5 FSAC

FSAC is a RWA algorithm for OBS networks, specifically designed to effectively route data over flex-spectrum networks under the WC constraint, while considering the effects of network impairments (Gravett et al., 2017). The algorithm is evaluated in both fixed grid and grid-less optical networks and is found to out-perform both SPR and ACRWA on all topologies and loads investigated.

Each edge node in a network which uses FSAC, holds a pheromone table $\Xi$, which contains $n$ entries for each possible destination $d$. Each entry $\zeta_{m}$ is a tuple $<$ $r_{m}^{s d}, \varpi_{m}, \sigma_{m}, \chi_{m}, \tau_{m}, \eta_{m}>$ consisting of a route $r_{m}^{s d}$ between the current edge node $s$ and a destination $d$, a wavelength $\varpi_{m}$, a success counter $\sigma_{m}$, a fail counter $\chi_{m}$, a pheromone value $\tau_{m}$, and the desirability value of an entry $\eta_{m} . \eta_{m}$ is given as the reciprocal of the length of the route $r_{m}^{s d}$. The entries are created by randomly selecting a route from a candidate list of $K$-shortest paths, randomly selecting a wavelength from the set of available wavelengths, initialising all counters to zero, and initialising the pheromone value to a small random value.

When a burst request is made, the source node $s$ determines which entry to use for the transmission by by generating a random value $g \sim U(0,1)$ which will determine if the burst is going to exploit previous routes or explore new routes as follows

- If $g<\alpha_{1}$ (where $\alpha_{1} \in(0,1]$ ), then an entry $\hat{\zeta}$ is chosen according to the following expression

$$
\begin{equation*}
\hat{\zeta}=\arg \max _{\zeta_{m} \in \Xi_{s d}} \tau_{m} \eta_{m}^{\beta} \tag{2.6}
\end{equation*}
$$

where $\eta_{m}$ the desirability of an entry, $\tau_{m}$ the pheromone concentration of an entry; $\beta$ is a constant and $\Xi_{s d}$ is the set of entries in the pheromone table for routing between nodes $s$ and $d$.

- If $g<\left(\alpha_{1}+\alpha_{2}\right)$, then the probability of an entry $o$ being selected is given by

$$
\begin{equation*}
p_{o}=\frac{\tau_{o} \eta_{o}^{\beta}}{\sum_{\zeta_{m} \in \Xi_{s d}} \tau_{m} \eta_{m}^{\beta}} \tag{2.7}
\end{equation*}
$$

- If $g \geq\left(\alpha_{1}+\alpha_{2}\right)$, then a new entry is generated by randomly selecting a wavelength and a path (with source $s$ and destination $d$ ) from the candidate list of $K$-shortest paths. The new entry is then used to route the current request, and replace the entry for routing between $s$ and $d$ with the lowest pheromone value in the pheromone table.

The forward ant is then sent in the form of the BCP along the chosen route and
wavelength, in order to set up the switching configuration at each intermediate node. The burst is sent shortly after sending the BCP. After a burst has been successfully delivered, a feedback ant is sent in the form of an acknowledgement message (hereby referred to as BCP-ACK) to inform the source node of a successful or failed delivery, to allow it update the pheromone concentration of an entry $m$. The authors tested various formulas for calculating the new pheromone value, and the one they found to perform best was

$$
\begin{equation*}
\tau_{m}=e^{\psi \frac{\sigma_{m+1}}{\chi_{m}+\sigma_{m}+1}} \tag{2.8}
\end{equation*}
$$

where $\psi$ is a preset constant and $\sigma$ and $\chi$ are the count of the successful and failed transfers made using the entry $m$, respectively. After an intermediate node forwards the BCP it starts a time-out timer, which is only stopped after a BCP traversal acknowledgement is received from the receiving node. If no acknowledgement is received by the time the timer runs out, the resources reserved for the burst are released and a BCP-ACK informing the source node of a failure is sent to the source node, along intermediate nodes, allowing them to also free up reserved resources. This is to deal with the situation where a BCP is lost due to optical impairments.

### 2.6.6 Discussion

Due to the nature of ACO, the algorithms presented above are dynamic distributed routing algorithms. The first three algorithms (ACOR, DABR, and ACRWA) are node-by-node routing algorithms, while UCBRWA and FSAC are source routed. The designers of ACOR, ACRWA, FSAC and UCBRWA stated that they considered the WC constraint in the design of their algorithms, while DABR does not. Earlier algorithms (ACOR and DABR) focused on routing alone while they performed the wavelength assignment using other means. This is probably due to the fact that they were designed for fixed grid networks, hence they make the assumption that all wavelengths will be equivalent. However, this is not true for flexi-grid networks (Yu et al., 2013), and might negatively affect their performance on flexi-grid. Beginning with the ACRWA algorithm, there is a shift towards assigning wavelengths along with routes at the source node.

Intuitively, the node-by-node routing algorithms seem to be single path routing algorithms as only one route can be dominant at a time. While the source routed algorithms seem like they could be multi-path algorithms, due to the possibility that more than one entry could be dominant on the table. The operation of the source routed algorithms which requires response from previous transmissions to update the pheromone seems like it might create a scenario where a few entries dominate
and the algorithm switches between them. Further investigation on this will have to be performed.

Two network information inputs commonly used by the algorithms are the length of a route and some measure of the number of free wavelengths along a route. Every algorithm except for UCBRWA used at least one of these measures in some form. However, due to the authors not providing an analysis of why their algorithms work, it cannot be said what effects these inputs have on the performance of the algorithms. Possible benefits of minimising the route length are: minimising the power needed to make a transmission, minimising the effects of impairments, minimising the delay of the transmission and minimising the number of intermediate nodes used passed by transmissions, hence minimising the opportunities for blocking. The authors of ACOR and DABR state that they use the number of free wavelengths as a measure of the amount of congestion along a route. They state that this would allow the algorithms to favour least congested routes, hence reducing the blocking probability and balancing the load across the network.

It was observed that the authors of the node-by-node algorithms do not provide a means of calculating the offset between the BCP and burst as required by OBS. ACOR is not evaluated on an OBS network, hence this was not pertinent to its designer, and ACRWA gets around this issue by evaluating on an architecture where the need for an offset is eliminated. This increases the difficulty of implementing such algorithms, and reproducing the results obtained by the authors.

An issue noted in the evaluation procedure used for the FSAC algorithm, is the fact that the authors used a fixed transmission speed for all transmissions on the network. This neglects the multi-line rate nature of flex-spectrum transmission, which is a key aspect and benefit of flex-spectrum transmission. Thus the evaluation performed over flex-spectrum, reduces to evaluation performed over a fixed grid network (Yu et al., 2013). Study of the implementation of flex-spectrum used by Gravett et al. (2017) in their simulator, found that the assignment of spectrum to a transmission was done by randomly selecting a centre, and using the selected centre alone to define the wavelength used for transmission of a burst. That is, the beginning and end points of an assigned spectrum were neglected; hence the determination of the occurrence of blocking on the simulator is incorrect. The above two points given above, inevitably means the results obtained from the flex-spectrum evaluations presented in (Gravett et al., 2017) are invalid.

### 2.7 Conclusion

This chapter discussed important aspects related to the current research. First, optical networks were discussed, focusing on the need for, benefits, challenges and enabling technologies of transparent flexi-grid optical networks. Next, various aspects of OBS, and design issues that must be considered when designing OBS systems were discussed. Various simulation concepts were introduced, and an introduction to discrete event simulator frameworks which might be used to build an OBS simulator was given. The method which will be used to validate the operation of the implemented simulator was also discussed. Finally, ACO and various ACO based algorithms for OBS networks were presented and discussed.

## Chapter 3

## Simulator Design \& Validation

### 3.1 Introduction

In this chapter, the design and validation of a simulator based on the description of Optical Burst Switching given in Section 2.3 is presented. The simulator is designed and built using the process detailed in Section 1.4. The features required from the simulator are defined in Section 3.3. The simulation framework to be used is selected in Section 3.4, based on the review of simulation frameworks performed in Section 2.4.2. A brief description of the architecture of the selected simulation framework is given in Section 3.5. The design of the simulator is given in Section 3.6. An in-depth description of the implementation of the simulator is given in 3.7 and the validation process used to ensure that the simulator is an accurate representation of an OBS network, as well as the results obtained are given in Section 3.8.

Two key assumptions of this research are that flexi-grid is beneficial to the BLP of OBS networks, and that non-linear optical impairments play a substantial role in the performance of these networks. An experiment is performed in Section 3.9 to determine if this assumptions hold true.

### 3.2 Motivation

The research into the improvement of ACO RWA algorithms will be performed by simulating the operation of OBS networks using computers. This in order to determine the performance of routing algorithms designed for, and applied to OBS networks. However, there were no simulators available to the author which could
support the current work; simulators available to the author were the ad-hoc simulator previously used by the authors of (Gravett et al., 2017), and the simulator by Espina et al. (2012). The ad-hoc simulator was rejected because it is not validated, and does not allow generalisation of findings to real world networks due to the time steps of its simulation clock not being defined in terms of time in the real world. The simulator by Espina et al. (2012) was not validated and does not have the facility to simulate flexi-grid networks and optical impairments. Both were deemed to be inflexible to the needs of the current and future research. Hence, it was deemed necessary to design and build a simulator.

### 3.3 Problem formulation

The aim of this chapter, is to design and implement a computer program which is capable of simulating flex-spectrum optical burst switching networks; hence allowing the researcher to simulate an approximate real world behaviour under various conditions. In order to allow for real world applicability of the results obtained, the simulator should be based on realistic assumptions and incorporate optical impairments. The key performance metric from simulations will be the Burst Loss Probability (BLP), which is given by

$$
\begin{equation*}
B L P=\frac{\text { Number of lost burst }}{\text { Number of sent burst }} \tag{3.1}
\end{equation*}
$$

The simulator should be capable of simulating the generation and sending of bursts, simulate the transmission of generated traffic according to a routing algorithm, simulate the effects of impairments on aforementioned transmissions and be able to support various topologies and network configurations. Finally, the simulator should be simple to understand, flexible and extensible.

The simulator will consist of structures which model the behaviour of nodes in a network. Nodes will only have access to local information; they will not have access to information obtained globally, such as the global burst loss ratio, the set of all sent transmissions or knowledge of the current activities of other nodes. A reasonable assumption that will be made is that there is infinite electronic buffering and processing capability at each node where this will be required in real life. The reason for this is because the chief interest is in understanding the transmission behaviour of the network. Burst losses will be due to either wavelength contention or the effect of impairments on the control messages or burst.

### 3.4 Selection of Simulator Framework

From the three simulation frameworks discussed in Section 2.4.2, Omnet++ was chosen as the simulator framework to be used. This choice was due to its component based architecture, which will help in ensuring that the simulator is easily extensible; its extensive documentation and active community, which might ease the learning curve; and the extra functionality it offers which the others do not (such as the NED language, graphical user interface and data processing tools), which might improve the speed of development and data analysis.

### 3.5 Omnet++ Architecture

Omnet++ uses a component based architecture. This allows for easier development of a simulator as a model can be broken down into sub-models and implemented in separate modules. Modules are classified as either simple modules or compound modules. Simple modules are modules which hold the implementation of an aspect of a model. The operational logic of simple modules is written in $\mathrm{C}++$. Compound modules consist of nested simple modules which have been connected together in order to aggregate their behaviours. Simple and compound modules may further be connected together into a network, which allows the evaluation of the interaction between modules in the network. Figure 3.1 shows the relationship between simple and compound modules, as well as networks.

Modules have gates through which they may communicate with each other. Communication is done with messages which may contain arbitrary data. Messages are sent either through a connection between the gate of one module and another, or


Figure 3.1: Simple and compound modules ("OMNET++ Simulation Manual", 2018)
through a direct mechanism. A connection may not cross the boundary of a compound module; that is, a sub-module of a compound module cannot be connected to a module outside the compound module. Connections between modules have a channel object associated with them. Channels allow the modelling of the various characteristics of physical links, such as data rate, propagation delay and bit error rate.

Statistics can be collected at each simple module through the C++ code written for the module. However, this is not convenient for global statistics collection in a compound module or network. Hence Omnet++ provides a mechanism known as signals which modules may listen for. A simple module may emit a signal with some information. The emitted signal will propagate to all compound modules which enclose it, with only the modules which are listening for the emitted signal acting on it.

Properties of modules, networks, messages, and channels can be specified using the frameworks NED language. NED allows the definition of the parameters and gates of modules, the sub-modules and connections in a compound module or network, and the fields of messages and channels.

### 3.6 Simulator Design

The simulator's design was based on the design of an OBS simulator by Espina et al. (2012). However, the design has been extensively modified and re-implemented to allow better flexibility and extensibility. Major changes include the implementation of a $K$-shortest path algorithm to aid the study of RWA algorithms, and the encapsulation of the physical aspects of the network into a single module (the Fibre module) leading to better modelling of WDM networks, better segregation of roles among all the modules and the removal of redundant modules. This section gives a conceptual overview of the design of the simulator. Components of the simulators design are emphasised using italics.

The simulator consists of three main modules; the Edge node, Core node and Fibre. Each module encapsulates a given aspect of the operation of an OBS network. Interaction between modules is mainly via message passing; ensuring that each module acts independently. In a few cases where this is considered appropriate, modules also interact by calling methods in other modules, in order to collect information or perform an action on that module.

Edge nodes are the source and destination of bursts, and are responsible for the aggregation of packets into burst, routing of the burst, and the sending of the BCP and burst at the appropriate times (based on wavelength availability and burst offset). The chief responsibility of Core nodes is the forwarding of messages from Edge nodes and other Core nodes towards their final destination. However, if the routing algorithm of interest is distributed, it may also be implemented here. The implementation of routing algorithms usually requires access to a list of shortest paths between a node and other edge node, hence every node is equipped with a list of $K$-shortest paths between the node and every Edge node (where $K \in \mathbb{N}$ ). The shortest paths are calculated at the start of the simulation using Yen's $K$-shortest path algorithm (Yen, 1971).

The Fibre module is meant to encapsulate all the details of sending data over a physical fibre link; such as the reservation of a light path, the transmission and subsequent traversal of data over a fibre link and the application of impairments and other physical layer phenomenon such as amplification and wavelength conversion. Separating the data transmission aspect from the nodes, allows for provision of a method to aid calculation of impairments; and independence of the algorithms implemented on the node from the physical layer design of the network, leading to improved extensibility and re-usability of modules.

After a message is sent from a node, the message then traverses the network along a Fibre. If the message is a BCP, an attempt is made to reserve the wavelength channel assigned to its corresponding burst before it is sent to the next node. The BCP is discarded if the reservation fails. If the reservation at a Fibre fails, the corresponding burst will be lost at the Fibre where reservation failed. The network can support sending bursts using fixed grid or flexi-grid. In the fixed grid scheme, the channel bandwidth spacing is usually set to 50 or 100 GHz , and a burst is sent on a single channel, while in the flexi-grid scheme, the channel bandwidth spacing is set to 12.5 GHz , and a burst may be sent on an arbitrarily sized bandwidth channel, made up of the concatenation of a number of the 12.5 GHz channels. In order to assign an appropriate width of flexi-grid channels to a transmission (assuming On-Off keying modulation), the following relationship is used (Merrifield \& Haran, 2013)

$$
\begin{equation*}
\Delta F * \Delta T=1 \tag{3.2}
\end{equation*}
$$

where $\Delta F$ is the channel bandwidth, and $\Delta T$ is the bit duration.
Immediately after a message traverses a Fibre, the loss of power (in dB) experienced by a transmitted signal across a length of fibre due to linear and non-linear


Figure 3.2: Usage of simulator modules in building a network
impairments is calculated using equation (2.1).
Statistics such as the total number of burst sent, number of BCP's and burst lost, and number of successful burst are collected globally. The collected data is used to calculate the BLP of the network. Data is also collected locally using counters at each simple module. By comparing the data obtained from global and local counters, developers can validate the operation of the simulator to ensure that all messages are being handled as required.

### 3.7 Simulator Implementation

The implemented simulator currently supports JIT scheduling, all three forms of burst assembly mentioned in Section 2.3, modelling fixed and flexi-grid DWDM networks, and networks where the WC constraint is imposed. The design of the simulator presented in Section 3.6 was reviewed with stakeholders (research supervisors and other students in the CBC) before implementation began.

As stated in Section 3.6, the design of the simulator consist of three main modules; the Edge node, Core node and Fibre. This section presents the implementation of the simulator design presented in Section 3.6. Figure 3.2 shows an Omnet++ generated image of a network built using all the modules described above. It consist of two Edge nodes at either end, connected by three paths, which have Core nodes


Figure 3.3: Edge Node components
and Fibre modules along them. Also visible in the figure, at the top right hand corner, is the network configuration module which is responsible for assigning a unique number to each Edge node, to act as the address of a node; and initially calculating the $K$-shortest paths between nodes, and filling the tables of nodes with the paths.

In this section, detailed descriptions of the implementation of each module is given. Omnet++ generated images showing the structure of modules are provided.

### 3.7.1 Edge Node

The Edge node (Figure 3.3) is implemented as a router with an OBS network interface, which may then be connected to a packet generator (the packet generator


Figure 3.4: Core Node module layout
currently employed is obtained from the INET framework ("INET Framework", 2018)). The Edge node also has a $K$-shortest path table which is populated with the $K$-shortest paths between the current node and every other edge node within the simulated network.

The OBS interface consists of an assembler, disassembler and routing table. The assembler consist of a dispatcher which sends incoming packets to the appropriate packet burstifier depending on its destination. When a packet burstifier satisfies the conditions for burst generation, it sends the burst to the router module, which generates the BCP, performs RWA (possibly using information from the routing table), calculates the offset between the BCP and Burst, and finally sends them off towards their destination. The routing table can be extended to fit the routing algorithm being modelled.

The disassembler, is the termination of every messages journey. The main job of the disassembler is to receive messages destined for the edge node, and record statistics about them. However, it may be extended to fulfil other tasks in an algorithm, such as updating the routing table or initiating the sending of an acknowledgement message by the router.


Figure 3.5: Fibre module layout

### 3.7.2 Core Node

The implementation of the Core node (Figure 3.4) consists of five modules, the input module, core control logic module, optical cross connect module, $K$-shortest path table and a routing table. The input module determines if the message received is a control message or burst and forwards the message either to the optical cross connect or core control logic respectively. The core control logic deals with creating a reservation for the appropriate output port for the message. The optical cross connect component is meant to model the operation of an optical cross connect, switching messages to the right output port as has been reserved. The core node routing table and $K$-shortest path table are similar to their counter parts in the edge node (except that the $K$-shortest path table consists of $K$ paths between a core node and every edge node) and may be used when implementing a routing algorithm at the core node.

### 3.7.3 Fibre

The implementation of the Fibre module (Figure 3.5) consist of three components; two mux components representing either end of a fibre link, and a Fibre table which records the transmission time and the optical power of transmissions, for use in the calculation of physical layer impairments. When sending messages, burst are only transmitted once at the first mux component they encounter (in order to simulate a transparent optical network), while control messages (BCP's and acknowledgement
packets) are re-transmitted, whenever they are being sent between two mux components (in order to simulate the OEO conversion undergone by control messages on OBS networks).

When a message arrives at either end of the Fibre to be sent to the other end, the following happens; if the message is a control message and the dedicated control channel is free, it is sent over the dedicated control channel and then sent on to the next node if the message is readable; else if the dedicated control channel is not free, the control message is queued. If the message is a BCP, the wavelength required by its burst is reserved if possible; the BCP is then sent to the other side. If the BCP is unreadable on the other side of the fibre or the wavelength reservation is unsuccessful on the other side, the reservation is not accepted and the BCP is discarded. If the message received is a burst and the required reservation has been made, it is sent on to the other end of the fibre; else the burst is lost.

On sending a message, the sending side records an entry in the fibre table module; citing the start time, end time, power and spectrum of the message it sent. On the receiving side, when a message is received, all entries between the start and end time of the message's transmission requested of the fibre table module, and used to calculate the impairment loss suffered by the received message. If the message's optical power is less than the set receiver sensitivity, and it is a control message, then it is discarded; else, if the message is a burst, it is marked as unreadable and sent on to the next node (due to the transparency of the network). The receiver sensitivity of the last multiplexer encountered by a burst, is what determines if a burst is considered readable or unreadable due to optical impairments.

### 3.7.4 Super Node

Aside from the components stated above, the simulator also has a component called a super node which consists of an Edge node and Core node put together. This structure was created in order to improve the ease with which mesh networks can be built using the simulator. It allows data transfer from the Edge node into the network, while still allowing the Core node to fulfil its forwarding duties. Figure 3.6 shows the layout of a super node, and Figure 3.7 shows an example of a network built using super nodes.


Figure 3.6: SuperNode module layout


Figure 3.7: Network built using super nodes

### 3.8 OBS Validation

Validation of the simulator consisted of verifying that the simulator was operating according to the conceptual model of the system described in Section 3.6, by examining event logs and the operation of the simulator to ensure that there were no programming errors present in the simulator and ensuring that certain key counters were as expected.

The simulator was validated by comparing simulation results against analytical results obtained by using the Reduced Link Load approximation model for OBS networks described in Section 2.4.3; in order to determine if the simulator is an accurate representation of the model.

### 3.8.1 Validation Procedure

Burst were sent on the 7 node network shown in Figure 3.8, along the routes given in the image. The WC constraint was imposed, with the wavelength used for each burst selected randomly, and optical impairments disabled. The scheduling scheme used was the JIT protocol. The transmission speeds used for burst and control channels was 10Gbps.

A configuration for a simulation consisted of the load offered to the network, and the number of channels available on the network. Packets were generated using exponentially distributed inter-arrival times (this is the packet distribution assumed by the reduced link load model described in Section 2.4.3), with the mean of the distribution set to $2 \mathrm{~ms}, 1.5 \mathrm{~ms}, 1 \mathrm{~ms}$ and 0.5 ms ; for a total of four load scenarios


Figure 3.8: Topology used in validation, with routes used in validation


Figure 3.9: Graphs showing the average burst loss probability vs load, for the simulations and approximations
(increasing the mean inter-arrival time decreases the load). The size of the generated packets were exponentially distributed with the mean set at 1.5 kB . The burst assembly scheme used was size based, with the maximum burst size being 15 kB . Each load scenario was tested on the network with 2, 4 and 6 burst transmission channels available. The combination of each load and network was run 30 times. The simulations were run for a simulation time of 200s in order to ensure that steady state was reached.

A Matlab script was written which applies the reduced link load model described in Section 2.4.3 to the network shown in Figure 3.8. The mean of the results obtained from the simulator are compared with the results obtained from the Matlab script.

### 3.8.2 Validation Results

The burst loss ratio obtained for the various scenarios are presented in Figure 3.9. From the graphs, it can be seen that the burst loss probability decreases with decreasing load and increasing number of transmission channels; which is consistent with expectations. It can also be clearly observed that the simulated results (legend entries that end in '-sim') closely match the approximated results (legend entries that
end in '-approx'). Hence, it can be concluded that the design and implementation of the simulator provides a good representation of an OBS network.

### 3.9 Determining the Effects of Flexi-grid and Impairments on Network Performance

### 3.9.1 Experimental Procedure

Simulations were run on the 14 node network given in Figure 3.10, with the WC constraint imposed, in order to determine the effects of flexi-grid and optical impairments on simulation results obtained. The wavelength used for each burst was selected randomly, and the scheduling scheme used was the JIT protocol. Bursts were sent between every edge node pair, using the shortest paths between node pairs.

Packets were generated using exponentially distributed inter-arrival times, with the mean of the distribution set to $50 \mu \mathrm{~s}, 40 \mu \mathrm{~s}, 30 \mu \mathrm{~s}, 20 \mu \mathrm{~s}$ and $10 \mu \mathrm{~s}$; for a total of five load scenarios (increasing the mean inter arrival time, decreases the load). The size of the generated packets were exponentially distributed with the mean set at 1.5 kB . The burst assembly scheme used was size based, with the maximum burst size being 15 kB . Each loading scenario was applied to the network with the optical impairments enabled and disabled, using 16, 50 GHz fixed grid burst transmission channels, and $64,12.5 \mathrm{GHz}$ flexi-grid burst transmission channels. The simulations were run 30 times, with different random seeds, in order to ensure statistically valid samples. Each experiment was run for a simulation time of a 200s in order to ensure that steady state was reached.

Each burst was sent on a channel with a randomly assigned transmission speed, such that a specified percentage of all burst sent over the network traversed the network at a given speed. This is was done in order to simulate the multi-class nature of flexi-grid networks. The percentage of burst that were transmitted at a given transmission speed are,

- $40 \%$ of traffic transmitted at 10 Gbps
- $30 \%$ of traffic transmitted at 20 Gbps
- $20 \%$ of traffic transmitted at 30 Gbps
- $10 \%$ of traffic transmitted at 40Gbps.


Figure 3.10: NSFNET Topology (Gravett, du Plessis, \& Gibbon, 2017). Link lengths in km .

The assumption being made is that in a real network, few users require and pay for higher transmission speeds. The transmission speeds of all control channels will be set to 40 Gbps .

### 3.9.2 Results and Analysis

The burst loss ratio obtained for the various scenarios are presented in Table 3.1 and Figure 3.11. From the graphs, three major observation can be made. Firstly, the burst loss probability for the fixed grid scenarios are higher than those of the flexigrid scenarios. This confirms the assertions by Wright et al. (2013) and Gerstel et al. (2012) that flexi-grid improves bandwidth utilisation of the network, in comparison to fixed grid.

Secondly, the difference between the burst loss probability values of the scenarios where impairments are applied and where they are not, are higher for the flexi-grid scenarios (an average difference of $4.5 \%$ ) than those of the fixed grid scenarios (an average difference of $3.1 \%$ ). This implies that optical impairments have a larger impact on flexi-grid transmission than they do fixed-grid. This is consistent with expectations, due to the fact that, fixed grid channel widths were designed with ample guard-bands to limit the effect of impairments (Wright et al., 2013; Gerstel et al., 2012), and the number of fixed grid channels present in a network is always constant, unlike in the flexi-grid scenarios.

Thirdly, the effects of impairments increase the higher the load for the flexi-grid

Table 3.1: Burst Loss Probabilities obtained for various scenarios

| Grid type | Impairments | Load $(\mu s)$ | Burst Loss |
| :---: | :---: | :---: | :---: |
|  |  | 50 | 0.2825 |
|  |  | 40 | 0.33186 |
|  | Enabled | 30 | 0.40209 |
| Fixed |  | 20 | 0.50995 |
|  |  | 10 | 0.69779 |
|  |  | 50 | 0.25286 |
|  | Disabled | 40 | 0.29959 |
|  |  | 30 | 0.36726 |
|  |  | 20 | 0.47475 |
|  |  | 10 | 0.67442 |
|  |  | 50 | 0.2591 |
| Flexi |  | 40 | 0.29956 |
|  |  | 30 | 0.3565 |
|  |  | 20 | 0.44329 |
|  |  | 10 | 0.59352 |
|  | 50 | 0.22532 |  |
|  |  | Disabled | 30 |
|  |  | 0.26126 |  |
|  |  | 20 | 0.31251 |
|  |  | 10 | 0.39193 |
|  |  |  |  |

scenarios, while it stays relatively constant for the fixed grid scenarios. For instance, the difference between the burst loss probability, when impairments are enabled and when they are not for the fixed grid at load $10 \mu s$ and $50 \mu s$ are $2.33 \%$ and $2.96 \%$, respectively; while for flexi-grid they are $5.9 \%$ and $3.38 \%$, respectively. This implies that, while it may be safe to discount impairments for fixed grid transmission, they cannot be neglected in flexi-grid scenarios, as the effect of impairments cannot be quantified by a constant burst loss probability. Once again, this could be attributed to the guard-bands built into fixed grid channels and the static nature of fixed-grid. All three observations further assert the validity of the simulator, as its behaviour is consistent with the literature.


Figure 3.11: Graphs showing the average burst loss probability vs load, for various fixed and flexi-grid scenarios

### 3.10 Conclusion

This chapter detailed the design of the simulator as well as the validation process which was used to ensure that it is an accurate representation of the operation of an OBS network. The activities of this chapter allow us to pursue research using the simulator with confidence.

The effects of flexi-grid and impairments on network performance are also determined; it was found that flexi-grid provides a substantial improvement in network performance over fixed-grid; however, it was also discovered that impairments have a greater, non-constant effect on transmissions made on flexi-grid networks than on fixed grid networks. Hence it is especially important that impairments are modelled when studying flexi-grid networks.

## Chapter 4

## Initial Investigations

### 4.1 Introduction

The previous chapter presented the design and validation of a new OBS simulator, capable of simulating OBS on flexi-grid networks with the WC constraint imposed. The availability of such a tool creates the opportunity to accurately study and compare various RWA algorithms and determine appropriate parameters which might be applicable to the real world.

The aim of this chapter, is to study the performance of current algorithms on flexigrid OBS networks, and select an algorithm which might be used as the basis for future investigation. The algorithm selected for future development will be the candidate algorithm that performs best in performance evaluations. However, before the algorithms can be compared against each other a parameter study needs to be performed on the candidate algorithms to determine which algorithm parameters will yield the best performance for the scenarios considered. The data obtained from the parameter study process provides an opportunity to study the effects of the parameters on the performance of the algorithms. Studying this will provide insight into the operation of the algorithms, which might reveal opportunities for improvement.

Two ACO algorithms were selected for this study, namely, ACOR and FSAC. These were presented in Section 2.6 .1 and 2.6.5, respectively. They were selected because they represent the two distinct approaches of implementing ACO over optical networks; that is in a node-by-node manner and source routed manner. They were also selected because they easily lent themselves to modification for use over flexi-grid OBS networks with WC imposed.

Details about the implementation of the algorithms are presented in Section 4.2. The Network scenarios on which the algorithms will be evaluated are presented in Section 4.3. The procedure used for the parameter study and the results obtained are presented in Section 4.4. The procedure used for the performance comparison and the results obtained are presented in Section 4.5. Finally, a conclusion is given in Section 4.7.

### 4.2 Adaptation \& Implementation of Algorithms

As stated in Section 2.6.6, ACOR does not provide a method for offset calculation. Hence, in order to determine what the offset should be, the ants count the number of core nodes on the route they take, and record this at the destination edge node. The average number of core nodes towards the source node, reported by the ants for a given time period ( 0.5 s was the time period used) is then calculated and used to calculate the offset for transmissions towards the source node of the ants. Ngo et al. (2006) used first-fit for wavelength assignment in their evaluation of ACOR on OCS. However, Teng and Rouskas (2005) demonstrated that on OBS networks, first-fit has a worse performance than random wavelength assignment; hence random wavelength assignment was selected as the method of wavelength assignment to be used with ACOR.

FSAC was implemented as described in Section 2.6.5. The pheromone values for initial entries and entries created in the creation phase were randomly generated between 1 and 1.5; in order to encourage exploration at the start. This was found to provide the best performance in preliminary test; as opposed to setting all pheromones to 1 as performed by Gravett et al. (2017).

Wavelength assignment for both algorithms was done using the elastic method with spectrum expansion/reduction (as shown in Section 2.3.3). For ACOR, a 12.5 GHz slot is randomly chosen to be the wavelength centre for the assigned wavelength. If the slot is not adequate for the transmission that needs to be made (checked using equation (3.2)), the node tries to expand the spectrum around the initially selected centre. If the spectrum cannot be expanded to the required width, using the current selected centre, then another 12.5 GHz slot is randomly chosen to be the wavelength centre. This process continues until the required amount of spectrum has been assigned, or all free slots have been considered. For FSAC, a randomly selected centre is used when a table entry is being created; the wavelength will be expanded around this slot in order to accommodate transmissions with high


Figure 4.1: SIMPLE Topology (Gravett, du Plessis, \& Gibbon, 2017). Link lengths in km .
transmission speeds.

### 4.3 Network Scenarios

Simulations will be performed using two network topologies; the 14 node NSFNET topology shown in Figure 3.10 and the 6 node SIMPLE topology shown in Figure 4.1. These two networks were chosen because they were found to be used widely by researchers in networking fields. Performing experiments on both allows the evaluation of the algorithms in a relatively simple scenario (SIMPLE topology), and in a larger, more complex scenario (NSFNET topology).

Simulations will be performed with impairment free control channels. The assumption being made is that a network operator could provide an impairment free control channel (such as in the form of a separate fibre). All simulations are performed with impairments enabled for data channels and the WC constraint imposed. The scheduling scheme used was the JIT protocol.

For the purpose of this study, three load categories are defined; high, medium and low load. These are measured using the mean Burst Loss Probability (BLP) obtained by the Shortest Path routing Algorithm (SPR) with random wavelength selection (the combination of SPR with random wavelength selection is hereby referred to as GSPR in discussions and results). High load scenarios are network scenarios where GSPR achieves a BLP greater than $30 \%$, medium load as a BLP greater than $10 \%$ but less than or equal to $30 \%$; and finally, low load as a BLP less than or equal to $10 \%$.

In order to simulate the multi-line rate nature of flexi-grid networks, each burst will be sent on a channel with a randomly assigned transmission speed, such that a specified percentage of all burst sent over the network traverse the network at a given speed. The percentage of burst that were transmitted at a given transmission speed are

- $40 \%$ of traffic transmitted at 10 Gbps
- $30 \%$ of traffic transmitted at 20 Gbps
- $20 \%$ of traffic transmitted at 30 Gbps
- $10 \%$ of traffic transmitted at 40 Gbps .

The assumption being made is that in a real network, few users require and pay for higher transmission speeds. The transmission speeds of all control channels will be set to 40 Gbps .

### 4.4 Parameter Study

In this section, parameter studies will be performed for ACOR and FSAC in order to determine what significant linear relationships exist between the algorithm's performance and its various parameters, as well as to determine what parameters yield an optimal performance. This is expected to give insight into the operation of each algorithm.

The experimental procedure used for the parameter study is presented, followed by an analysis of the results obtained for each algorithm.

### 4.4.1 Experimental Procedure

The parameter studies for all algorithms will be performed on both the SIMPLE and NSFNET topologies on all load levels defined in Section 4.3. Multiple values for each parameter of an algorithm were selected; the parameter values were then combined to create a unique algorithm configuration in the parameter study. Each configuration (a topology, a load, an algorithm, a set of parameter values) will only be run once, as the large amount of configurations obtained by combining the parameters prohibits repeating simulations for configurations due to resource limitations. The average BLP for each second, over the duration of a run will be collected.

The parameter values used for each variable, for ACOR and FSAC, are given in Table 4.1 and 4.2 , respectively. The number of runs per topology is obtained by multiplying the number of values for each relevant parameter by the number of loads which will be used (3 load levels).

Table 4.1: Parameter values used in parameter study of ACOR

| Parameter | Values |
| :---: | :---: |
| $v$ | $0.3,0.5,0.7$ |
| $\delta$ | $0.1,0.55,1,1.45$ |
| $\epsilon$ | $0.1,0.55,1$ |
| $g$ | $0.05,0.1,0.2,0.4$ |
| ant send probability | $0.2,0.4,0.6$ |
| ant send interval (ms) | $25,75,125$ |
| Number of Runs per topology | 3888 |

Table 4.2: Parameter values used in parameter study of FSAC

| Parameter | Values |
| :---: | :---: |
| $\alpha_{1}$ | $0.3,0.5,0.9$ |
| $\alpha_{2}$ | $\left(1-\alpha_{1}\right)^{*} 0.3,\left(1-\alpha_{1}\right) * 0.5,\left(1-\alpha_{1}\right) * 0.9$ |
| $\beta$ | $0,0.2,0.4,0.6,0.8,1,1.2,1.4$ |
| $\psi$ | $0.01,0.1,0.28,0.64,1,1.36$ |
| Number of Entries | $4,8,12$ |
| Number of Runs per topology | 3888 |

Simulations on the NSFNET topology are run with 200 GHz spectrum width (16 x 12.5 GHz flexi-grid channels) available on the network; and 100 GHz spectrum width ( $8 \times 12.5 \mathrm{GHz}$ flexi-grid channels) on the SIMPLE topology. Packets are generated using exponentially distributed inter-arrival times with the mean of the distribution set as specified in Table 4.3 (increasing the mean inter arrival time, decreases the load). The size of the generated packets were exponentially distributed with the mean set at 1.5 kB . The burst assembly scheme used was size based, with the maximum burst size being 15 kB .

The simulations will all be run with the same seed in a bid to minimise the variability due to random effects. Pearson correlation coefficients ${ }^{1}$ between the obtained BLP at various points of interest and the parameters of the algorithms will then be

[^1]Table 4.3: Mean inter-arrival time values used to achieve various loads in parameter studies

| Topology | High (ms) | Medium (ms) | Low (ms) |
| :---: | :---: | :---: | :---: |
| SIMPLE | 0.04 | 0.11 | 0.7 |
| NSFNET | 0.09 | 0.2 | 1.2 |

calculated for the various load configurations. Negative correlations indicate that an increase in the value of a parameter leads to a decrease in the BLP achieved by an algorithm. While positive correlations indicate that an increase in the value of a parameter leads to an increase in the BLP achieved by an algorithm. The absolute value of a correlation indicates its strength. The scale that will be used for evaluating the strength of the correlations is given below:

- 0 to 0.1 - very weak relationship
- 0.1 to 0.3 - weak relationship
- 0.3 to 0.7 - moderate relationship
- 0.7 to 0.9 - strong relationship
- 0.9 to 1 - very strong relationship

No instance of the use of a similar scale to interpret correlation coefficients could be found in the computer science discipline. Hence, the scale given above is based on scales for interpreting Pearson correlation coefficients used by researchers in chemistry (Asuero, Sayago, \& Gonzalez, 2006), medicine (Taylor, 1990) and the behavioural sciences (Cohen, 1988).

Statistically significant results and patterns observed for each algorithm will then be discussed. The statistical significance test of the correlations, gives assurance that the effects observed are not due to random processes. All significant results are significant at $p<0.05$. Scatter plots of the data collected are provided in Appendix D to allow for verification of the presented correlation values.

### 4.4.2 Results and Discussion

Every simulation except for those for FSAC at high load, on the SIMPLE network, was able to achieve a simulation time of 500 s within the time limit allowed for a run. A time limit was required due to resource limitations.

The Pearson correlation coefficients obtained from analysing the obtained data at
various seconds over the course of the simulation are presented in Tables 4.4 to 4.7. Statistically significant patterns observed for each algorithm are discussed in the following sections.

### 4.4.2.1 ACOR

The following is observed from Tables 4.4 and 4.5.

- $v$ determines the influence of $\Lambda$ (function of the path length) and $X$ (function of the percentage of free wavelengths). The correlation values observed for $v$ are observed to not be significant for the period of the full run. However, they tend to be significant for times well into the duration of the run. For the SIMPLE topology, $v$ is observed to be very weakly negatively correlated. While for the NSFNET topology $v$ is observed to be very weakly positively correlated at high and medium load. The strength of the correlations increase with time. At low load for the NSFNET topology, $v$ is weakly negatively correlated at 100s, and not significantly correlated for the rest of the time. The negative correlation indicates that increasing the influence of $\Lambda$, by increasing the value of $v$, minimises the BLP. While the positive correlation indicates that increasing the influence of $X$, by increasing the value of $v$, minimises the BLP. This indicates that for the NSFNET topology taking the shortest route might not be the best method to minimise the BLP.
- $\delta$ determines the influence of the path length on the algorithm. For the SIMPLE topology, correlation values observed for $\delta$ are observed to not be significant for the full period of the runs. The correlation values are observed to be significant for times well into the duration of the run. $\delta$ is observed to be weakly to moderately negatively correlated with the strength of the correlation increasing with time.

For the NSFNET topology, correlation values observed for $\delta$ are observed to not be significant for the full period of the runs at high load. However, at medium and low load, the correlations are significant for the full period of the runs. $\delta$ is observed to be weakly positively correlated in the cases where it is observed to be significant. The strength of the correlations increase with time.

- $\epsilon$ determines the influence of the estimated free wavelengths along a path. $\epsilon$ is found to only be correlated at a single point, at High load on the NSFNET topology(the correlation is very weakly negatively correlated), and for most of the period of the run at High load on the SIMPLE topology (where it is
Table 4.4: Correlation coefficients for ACOR at various simulation times on the SIMPLE topology. Significant values given in bold.

|  | High Load |  |  |  |  |  | Medium Load |  |  |  |  |  | Low Load |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | 3s | 100s | 200s | 300s | 400s | 500s | 3s | 100s | 200s | 300s | 400s | 500s | 3s | 100s | 200s | 300s | 400s | 500s |
| $v$ | 0.00 | -0.04 | -0.06 | -0.07 | -0.08 | -0.09 | -0.01 | -0.03 | -0.07 | -0.07 | -0.08 | -0.09 | -0.03 | -0.11 | -0.12 | -0.14 | -0.15 | -0.16 |
| $\delta$ | -0.04 | -0.25 | -0.36 | -0.40 | -0.43 | -0.44 | -0.01 | -0.16 | -0.31 | -0.40 | -0.45 | -0.48 | 0.01 | 0.04 | 0.00 | -0.05 | -0.10 | -0.16 |
| $\epsilon$ | 0.00 | 0.07 | 0.07 | 0.07 | 0.07 | 0.07 | -0.01 | -0.01 | -0.01 | 0.00 | 0.01 | 0.01 | 0.00 | 0.01 | 0.01 | 0.01 | 0.01 | 0.00 |
| $g$ | 0.05 | 0.40 | 0.55 | 0.60 | 0.62 | 0.64 | 0.03 | 0.10 | 0.23 | 0.29 | 0.32 | 0.34 | -0.05 | -0.18 | -0.15 | -0.12 | -0.10 | -0.07 |
| ant send probability | -0.54 | -0.37 | -0.25 | -0.17 | -0.13 | -0.11 | -0.58 | -0.37 | -0.26 | -0.18 | -0.13 | -0.09 | -0.54 | -0.46 | -0.44 | -0.44 | -0.43 | -0.41 |
| ant send interval | 0.76 | 0.45 | 0.30 | 0.21 | 0.16 | 0.13 | 0.70 | 0.46 | 0.36 | 0.28 | 0.23 | 0.20 | 0.67 | 0.41 | 0.39 | 0.39 | 0.39 | 0.38 |

Table 4.5: Correlation coefficients for ACOR at various simulation times on the NSFNET topology. Significant values given in bold.

|  | High Load |  |  |  |  |  | Medium Load |  |  |  |  |  | Low Load |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | 3 s | 100s | 200s | 300s | 400s | 500s | 3s | 100s | 200s | 300s | 400s | 500s | 3s | 100s | 200s | 300s | 400s | 500s |
| $v$ | -0.03 | 0.02 | 0.06 | 0.07 | 0.08 | 0.09 | -0.03 | 0.00 | 0.03 | 0.05 | 0.06 | 0.07 | -0.05 | -0.06 | -0.02 | 0.01 | 0.03 | 0.04 |
| $\delta$ | 0.03 | 0.14 | 0.23 | 0.28 | 0.31 | 0.33 | 0.06 | 0.19 | 0.26 | 0.29 | 0.30 | 0.30 | 0.05 | 0.21 | 0.27 | 0.29 | 0.29 | 0.30 |
| $\epsilon$ | -0.01 | -0.06 | -0.05 | -0.05 | -0.05 | -0.04 | -0.01 | -0.03 | -0.03 | -0.02 | -0.02 | -0.02 | 0.00 | 0.01 | 0.01 | 0.01 | 0.01 | 0.01 |
| $g$ | 0.00 | 0.34 | 0.53 | 0.63 | 0.69 | 0.71 | -0.03 | 0.29 | 0.52 | 0.65 | 0.71 | 0.74 | -0.05 | 0.19 | 0.48 | 0.65 | 0.75 | 0.80 |
| ant send probability | -0.52 | -0.51 | -0.38 | -0.28 | -0.21 | -0.17 | -0.52 | -0.51 | -0.40 | -0.30 | -0.23 | -0.17 | -0.50 | -0.52 | -0.43 | -0.34 | -0.27 | -0.22 |
| ant send interval | 0.80 | 0.59 | 0.44 | 0.32 | 0.25 | 0.19 | 0.79 | 0.63 | 0.49 | 0.37 | 0.28 | 0.22 | 0.80 | 0.65 | 0.53 | 0.42 | 0.33 | 0.27 |

very weakly positively correlated, and constant). The general non-significance and weakness of $\epsilon$ seems to indicate that the measurement of percentage of free wavelengths has little effect on the performance of the algorithm. The observations made contradicts observations made about $v$ on the NSFNET topology.

- The exploration probability, $g$, is observed to mostly range from weakly positively correlated to moderately positively correlated. The strength of the correlation is observed to increase with time. The exception to this is on the SIMPLE topology at low loads, where the significant correlations are found to be weakly negative. The strength of this observation is found to decrease with time. This may be, because at low loads on the SIMPLE topology, all routes are initially as viable as the others. But as the bursts start concentrating along various routes, due to the increasing pheromone concentrations, this becomes less so. This indicates that in general, exploration decreases the performance of the algorithm.
- The ant send probability ranges from weakly negatively correlated to moderately negatively correlated for both topologies, loads and all times. This indicates that increasing the level of network monitoring improves the performance of the algorithm. Also, due to the manner in which pheromones are maintained, a high ant send probability might be required to ensure the dominance of the best path. The strength of the correlation decreases with time.
- The correlation of the ant send interval is found to weaken with time from strongly to weakly positively correlated. The strength of the correlation declines over time to become weakly positively correlated. This indicates that the ant send interval should be minimised to obtain better BLP values. Also indicates, that the effects of ants on the performance of the algorithm decreases with time.

From the observations made above, a trend that was observed, is that the algorithm tends to favour exploitation over exploration, as can be observed from the positive correlation coefficients obtained for the ant send interval and the exploration probability $g$. For the SIMPLE topology, favouring the shortest path is sufficient to improve the performance of the algorithm, while for the NSFNET topology this strategy is detrimental to the algorithms performance. $\epsilon$ was generally not statistically significant, implying that the estimation of the percentage of free wavelengths had no effect. However, $\delta$ which controls the effect of the path length often had a
statistically significant effect, though it was not always to the benefit of the algorithm.

### 4.4.2.2 FSAC

The following is observed from Tables 4.6 and 4.7.

- The number of entries in a pheromone table for a given destination is weakly negatively correlated for all times and configurations investigated. This could be due to the fact that the entries are not unique in FSAC, hence increasing the number of entries, increases the probability of encountering a better entry when exploring. The strength of the correlation slightly decreases with time.
- $\beta$ determines the influence of the path length on the algorithm. $\beta$ has a significant negative correlation with the BLP obtained at the beginning of all simulation. The strength of this correlation increases with decreasing load. There is no significant relationship between the value of $\beta$ and the obtained BLP as the simulation continues, except at low loads on the NSFNET topology, where the correlation is significant (albeit decreasing in strength) for the entire period of the run. Possible reasons for this might be that at the beginning of a simulation a high $\beta$ value allows the algorithm to rapidly approach the performance of SPR on the network.
- $\alpha_{1}$ controls the degree of exploitation done by the algorithm. $\alpha_{1}$ is strongly negatively correlated for all times and configurations investigated. This implies that the algorithm is a greedy algorithm.
- $\alpha_{2}$ controls the degree of exploration done by the algorithm. $\alpha_{2}$ is moderately positively correlated at the beginning of simulations, and becomes weakly positively correlated with time. This observation combined with those made about $\beta$ and $\alpha_{1}$ imply that the FSAC algorithm is a greedy algorithm (that is it obtains better outcomes by exploiting locally optimal entries than by exploring other entries).
- $\psi$ scales the maximum possible pheromone achievable by an entry. $\psi$ is observed to be very weakly negatively correlated for all times at high and medium load on the SIMPLE topology, and at high load on the NSFNET topology. $\psi$ is observed to be very weakly positively correlated on the NSFNET topology at low load, for times well into the duration of the simulation. A consistent pattern could not be observed with the parameter, indicating that it has a
non-linear relationship to the performance of the algorithm, or its effect is dependent on that of another variable.

From the observations made above, it can be said that the algorithm favours exploitation over exploration, this is observed from the correlation coefficients obtained for $\alpha_{1}$ and $\alpha_{2}$. It is also observed that aside from at the beginning of a simulation, the path length has very little effect on the performance of the algorithm.
Table 4.6: Correlation coefficients for FSAC at various simulation times on the SIMPLE topology. Significant values given in bold.

|  | High Load |  |  | Medium Load |  |  |  |  |  | Low Load |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | 3 s | 100s | 200s | 3 s | 100s | 200s | 300s | 400s | 500s | 3s | 100s | 200s | 300s | 400s | 500s |
| Number of Entries | -0.19 | -0.16 | -0.17 | -0.18 | -0.15 | -0.15 | -0.16 | -0.16 | -0.16 | -0.19 | -0.19 | -0.18 | -0.18 | -0.18 | -0.18 |
| $\beta$ | -0.09 | 0.00 | 0.00 | -0.17 | 0.02 | 0.02 | 0.02 | 0.02 | 0.02 | -0.42 | -0.03 | -0.01 | -0.01 | 0.00 | 0.00 |
| $\alpha_{1}$ | -0.80 | -0.74 | -0.73 | -0.82 | -0.71 | -0.71 | -0.71 | -0.71 | -0.71 | -0.68 | -0.65 | -0.64 | -0.64 | -0.63 | -0.63 |
| $\alpha_{2}$ | 0.31 | 0.18 | 0.18 | 0.38 | 0.13 | 0.13 | 0.13 | 0.13 | 0.13 | 0.36 | 0.10 | 0.09 | 0.09 | 0.09 | 0.09 |
| $\psi$ | -0.12 | -0.10 | -0.10 | -0.11 | -0.07 | -0.07 | -0.07 | -0.07 | -0.07 | -0.07 | 0.04 | 0.05 | 0.05 | 0.05 | 0.05 |

Table 4.7: Correlation coefficients for FSAC at various simulation times on the NSFNET topology. Significant values given in bold.

|  | High Load |  |  |  |  |  | Medium Load |  |  |  |  |  | Low Load |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | 3 | 100s | 200s | 300s | 400s | 500s | 3s | 100s | 200s | 300s | 400s | 500s | 3s | 100s | 200s | 300s | 400s | 500s |
| Number of Entries | -0.24 | -0.17 | -0.17 | -0.17 | -0.17 | -0.14 | -0.25 | -0.16 | -0.16 | -0.16 | -0.16 | -0.16 | -0.20 | -0.25 | -0.22 | -0.21 | -0.21 | -0.21 |
| $\beta$ | -0.26 | 0.02 | 0.03 | 0.03 | 0.04 | 0.11 | -0.38 | 0.01 | 0.03 | 0.03 | 0.04 | 0.04 | -0.56 | -0.17 | -0.11 | -0.08 | -0.07 | -0.06 |
| $\alpha_{1}$ | -0.83 | -0.72 | -0.72 | -0.71 | -0.71 | -0.71 | -0.79 | -0.70 | -0.69 | -0.68 | -0.68 | -0.68 | -0.59 | -0.71 | -0.68 | -0.67 | -0.66 | -0.66 |
| $\alpha_{2}$ | 0.47 | 0.15 | 0.15 | 0.15 | 0.15 | 0.16 | 0.46 | 0.11 | 0.10 | 0.10 | 0.10 | 0.10 | 0.35 | 0.23 | 0.19 | 0.17 | 0.17 | 0.16 |
| $\psi$ | -0.06 | -0.06 | -0.06 | -0.06 | -0.06 | -0.07 | -0.03 | -0.04 | -0.04 | -0.04 | -0.04 | -0.04 | -0.03 | 0.05 | 0.08 | 0.09 | 0.10 | 0.10 |

### 4.5 Performance Comparisons

In this section, a performance evaluation of ACOR and FSAC is performed to determine which is the best performing algorithm. The mean BLP of both algorithms at a specified time are compared to determine which performs best. The performance of the algorithms over time are also analysed. This evaluation is necessary as the simulations done for the parameter study were not repeated, hence the results are not considered reliable enough to allow conclusion of which algorithm performs best.

The experimental procedure used in the performance comparison is presented, followed by a discussion of the parameters selected. The presentation and analysis of the results obtained is then performed. The presented results consist of a discussion of the final BLP obtained by the algorithms (Section 4.5.3), and an analysis of the performance over time of the algorithms (Section 4.5.4).

### 4.5.1 Experimental Procedure

Performance evaluations will be performed to determine the performance of the various algorithms on the scenarios presented in Section 4.3. Using the best parameters that were obtained from the parameter study process. Each configuration (a topology, a load, and an algorithm) will be run 30 times with different random seeds, in order to ensure the reliability of the results.

Packets are generated using exponentially distributed inter-arrival times with the mean of the distribution set as specified in Table 4.8 (increasing the mean inter arrival time decreases the load). The size of the generated packets were exponentially distributed with the mean set at 1.5 kB . The burst assembly scheme used was size based, with the maximum burst size being 15 kB . Preliminary study indicated that spectrum width had an effect on the performance of the algorithms, hence it was decided that further study with larger spectrum widths would be done. The algorithms were run on the NSFNET and SIMPLE topologies with increased spectrum widths, while still maintaining the same load level. the NSFNET topology is run with 200 GHz ( $16 \times 12.5 \mathrm{GHz}$ flexi-grid channels), $300 \mathrm{GHz}(24 \times 12.5 \mathrm{GHz}$ flexi-grid channels) and 400 GHz ( $32 \times 12.5 \mathrm{GHz}$ flexi-grid channels) spectrum widths; and the SIMPLE topology is run with 100 GHz ( 8 x 12.5 GHz flexi-grid channels), 200 GHz ( $16 \times 12.5 \mathrm{GHz}$ flexi-grid channels) and 300 GHz ( $24 \times 12.5 \mathrm{GHz}$ flexi-grid channels) spectrum widths. The mean inter-arrival times for packet generation used for the increased spectrum widths can also be seen in Table 4.8.

Table 4.8: Mean inter-arrival time values used to achieve various loads in performance comparisons

| Topology | Spectrum Width <br> $(\mathbf{G H z})$ | High (ms) | Medium (ms) | Low (ms) |
| :---: | :---: | :---: | :---: | :---: |
| SIMPLE | 100 | 0.065 | 0.11 | 0.7 |
|  | 200 | 0.033 | 0.055 | 0.35 |
|  | 300 | 0.023 | 0.038 | 0.25 |
| NSFNET | 200 | 0.09 | 0.2 | 0.98 |
|  | 300 | 0.062 | 0.134 | 0.65 |
|  | 400 | 0.048 | 0.105 | 0.5 |

The results obtained from the simulations, are the final BLP and BLP values over time for all runs that were performed. The final BLP for low, medium and high loads on the SIMPLE topology, as well as low and medium loads on the NSFNET topology is obtained at 800s. The final BLP for high loads on the NSFNET topology is obtained at 600s (this was due to resource constraints). The final BLP's for each algorithm will be averaged and a $95 \%$ confidence interval obtained for each mean. The best algorithm will be the algorithm with the lowest mean. The algorithms will also subjected to the Mann-Whitney U test ${ }^{2}$ against each other, in order to ensure that there is a statistically significant difference in their performance. Tables of the final BLP obtained for each algorithm can be found in Appendix A.

### 4.5.2 Parameter Selection

The parameters chosen for ACOR and FSAC, for various loads, on the SIMPLE and NSFNET topologies are given in Tables 4.9 and 4.10, respectively. This parameters were manually fine-tuned using insights gained from the parameter study performed in Section 4.4. While fine-tuning the parameters, it was observed that the best performing set of parameters, for both algorithms were problem dependent; as no single set of parameters was the best performing for every topology and load.

For ACOR, the ant send interval was held constant, and the other parameters varied to determine what their best values would be. it was found that the best ant send probability remained near constant across all scenarios. The parameters selected for $v, \delta \epsilon$ and $g$, tended to be low values. This suggest that non of them have a strong effect on the performance of the algorithm. No pattern could be observed concerning the parameters selected for $v, \delta \epsilon$ and $g$.

[^2]A major observation made while fine-tuning FSAC, was that in order to obtain the best performance, $\alpha_{2}$ should satisfy the following

$$
\begin{equation*}
\alpha_{2}>1-\left(\alpha_{1}+\alpha_{2}\right) \tag{4.1}
\end{equation*}
$$

equation (4.1) says that the probability of exploring entries within the entry table should be higher than the probability of creating new entries. It was found that if the probability of creating new entries is greater than or equal to the probability of exploring entries, the algorithm will not have enough opportunities to explore newly created entries; hence, the performance of the algorithm will be negatively affected.

Table 4.9: Parameter values used for ACOR in performance comparisons

| Algorithm | Topology | Load | $\boldsymbol{v}$ | $\boldsymbol{\delta}$ | $\boldsymbol{\epsilon}$ | $\boldsymbol{g}$ | Ant Send Probability | Ant Send Interval (ms) |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| ACOR |  | High | 0.3 | 0.1 | 1 | 0.02 | 0.8 | 50 |
|  |  | Medium | 0.1 | 0.05 | 0.1 | 0.1 | 0.8 | 50 |
|  |  | Low | 0.2 | 0.02 | 0.1 | 0.1 | 0.6 | 50 |
|  | NSFNET | High | 0.3 | 0.02 | 0.1 | 0.1 | 0.8 | 50 |
|  |  | 0.3 | 0.02 | 0.55 | 0.1 | 0.8 | 50 |  |
|  |  | Low | 0.2 | 0.02 | 0.55 | 0.1 | 0.8 | 50 |

Table 4.10: Parameter values used for FSAC in performance comparisons

| Algorithm | Topology | Load | $\boldsymbol{\alpha}_{\mathbf{1}}$ | $\boldsymbol{\alpha}_{\mathbf{2}}$ | $\boldsymbol{\beta}$ | $\boldsymbol{\psi}$ | Number of Entries |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| FSAC |  | High | 0.98 | 0.0175 | 0.1 | 0.28 | 10 |
|  |  | Medium | 0.98 | 0.0175 | 0.1 | 0.28 | 14 |
|  |  | Low | 0.98 | 0.0175 | 0.1 | 0.64 | 12 |
|  |  | High | 0.98 | 0.0175 | 0.1 | 0.64 | 14 |
|  |  | Medium | 0.98 | 0.0175 | 0.2 | 0.28 | 14 |
|  |  | Low | 0.98 | 0.0175 | 0.1 | 0.1 | 14 |

For FSAC, the values which offered the best performance for $\alpha_{1}$ and $\alpha_{2}$ remained constant across all scenarios; with $\alpha_{1}$ being a 0.98 , and $\alpha_{2}$ being 0.0175 . This has the effect of maximising exploitation while keeping exploration low. $\beta$ remained mostly constant, and low for all scenarios, hence minimising the effect of the path length on the routing algorithm. Based on the parameter study the value of $\beta$ can be explained as the algorithm trying to minimise the effect of $\beta$ for the duration of the run; since it only has a significant effect at the beginning of simulations it makes no sense to give it a high influence for the duration of a simulation.

### 4.5.3 Overall Performance Comparison

Figure 4.2 presents the results of the experiments performed on the SIMPLE topology. At low load (Figure 4.2a), FSAC performs worse than both ACOR and GSPR at the 100 GHz spectrum width mark. However, as the spectrum width increases FSAC becomes the best performing algorithm, with its performance at 300 GHz better than its performance at 200 GHz . ACOR's performance remains steady, and worse than GSPR for all spectrum widths. At medium load (Figure 4.2b), FSAC provides the best performance at all spectrum widths, with dramatic improvement being seen as the spectrum width increases. The performance of FSAC seen at 200 GHz and 300 GHz is comparable. Once again ACOR's performance remains steady, and worse than GSPR for all spectrum widths. At high load (Figure 4.2c), FSAC performs best at all spectrum widths. Once again dramatic improvement is seen as the spectrum width increases. However, at 300 GHz the performance of FSAC worsens over that obtained at 200 GHz . Once again ACOR's performance remains steady, and worse than GSPR for all spectrum widths.

Figure 4.3 presents the results of the experiments performed on the NSFNET topology. At low load (Figure 4.3a), FSAC performs worse than both ACOR and GSPR for all spectrum widths. However, FSAC's performance improves as the spectrum width increases. ACOR's performs worse than GSPR for all spectrum widths and its performance worsens as the spectrum width increases from 200 GHz to 300 GHz , with a slight improvement at 400 GHz . At medium load (Figure 4.3b), FSAC performs worse than both GSPR and ACOR at 200 GHz . However, it performs better as the spectrum width increases, with its performance better than GSPR and ACOR at 300 GHz and 400 GHz . ACOR performs worse than both GSPR and FSAC for all spectrum widths, except at the 300 GHz spectrum width where its performance is better than GSPR, and comparable to FSAC. At high load (Figure 4.3c), FSAC performs better than GSPR and ACOR for all spectrum widths, with its performance improving as the spectrum width increases. ACOR performs worse than FSAC and on par with GSPR for all spectrum widths, except at the 300 GHz spectrum width where its performance is slightly better than GSPR.

Table 4.11 presents results obtained from applying the Mann-Whitney U test to the results obtained from two algorithms. The Mann-Whitney U test compares the individuals in two independent groups to each other, to determine if there is a statistically significant difference in their distributions (hence, a difference in their means). In this case, a group is composed of the final BLP's obtained from the 30 runs, performed for a algorithm on a given network configuration. Table 4.11



| 0 GSPR |
| :--- |
| Qa FSAC |
| QaACOR |

К. at various loads

Table 4.11: Performance comparison of ACOR, FSAC and GSPR in cases with statistically significant results.

|  |  | Number of Statistically Significant Results |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Algorithms |  |  | Total | Won | Lost |
| ACOR | FSAC | 16 | 5 | 11 | Percent Won |
|  | GSPR | 7 | 1 | 6 | $31.25 \%$ |
|  | Total | $\mathbf{2 3}$ | $\mathbf{6}$ | $\mathbf{1 7}$ | $\mathbf{2 6 . 2 9 \%}$ |
| FSAC | ACOR | 16 | 11 | 5 | $68.75 \%$ |
|  | GSPR | 18 | 13 | 5 | $72.22 \%$ |
|  | Total | $\mathbf{3 4}$ | $\mathbf{2 4}$ | $\mathbf{1 0}$ | $\mathbf{7 0 . 5 9 \%}$ |
| GSPR | ACOR | 7 | 6 | 1 | $85.71 \%$ |
|  | FSAC | 18 | 5 | 13 | $27.78 \%$ |
|  | Total | $\mathbf{2 5}$ | $\mathbf{1 1}$ | $\mathbf{1 4}$ | $\mathbf{4 4 . 0 0 \%}$ |

summarises the total number of significant results for each pair of algorithms, and the total number of significant results where the algorithm in the first column, performs better than the algorithm in the second column. It also summarises the total number statistically significant cases where the algorithm in the first column wins and loses.

In Table 4.11, it is observed that FSAC performs better than ACOR. FSAC performs better in $70.59 \%$ of all significant cases, compared to $26.09 \%$ for ACOR and $44 \%$ for GSPR. FSAC has more significant results against the GSPR algorithm than ACOR does ( 18 for FSAC, and 7 for ACOR), of which it performs better than GSPR $72.22 \%$ of the significant cases, compared to $14.29 \%$ of the significant cases for ACOR. ACOR only performed better than FSAC in $31.25 \%$ of the significant cases.

### 4.5.4 Timewise Performance Comparison

Figures 4.4 to 4.9 show the performance over time of the algorithms tested. These curves were obtained by taking the average performance of an algorithm over time. From the graphs it can be noticed that the performance of GSPR remains constant over time as expected (due to it not performing any intelligent actions), and FSAC and ACOR's performance improves exponentially over time due to the processes of
the algorithms.
It is observed in all the graphs that FSAC starts at a lower BLP then ACOR; this combined with the improved final BLP in most scenarios, implies that FSAC will have a better average BLP over the lifetime of the network. It is noticed that the rate of convergence on the SIMPLE network is faster than on the NSFNET network, and that as the spectrum width increases, the rate of convergence also increases. Also as the load increases the rate of convergence for FSAC also increases.

For ACOR there is no real visible change in the curve of the graph across loads and spectrum widths. Also it is noticed from the gradient of FSAC at medium and high loads, that there might be possibility for improvements.






Figure 4.7: Time wise performance of algorithms on NSFNET at various loads with 200 GHz spectrum width.
Figure 4.8: Time wise performance of algorithms on NSFNET at various loads with 300 GHz spectrum width.

(c) High load

(b) Medium load



### 4.6 Discussion

As mentioned in Section 4.5.1, at the 100 GHz spectrum width on SIMPLE and the 200 GHz spectrum width on NSFNET, the algorithms struggled to better GSPR. Analysing the entry tables at nodes for FSAC, it was observed that FSAC tended to favour a single path for transmitting data between a source and destination. However, it tends to have multiple entries which use this single path, and which send on different wavelengths. What this means is that the algorithm works by segmenting resources along a path such that a given wavelength along a route is mainly used for transmission between a pair of nodes. The nodes learn from the feedback they receive that a given wavelength is in use by another node. However, in order for this to work, nodes needs to send at a high rate in order to re-affirm their hold over a wavelength along a route. Hence the poor performance at low loads might be due to the rate of transmission not being high enough for a node to successfully keep hold of a wavelength. The reason why increasing the spectrum width leads to improved performance is because the increase in spectrum resources, leads to reduced contention for each wavelength.

The poor performance of ACOR can be attributed to various reasons. From Section 4.4.2.1, it was observed that ACOR does not effectively utilise congestion information available to it. ACOR is a single path algorithm (as mentioned in Section 2.6.6), which uses conventional methods of wavelength assignment (random Wavelength in the case of the implementation used) similar to the GSPR algorithm. Hence its performance compared to that of GSPR should be similar. The fact that its performance compared to the GSPR algorithm is generally worse might be due to the fact that due to the nature of the algorithm it explores more than GSPR. The results obtained for ACOR have larger confidence intervals than those obtained for FSAC, which possibly indicates that ACOR is more sensitive to random effects.

### 4.7 Conclusion

The aim of this chapter was to determine which of two algorithms could serve as the basis for future development. To that end, a parameter study was performed on ACOR and FSAC to understand the way they work and determine parameters which will be used in the performance comparison of both algorithms. A performance comparison was performed and the results presented.

From the parameter study, it was observed that both ACO algorithms largely favour
exploitation over exploration, as the effect of parameters which would encourage exploration tended to have to be minimised. However, intuitively, it can be seen that exploration is the driver of improvements of the algorithms. Hence methods which maximise the success of exploratory phases would be necessary to improve performance. It was also noted that both algorithms were unable to properly apply their information sources, leading to certain poor performance.

FSAC was shown to perform better than both ACOR and GSPR in the majority of cases considered. Hence it is chosen as the algorithm which will be developed further. Further development of this algorithm will be carried out in Chapter 5.

## Chapter 5

## Proposed Improvements

### 5.1 Introduction

The previous chapter described experiments on two algorithms to understand how they operate and determine which performs best. FSAC was the best performing algorithm hence it was selected as the appropriate algorithm to undergo further development. While FSAC was the best performing algorithm, it was observed that its performance at low loads and on the complex NSFNET network was poor. Hence, the aim of this chapter will be to determine what modifications to FSAC might improve the aforementioned aspects.

The main focus of this chapter is to propose modifications which might improve the performance of FSAC (Section 5.2). A modification is also proposed to investigate how much of an effect various information sources have on the behaviour of the algorithms (Section 5.3). A parameter study is performed on the modified algorithms to determine what effects the modifications have on the performance of the algorithm, and what the optimal parameters for the algorithms might be (Section 5.4). Finally, a comparison is performed between FSAC and the modified algorithms to determine if they improve on the performance of FSAC (Section 5.5).

### 5.2 Incorporating Route Congestion Information

Based on observations made from the correlation data presented in Section 4.4.2.2, and observations about the operation of FSAC made in Section 4.5.3, it was concluded that seeking to minimise the path length does not constitute an effective
strategy to minimise the BLP of the network. An alternative strategy could be to minimise the number of nodes along a route. This would act to minimise the number of points of contention. However, this would be a static measure (that is a non changing attribute of a network), and might not offer noticeable improvements over minimising the shortest path, especially as the shortest path tends to have the least number of nodes. A more interesting proposal is using the congestion along a route to determine if a table entry will offer a greater chance of success than other entries in the table. To that end, modifications to the FSAC algorithm was made to take into account the congestion along routes.

In this section, steps taken to incorporate the use of estimated congestion information into the FSAC algorithm are presented, as well as other modifications which might improve the performance of the algorithm.

### 5.2.1 Congestion Estimation

The first question that needs to be asked is "how can congestion along a route be measured in an effective manner?". Three methods of congestion measurement (denoted with the prefix "CM") are proposed. Two of them estimate the congestion along a route, using the amount of free wavelengths along the route, while the other estimates the congestion along a route using the number of BCP's lost along a route due to contention.

- CM1 - CM1 measures the congestion along a path from node $s$ to node $d$ as

$$
B^{s d}=1-\prod_{k=1}^{J}\left(1-B_{k}\right)
$$

with

$$
\begin{equation*}
B_{k}(M)=\frac{M}{M+1} \tag{5.1}
\end{equation*}
$$

Where $M=W-U_{k}$, with $W$ being the total number of wavelengths and $U_{k}$ being the number of available wavelengths on fibre link $k . J$ is the number of links on the path.

Equation (5.1) is an adaptation of the Erlang-B formula given in equation (2.3), for a network which maintains the WC constraint. This is similar to how DABR (Section 2.6.2) estimates congestion along a path (see equation (2.4)) except that it has been adapted for the case with WC constraint.

- CM2 - CM2 follows the same procedure as CM1, except that $M$ becomes the
percentage of BCP's lost due to contention.
It is assumed that control channels are impairment free, hence, BCP's cannot be lost for any other reason than contention. Burst lost due to impairments are not considered in this measure, due to the fact that the majority of lost burst in a network are still due to contention as shown in Figure 3.11. And the impairments experienced by transmissions on a route, is considered to be a function of the number of transmissions being made on that route (as can be seen in equation (2.1)). Hence, using only the percentage of BCP's lost due to contention is sufficient to estimate how busy a route is.
- CM3 - CM3 measures the percentage of BCP's lost along a route by collecting the number of BCP's lost at nodes along the route due to contention, and total number of BCP's sent along the route, and calculating the percentage of BCP's lost along that route. Only BCP's lost due to contention are considered, for the same reasons given for CM2.


### 5.2.2 Incorporating Congestion

The implementation of FSAC remains as described in Section 4.2. However, the following modifications have been made in order to integrate congestion measurements into FSAC.

1. The tuple used by FSAC is redefined as $<r_{m}^{s d}, \varpi_{m}, \sigma_{m}, \chi_{m}, \tau_{m}, \theta_{m}>$; where $\theta_{m}$ is the congestion measured for the route $r_{m}^{s d}$. Equation (2.6) is replaced with

$$
\begin{equation*}
\hat{\zeta}=\arg \max _{\zeta_{m} \in \Xi_{s d}} \tau_{m}^{\gamma} \theta_{m}^{\epsilon} \tag{5.2}
\end{equation*}
$$

and equation (2.7) is replaced with

$$
\begin{equation*}
p_{o}=\frac{\tau_{o}^{\gamma} \theta_{o}^{\epsilon}}{\sum_{\zeta_{m} \in \Xi_{s d}} \tau_{m}^{\gamma} \theta_{m}^{\epsilon}} \tag{5.3}
\end{equation*}
$$

$\gamma$ and $\epsilon$ are pre-set constants. $\gamma$ is added to the algorithm, in order to add an extra degree of control to the algorithm.
2. Congestion information is recorded by BCP-ACK's of successful burst, travelling along the reverse route, towards the source node of their corresponding burst. On arriving at the source node, all entries which use the the route followed by the corresponding burst, are updated with the estimated BLP.

### 5.2.3 Other Modifications

The following changes are also made to FSAC based on observations made about its performance in Section 4.4.2.2

- It is stipulated that table entries for a destination must be unique. That is new entry additions to a destination must have a route, wavelength combination which is not currently in the entry table. This change was made in order to increase the speed at which good solutions might be encountered. This change is also expected to reduce the influence of the number of table entries on the performance of the algorithm, hence making the algorithm more efficient as the number of entries required for the algorithm's optimal operation can be minimised, hence minimising the number of computational operations that need to be performed at an edge node.
- The parameter $\gamma$ is used to control the effect of the pheromone concentration of a table entry, as shown in equation (5.2) and (5.3). The role of $\gamma$ is to control the greediness of the algorithm. This parameter was added to allow better control over the algorithms exploitation of good entries.
- The pheromone concentration equation given in equation (2.8), which is used to determine the pheromone concentration of an entry $m$ is replaced with

$$
\begin{equation*}
\tau_{m}=\frac{\sigma_{m}+1}{\sigma_{m}+\chi_{m}+1} \tag{5.4}
\end{equation*}
$$

The parameter $\psi$ is eliminated, and the exponent is removed. Investigation into the effects of $\psi$, showed that the exponent combined with $\psi$ as seen in equation (2.8), allowed for the control of the greediness of the algorithm. However, as seen in Section 4.4.2.2, the effect of $\psi$ was very weak and inconsistent. Hence, it was determined that the role of $\psi$ could be performed more effectively by the newly added $\gamma$.

Pheromone concentrations are initialised as random numbers between 0 and 1 in order to encourage exploration at the start of a simulation.

### 5.3 Information Deprived FSAC

In Section 4.4.2.2, it was observed that both $\alpha_{1}$ and $\alpha_{2}$ had a dominant effect on the performance of FSAC. This observation lead to the question, "how effective could FSAC be without any information about the network?". That is, how well would

FSAC perform without access to any information about the network, such as static information like the length of routes, and dynamic information like the congestion of routes. Knowing this was considered important, in order to quantify how much of an effect network information provided to the algorithm actually makes on the performance of the algorithm.

Hence, an Information Deprived FSAC (ID-FSAC) algorithm is presented. The tuple used by FSAC is redefined as $<r_{m}^{s d}, \varpi_{m}, \sigma_{m}, \chi_{m}, \tau_{m}, \Gamma_{m}>$; where $\Gamma_{m}$ is the number of times entry $m$ has been used. Equation (2.6) is replaced simply with

$$
\hat{\zeta}=\arg \max _{\zeta_{m} \in \Xi_{s d}} \tau_{m}
$$

and equation (2.7) is replaced with

$$
\begin{equation*}
p_{o}=\frac{\sum_{\zeta_{m} \in \Xi_{s d}} \Gamma_{m}-\Gamma_{o}}{\sum_{\zeta_{m} \in \Xi_{s d}} \Gamma_{m}} \tag{5.5}
\end{equation*}
$$

The intended effect of equation (5.5) is to probabilistically explore least used entries. As with the algorithms that incorporate congestion measures, the pheromone concentration equation (equation (2.8)) is replaced with equation (5.4) and table entries for a destination are made unique.

### 5.4 Parameter Study

In this section, correlation data obtained for the algorithms which incorporate the congestion measures and ID-FSAC are presented. The parameter studies are performed using the procedure described in Section 4.4.1. The values of the parameters investigated for the CM algorithms and ID-FSAC are given in Table 5.1 and 5.2, respectively.

The Pearson correlation coefficients obtained from analysing the obtained data at various seconds over the course of the simulation are presented in Tables 5.3 to 5.10. Statistically significant patterns $(p<0.05)$ observed for each algorithm are discussed in the following sections. Every simulation except for those performed at high load on the SIMPLE network, was able to achieve a simulation time of 500 s within the time limit allowed for a run. A time limit was required due to resource limitations.

Table 5.1: Parameter values used in parameter study of CM1, CM2 and CM3.

| Parameter | Values |
| :---: | :---: |
| $\alpha_{1}$ | $0.3,0.5,0.9$ |
| $\alpha_{2}$ | $\left(1-\alpha_{1}\right) * 0.3,\left(1-\alpha_{1}\right) * 0.5,\left(1-\alpha_{1}\right) * 0.11$ |
| $\gamma$ | $0,0.2,0.4,0.6,0.8,1,1.2,1.4$ |
| $\epsilon$ | $0,0.2,0.4,0.6,0.8,1,1.2,1.4$ |
| Number of Entries | $4,8,12$ |
| Number of Runs per topology | 5184 |

Table 5.2: Parameter values used in parameter study of ID-FSAC

| Parameter | Values |
| :---: | :---: |
| $\alpha_{1}$ | $0.3,0.5,0.9$ |
| $\alpha_{2}$ | $\left(1-\alpha_{1}\right)^{*} 0.3,\left(1-\alpha_{1}\right)^{*} 0.5,\left(1-\alpha_{1}\right)^{*} 0.1$ |
| Number of Entries | $4,8,12$ |
| Number of Runs per topology | 81 |

### 5.4.1 CM1

The following is observed from Tables 5.3 and 5.4

- The number of entries in a pheromone table for a given destination is observed to be very weakly positively correlated at medium and low load for the SIMPLE topology. It is also observed to be very weakly negatively correlated at 3s on the NSFNET topology at medium load, and weakly positively correlated at 3s on the NSFNET topology at low load. No significant correlations were observed at any other point. The generally weakly positive and non-significant correlation coefficients obtained, contrasted with those obtained for the FSAC indicates that making the table entries unique had the desired effect.
- $\epsilon$ is observed to be weakly positively correlated at high and medium load on the SIMPLE topology, and at high load on the NSFNET topology. However, at low loads on the SIMPLE and NSFNET topology, $\epsilon$ is observed to be weakly negatively correlated. This indicates that the effect of the congestion measure, should be minimised at high and medium loads, but should be increased at low loads.
- $\gamma$ is observed to be weakly negatively correlated for all loads on the SIMPLE topology, and for high and medium loads on the NSFNET topology. At low load on the NSFNET topology, $\gamma$ is only significantly correlated at the begin-

Table 5.3: Correlation coefficients for CM1 at various simulation times on the SIMPLE topology. Significant values given in bold.

|  | High Load |  |  | Medium Load |  |  |  |  |  | Low Load |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | 3s | 100s | 200s | 3s | 100s | 200s | 300s | 400s | 500s | 3s | 100s | 200s | 300s | 400s | 500s |
| Number of Entries | 0.01 | 0.02 | 0.05 | 0.06 | 0.09 | 0.09 | 0.08 | 0.08 | 0.08 | -0.02 | 0.07 | 0.07 | 0.07 | 0.07 | 0.07 |
| $\epsilon$ | 0.24 | 0.23 | 0.23 | 0.07 | 0.11 | 0.11 | 0.11 | 0.11 | 0.11 | -0.39 | -0.19 | -0.18 | -0.18 | -0.18 | -0.18 |
| $\gamma$ | -0.37 | -0.33 | -0.33 | -0.24 | -0.21 | -0.21 | -0.21 | -0.21 | -0.21 | -0.02 | -0.05 | -0.06 | -0.06 | -0.06 | -0.06 |
| $\alpha_{1}$ | -0.69 | -0.68 | -0.68 | -0.77 | -0.73 | -0.73 | -0.73 | -0.73 | -0.73 | -0.67 | -0.69 | -0.69 | -0.69 | -0.69 | -0.69 |
| $\alpha_{2}$ | 0.25 | 0.23 | 0.24 | 0.28 | 0.19 | 0.19 | 0.19 | 0.19 | 0.19 | 0.28 | 0.12 | 0.12 | 0.12 | 0.11 | 0.11 |


|  | High Load |  |  |  |  |  | Medium Load |  |  |  |  |  | Low Load |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | 3s | 100s | 200s | 300s | 400s | 500s | 3s | 100s | 200s | 300s | 400s | 500s | 3s | 100s | 200s | 300s | 400s | 500s |
| Number of Entries | -0.04 | -0.01 | -0.01 | -0.01 | -0.02 | 0.00 | -0.05 | -0.01 | -0.01 | -0.01 | -0.01 | -0.01 | 0.17 | 0.02 | 0.02 | 0.02 | 0.02 | 0.02 |
| $\epsilon$ | 0.08 | 0.12 | 0.12 | 0.12 | 0.12 | 0.14 | -0.10 | 0.01 | 0.01 | 0.01 | 0.02 | 0.02 | -0.27 | -0.34 | -0.30 | -0.29 | -0.28 | -0.28 |
| $\gamma$ | -0.23 | -0.19 | -0.19 | -0.19 | -0.19 | -0.17 | -0.14 | -0.13 | -0.13 | -0.13 | -0.13 | -0.13 | -0.15 | -0.03 | -0.04 | -0.04 | -0.04 | -0.04 |
| $\alpha_{1}$ | -0.77 | -0.71 | -0.72 | -0.72 | -0.72 | -0.72 | -0.79 | -0.71 | -0.71 | -0.71 | -0.71 | -0.71 | -0.59 | -0.61 | -0.61 | -0.61 | -0.61 | -0.61 |
| $\alpha_{2}$ | 0.27 | 0.15 | 0.15 | 0.15 | 0.15 | 0.16 | 0.31 | 0.11 | 0.11 | 0.11 | 0.11 | 0.11 | 0.28 | 0.12 | 0.09 | 0.08 | 0.08 | 0.08 |

Table 5.5: Correlation coefficients for CM2 at various simulation times on the SIMPLE topology. Significant values given in bold.

ning, where it is weakly negatively correlated. It is observed that the strength of the correlation coefficients decrease as the load decreases.

- $\alpha_{1}$ is observed to range from moderately to strongly negatively correlated with the obtained BLP at the considered data points in the simulations.
- $\alpha_{2}$ is weakly positively correlated with the obtained BLP at the considered data points in the simulations.

From the correlation coefficients obtained for $\alpha_{1}$ and $\alpha_{2}$, it can be said that the algorithm favours exploitation over exploration. This is similar to what has been observed about FSAC in Section 4.4.2.2. Increasing the effect of the congestion measure is found to have a detrimental effect at medium and high loads. However, at low loads increasing the effect of the congestion measure has a positive effect on the obtained BLP.

### 5.4.2 CM2

The following is observed from Tables 5.5 and 5.6

- The number of entries in a pheromone table for a given destination is found to be very weakly positively correlated at low load on the SIMPLE topology. It is not significantly correlated at any other point.
- $\epsilon$ is observed to be weakly positively correlated at high and medium loads for the SIMPLE network, and at high load for the NSFNET topology. It is however found to be weakly negatively correlated at low load for the SIMPLE topology, and medium and low load for the NSFNET topology.
- $\gamma$ observed to be weakly negatively correlated at medium and high loads for both topologies. and weakly positively correlated at low loads on both both topologies.
- $\alpha_{1}$ is observed to range from moderately to strongly negatively correlated with the obtained BLP at the considered data points in the simulations.
- $\alpha_{2}$ is weakly positively correlated with the obtained BLP at the considered data points in the simulations

From the correlation coefficients obtained for $\alpha_{1}$ and $\alpha_{2}$, it can be said that the algorithm favours exploitation over exploration. This is similar to what has been observed about FSAC in Section 4.4.2.2. Increasing the effect of the congestion measure is found to have a detrimental effect at medium and high loads. However,
at low loads increasing the effect of the congestion measure has a positive effect on the obtained BLP. Increasing the effect of the pheromone value has a positive effect at high and medium loads. However, at low loads this switches and the effect becomes detrimental.

### 5.4.3 CM3

The following is observed from Tables 5.7 and 5.8

- The number of entries in a pheromone table for a given destination is not significantly correlated with the obtained BLP at the considered data points in the simulations.
- $\epsilon$ is observed to be weakly positively correlated at medium and high load on both topologies. No statistically significant correlations were observed at low loads.
- $\gamma$ observed to range from weakly to moderately negatively correlated, for both topologies at all loads.
- $\alpha_{1}$ is moderately negatively correlated with the obtained BLP at the considered data points in the simulations.
- $\alpha_{2}$ is weakly positively correlated with the obtained BLP at the considered data points in the simulations.

From the correlation coefficients obtained for $\alpha_{1}$ and $\alpha_{2}$, it can be said that the algorithm favours exploitation over exploration. This is similar to what has been observed about FSAC in Section 4.4.2.2. Increasing the effect of the congestion measure is found to have a detrimental effect at medium and high loads, with no significant effect been observed at low loads. Increasing the effect of the pheromone value has a positive effect at all loads.

### 5.4.4 ID-FSAC

The following is observed from Tables 5.9 and 5.10

- The number of entries in a pheromone table for a given destination is not significantly correlated with the obtained BLP at the considered data points in the simulations. This indicates that making the table entries unique had the desired effect.
Table 5.6: Correlation coefficients for CM2 at various simulation times on the NSFNET topology. Significant values given in bold.

Table 5.7: Correlation coefficients for CM3 at various simulation times on the SIMPLE topology. Significant values given in bold.

|  | High Load |  |  | Medium Load |  |  |  |  |  | Low Load |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | 3 s | 100s | 200s | 3 s | 100s | 200s | 300s | 400s | 500s | 3 s | 100s | 200s | 300s | 400s | 500s |
| Number of Entries | 0.02 | 0.02 | 0.02 | 0.03 | 0.04 | 0.04 | 0.03 | 0.03 | 0.02 | 0.03 | 0.03 | 0.03 | 0.03 | 0.03 | 0.03 |
| $\epsilon$ | 0.22 | 0.19 | 0.19 | 0.13 | 0.11 | 0.11 | 0.10 | 0.11 | 0.10 | 0.02 | 0.03 | 0.03 | 0.03 | 0.03 | 0.03 |
| $\gamma$ | -0.41 | -0.38 | -0.38 | -0.43 | -0.38 | -0.38 | -0.38 | -0.38 | -0.34 | -0.24 | -0.27 | -0.27 | -0.27 | -0.27 | -0.27 |
| $\alpha_{1}$ | -0.67 | -0.67 | -0.67 | -0.68 | -0.66 | -0.67 | -0.67 | -0.67 | -0.69 | -0.60 | -0.67 | -0.66 | -0.66 | -0.66 | -0.66 |
| $\alpha_{2}$ | 0.30 | 0.27 | 0.28 | 0.34 | 0.25 | 0.25 | 0.25 | 0.25 | 0.28 | 0.30 | 0.24 | 0.23 | 0.23 | 0.23 | 0.23 |

Table 5.8: Correlation coefficients for CM3 at various simulation times on the NSFNET topology. Significant values given in bold.

|  | High Load |  |  |  |  |  | Medium Load |  |  |  |  |  | Low Load |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | 3s | 100s | 200s | 300s | 400s | 500s | 3s | 100s | 200s | 300s | 400s | 500s | 3 s | 100s | 200s | 300s | 400s | 500s |
| Number of Entries | 0.01 | -0.01 | -0.02 | -0.02 | -0.03 | -0.02 | 0.01 | -0.01 | -0.01 | -0.01 | -0.02 | -0.02 | -0.02 | 0.01 | 0.00 | 0.00 | 0.00 | 0.00 |
| $\epsilon$ | 0.14 | 0.11 | 0.11 | 0.11 | 0.10 | 0.10 | 0.07 | 0.05 | 0.05 | 0.05 | 0.05 | 0.05 | 0.02 | -0.02 | -0.01 | -0.01 | -0.01 | -0.01 |
| $\gamma$ | -0.46 | -0.37 | -0.37 | -0.37 | -0.37 | -0.34 | -0.41 | -0.32 | -0.32 | -0.32 | -0.32 | -0.31 | -0.22 | -0.26 | -0.25 | -0.25 | -0.25 | -0.25 |
| $\alpha_{1}$ | -0.67 | -0.67 | -0.67 | -0.67 | -0.68 | -0.69 | -0.70 | -0.67 | -0.67 | -0.67 | -0.67 | -0.67 | -0.25 | -0.72 | -0.70 | -0.69 | -0.69 | -0.69 |
| $\alpha_{2}$ | 0.38 | 0.24 | 0.24 | 0.24 | 0.24 | 0.27 | 0.43 | 0.22 | 0.22 | 0.22 | 0.22 | 0.22 | 0.16 | 0.32 | 0.30 | 0.29 | 0.28 | 0.28 |

Table 5.9: Correlation coefficients for ID-FSAC at various simulation times on the SIMPLE topology. Significant values given in bold.

|  | High Load |  |  | Medium Load |  |  |  |  |  | Low Load |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | 3s | 100s | 200s | 3s | 100s | 200s | 300s | 400s | 500s | 3s | 100s | 200s | 300s | 400s | 500s |
| Number of Entries | 0.04 | 0.05 | 0.06 | 0.03 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.12 | 0.06 | 0.07 | 0.07 | 0.07 | 0.07 |
| $\alpha_{1}$ | -0.95 | -0.91 | -0.91 | -0.93 | -0.94 | -0.95 | -0.95 | -0.95 | -0.95 | -0.78 | -0.93 | -0.93 | -0.93 | -0.93 | -0.93 |
| $\alpha_{2}$ | 0.69 | 0.61 | 0.61 | 0.50 | 0.52 | 0.52 | 0.52 | 0.52 | 0.53 | 0.51 | 0.53 | 0.54 | 0.55 | 0.55 | 0.54 |

Table 5.10: Correlation coefficients for ID-FSAC at various simulation times on the NSFNET topology. Significant values given in bold.

|  | High Load |  |  |  |  |  | Medium Load |  |  |  |  |  | Low Load |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | 3s | 100s | 200s | 300s | 400s | 500s | 3s | 100s | 200s | 300s | 400s | 500s | 3s | 100s | 200s | 300s | 400s | 500s |
| Number of Entries | -0.06 | -0.01 | -0.01 | -0.01 | -0.01 | -0.01 | -0.12 | 0.01 | 0.02 | 0.02 | 0.02 | 0.02 | -0.33 | 0.00 | 0.02 | 0.02 | 0.02 | 0.02 |
| $\alpha_{1}$ | -0.96 | -0.94 | -0.94 | -0.94 | -0.94 | -0.94 | -0.96 | -0.94 | -0.94 | -0.94 | -0.94 | -0.94 | -0.32 | -0.94 | -0.93 | -0.93 | -0.93 | -0.92 |
| $\alpha_{2}$ | 0.58 | 0.51 | 0.51 | 0.52 | 0.52 | 0.52 | 0.67 | 0.51 | 0.52 | 0.52 | 0.52 | 0.52 | 0.35 | 0.53 | 0.53 | 0.53 | 0.53 | 0.53 |

- $\alpha_{1}$ is very strongly negatively correlated for all times and configurations investigated.
- $\alpha_{2}$ is moderately positively correlated for all times and configurations investigated. This observation combined with that made about $\alpha_{1}$ implies that the ID-FSAC algorithm is a greedy algorithm (that is it obtains better outcomes by exploiting locally optimal entries than by exploring other entries).

From the correlation coefficients obtained for $\alpha_{1}$ and $\alpha_{2}$, it can be said that the algorithm favours exploitation over exploration.

### 5.5 Performance Comparison

In this section, the results obtained from performing a comparison of the congestion measures (CM1, CM2 and CM3), ID-FSAC, FSAC and GSPR are presented, in order to determine which performs best. The experiments are performed using the procedure described in Section 4.5.1.

A discussion of the parameters selected is given. This is then followed by the presentation and analysis of the results obtained. The presented results consist of a discussion of the final BLP obtained by the algorithms (Section 5.5.2), and an analysis of the performance over time of the algorithms (Section 5.5.3).

### 5.5.1 Parameter Selection

The parameters chosen for the congestion measures and ID-FSAC, for various loads, on the SIMPLE and NSFNET topologies are given in Tables 5.11 and 5.12, respectively. As was observed in Section 4.5.2, the best performing set of parameters, for all algorithms was problem dependent; as no single set of parameters was the best performing for every topology and load. It was also observed that in order to obtain the best performance from CM1, CM2, CM3 and ID-FSAC; the value of $\alpha_{2}$ has to satisfy equation (4.1).

For all the algorithms, the values which offered the best performance for $\alpha_{1}$ and $\alpha_{2}$ remained constant across all scenarios; with $\alpha_{1}$ being a 0.98 , and $\alpha_{2}$ being 0.0175 . This has the effect of maximising exploitation while keeping exploration low. $\gamma$ tended to have a higher value on the SIMPLE topology than on the NSFNET topology for all CM algorithms. $\epsilon$ tended to have a higher value on the NSFNET topology than on the SIMPLE topology for all CM algorithms. This suggest that
the congestion measure is more effective on the more complex NSFNET topology, while on the SIMPLE topology, a better strategy is to exploit the best solution in the pheromone tables. For ID-FSAC, the number of entries per destination, was lower for high loads than for low loads; indicating that at higher loads the ID-FSAC algorithm tends to minimise exploration.

Table 5.11: Parameter values used for CM1, CM2 and CM3 in performance comparisons

| Algorithm | Topology | Load | $\alpha_{1}$ | $\alpha_{2}$ | $\gamma$ | $\epsilon$ | Number of Entries |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| CM1 | SIMPLE | High | 0.98 | 0.0175 | 1.4 | 0.6 | 8 |
|  |  | Medium | 0.98 | 0.0175 | 0.8 | 0.4 | 12 |
|  |  | Low | 0.98 | 0.0175 | 0.1 | 0.8 | 8 |
|  | NSFNET | High | 0.98 | 0.0175 | 0.6 | 1.4 | 12 |
|  |  | Medium | 0.98 | 0.0175 | 0.8 | 1.2 | 12 |
|  |  | Low | 0.98 | 0.0175 | 0.8 | 1.2 | 12 |
| CM2 | SIMPLE | High | 0.98 | 0.0175 | 1 | 0.1 | 8 |
|  |  | Medium | 0.98 | 0.0175 | 1 | 1.2 | 8 |
|  |  | Low | 0.98 | 0.0175 | 0.1 | 0.6 | 12 |
|  | NSFNET | High | 0.98 | 0.0175 | 0.2 | 0.8 | 12 |
|  |  | Medium | 0.98 | 0.0175 | 0.1 | 1 | 12 |
|  |  | Low | 0.98 | 0.0175 | 0.1 | 1.4 | 12 |
| CM3 | SIMPLE | High | 0.98 | 0.0175 | 1.2 | 0.1 | 12 |
|  |  | Medium | 0.98 | 0.0175 | 1.4 | 0.1 | 8 |
|  |  | Low | 0.98 | 0.0175 | 0.1 | 0.2 | 8 |
|  | NSFNET | High | 0.98 | 0.0175 | 0.8 | 0.2 | 12 |
|  |  | Medium | 0.98 | 0.0175 | 1 | 0.2 | 12 |
|  |  | Low | 0.98 | 0.0175 | 0.8 | 1.2 | 12 |

### 5.5.2 Overall Performance Comparison

Figure 5.1 presents the results of the experiments performed on the SIMPLE topology. At low load (Figure 5.1a), all ACO algorithms perform worse than GSPR at the 100 GHz spectrum width mark. However, as the spectrum width increases the ACO algorithms perform better than GSPR, with better performance at 300 GHz than at 200 GHz . At 100 GHz , CM1 performs best, followed by CM2, CM3, FSAC and ID-FSAC. This order holds as the spectrum width increases, except at 300 GHz where ID-FSAC has a slightly better performance than FSAC. At medium load

Table 5.12: Parameter values used for ID-FSAC in performance comparisons

| Algorithm | Topology | Load | $\boldsymbol{\alpha}_{\boldsymbol{1}}$ | $\boldsymbol{\alpha}_{\boldsymbol{2}}$ | Number of Entries |
| :---: | :---: | :--- | :---: | :---: | :---: |
| ID-FSAC |  | High | 0.98 | 0.0175 | 4 |
|  |  | Medium | 0.98 | 0.0175 | 8 |
|  |  | Low | 0.98 | 0.0175 | 8 |
|  |  | NSFNET | High | 0.98 | 0.0175 |
|  |  | 0.98 | 0.0175 | 4 |  |
|  |  | Low | 0.98 | 0.0175 | 8 |

(Figure 5.1b), FSAC performs best at 100 GHz followed by CM2, CM1, CM3, and ID-FSAC. This order holds as the spectrum width increases. However, the performance of all the ACO algorithms are comparable as the difference between FSAC's BLP and that of the other ACO algorithms is less than a 1\%. At high load (Figure 5.1c), FSAC performs best at 100 GHz followed by CM1, CM2, CM3, and ID-FSAC. At 200 GHz , CM2 takes the top spot, followed by FSAC, CM1, CM3 and ID-FSAC. At 300 GHz , CM3 performs best, followed by FSAC, CM2, CM1 and ID-FSAC. The performance of all the ACO algorithms are comparable as the difference between FSAC's BLP and that of the other ACO algorithms is less than a $1 \%$. All ACO algorithms perform better than GSPR for all spectrum widths, with the performance of the algorithms improving at 200 GHz compared to 100 GHz . However, something irregular occurs at 300 GHz , where the performance of the ACO algorithms becomes worse compared to their performance at 200 GHz . This is contrary to the pattern previously observed, where the performance of the algorithms improves, or doesn't change as the spectrum width increases. An unknown change in the dynamics of the network at this point which the algorithms aren't able to cater for, might be responsible for the observed worsening.

Figure 5.2 presents the results of the experiments performed on the NSFNET topology. At low load (Figure 5.2a), CM1 is the best performing ACO algorithm, followed by CM2, FSAC, CM3 and ID-FSAC. The order holds for all spectrum widths. At 200 GHz , none of the ACO algorithms perform better than GSPR, however, as the spectrum width increases, the performance of all the ACO algorithms improves, with CM1's performance being comparable to GSPR at 300 GHz , and surpassing GSPR at 400 GHz . At medium load (Figure 5.2 b ), CM1 is the best performing ACO algorithm, followed by CM3, FSAC, CM2 and ID-FSAC. The order holds for all spectrum widths. At 100 GHz , only CM1 performs better than GSPR, however, as the spectrum width increases, the performance of all the ACO algorithms improves,
with the performance of all ACO algorithms being comparable to GSPR 300 GHz , and all ACO surpassing GSPR at 400 GHz . At high load (Figure 5.2c), CM1 is the best performing ACO algorithm at 200 GHz , followed by CM2, FSAC, CM3 and ID-FSAC. At 300 GHz this order changes and CM2 performs best, followed by CM1, FSAC, CM3 and ID-FSAC. This order holds as the spectrum width increases. At 200 GHz , only ID-FSAC performs worse than GSPR, however, as the spectrum width increases, all the ACO algorithms perform better than GSPR. FSAC performs better than GSPR and ACOR for all spectrum widths, with its performance improving as the spectrum width increases. ACOR performs worse than FSAC and on par with GSPR for all spectrum widths, except at the 300 GHz spectrum width where its performance is slightly better than GSPR.

Table 5.13 presents results obtained from applying the Mann-Whitney U test to the results obtained from all algorithms. The table summarises the total number of significant results for each pair of algorithms, and the total number of significant results where the algorithm in the first column, performs better than the algorithm in the second column. It also summarises the total number statistically significant cases where the algorithm in the first column wins and loses. It is observed that all ACO algorithms perform better than ID-FSAC in all significant cases. However, ID-FSAC performs better than GSPR in the majority ( $62.5 \%$ ) of significant cases. This implies that a key part of the performance of the ACO's algorithms lies in the structure of the algorithm, however, incorporating other information sources into the algorithm yields considerable improvements. CM1 is observed to be the best performing algorithm, as comparisons to all other algorithms show it to be the best performer in the majority ( $91.67 \%$ ) of significant cases. This is followed by CM2, FSAC and CM3.

Studying the entry tables of the algorithms which incorporate congestion measurement, it was observed that they worked similar to FSAC (as detailed in Section 4.5.3), where a single route is heavily used for transmission between a source and destination, with multiple wavelengths on that path being employed. However, they tended to select less congested routes to perform their transmissions. The improvement of the algorithm at low loads is due to the the algorithm using less congested routes, hence reducing the contention for occupied wavelengths and reducing the BLP.


Table 5.13: Performance comparison of CM1, CM2, CM3, FSAC, GSPR, and IDFSAC, in cases with statistically significant results.

|  |  | Number of Statistically Significant Results |  |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Algorithms |  |  |  |  |  | Total | Won | Lost | Percent Won |
| CM1 | CM2 | 11 | 9 | 2 |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | CM3 | 13 | 12 | 1 |  |  |  |  |  |$]$

### 5.5.3 Timewise Performance Comparison

Figures 5.3 to 5.8 show the performance over time of the algorithms tested. These curves were obtained by taking the average performance of an algorithm over time. From the graphs it can be noticed that the performance of GSPR remains constant over time as expected (due to it not performing any intelligent actions), and the performance of the ACO algorithms improves exponentially over time due to the processes of the algorithms.

It is observed that FSAC starts at a lower BLP then the other ACO algorithms, on SIMPLE at medium and low loads. On the SIMPLE topology CM1 starts at the lower BLP. On all test performed on NSFNET, CM1 starts at the lowest BLP. this combined with the improved final BLP in most scenarios, implies that CM will have a better average BLP over the lifetime of the network. It is noticed that the rate of convergence on the SIMPLE network is faster than on the NSFNET network for all ACO algorithms, and that as the spectrum width and load increases, the rate of convergence also increases for all algorithms.

It is observed from the gradient of FSAC at medium and high loads, and from the gradient of all other ACO algorithms in all scenarios, that there might be possibility for improvements.




### 5.6 Conclusion

In this chapter, changes which could be made to improve the performance and robustness of FSAC were proposed and implemented. FSAC was adapted to consider network congestion measurement in its decision making processes and multiple methods of measuring the congestion on the network were proposed (these are hereby collectively referred to as CM algorithms). An adaptation of the FSAC algorithm (called ID-FSAC) which only uses knowledge available at a node of the success and failures of transmissions made using an entry, to make decisions was also proposed to serve as a baseline for evaluation of the impact of the additional network information (route lengths and congestion measures) given to the other algorithms.

Parameter studies was performed on the CM algorithms in order to determine the effects of the congestion measures on the operation of the algorithm, and determine parameters which will be used in the performance comparison of the algorithms. The results of the performance comparison of FSAC, ID-FSAC and the CM algorithms were then presented.

From the parameter study, it was noted, that the CM algorithms largely favoured exploitation over exploration (similar to the FSAC algorithm on which they are based). Increasing the effect of the congestion measures introduced tended to have a detrimental effect at high and medium load, probably because they encourage exploration on fast changing networks. However, at low loads, two of the three congestion measures presented had a positive effect on the performance of the network.

The results of the algorithm comparison showed that the algorithms with congestion measures attained better network performance than that obtained by FSAC. The improvements observed were especially striking for results obtained from simulations run on the complex NSFNET network, and at low loads on both the SIMPLE and NSFNET networks. Hence it can be said that the objective of this chapter as stated in the introduction has been achieved. The best performing congestion measure of the three presented was CM1. This will hereby be referred to as CM-FSAC.

Comparing the results obtained from ID-FSAC against those obtained by FSAC and the CM algorithms, it was observed that ID-FSAC never attained a significantly lower BLP than either FSAC or the CM algorithms. However it was often not far off from the results obtained by either FSAC or the CM algorithms. This speaks to the incredible potential of ACO for solving the RWA problem on OBS networks; as it shows that even a very rudimentary implementation can have a considerable impact on the performance of OBS networks.

## Chapter 6

## Conclusion

### 6.1 Introduction

This research improved the performance of a pre-existing ACO algorithm by introducing path congestion measures into the algorithm. The modified algorithm was evaluated on flexi-grid scenarios with the WC constraint imposed, in the presence of optical impairments.

The contributions of this research are discussed in Section 6.2. Identified limitations and recommendations for future research are given in Sections 6.3 and 6.4, respectively. Finally, an overall summary of this work is given in Section 6.5.

### 6.2 Research Contributions

The two objectives of this research as stated in Section 1.2 are:

1. Develop a simulator which is capable of simulating the operation of OBS over flexi-grid networks and can be used to evaluate the performance of algorithms.
2. Improve on the performance of current dynamic ACO based solutions to the RWA problem on flexi-grid OBS networks in the presence of the WC constraint and impairments.

The following contributions were made in the course of fulfilling the objectives given above

1. Implementation and validation of a flexi-grid OBS network simulator.
2. Verification of flexi-grid assumptions. That is, the assumption that flexi-grid leads to lower BLP's on OBS networks, and that non-linear impairments play a substantial role in the performance of these networks.
3. Parameter study of ACOR and FSAC, on a transparent flexi-grid OBS network, in the presence of impairments.
4. An improved FSAC algorithm, called CM-FSAC, which uses estimated congestion information about the network when routing burst.
5. Parameter study of CM-FSAC.

The remainder of this section gives a review of the steps taken to achieve both objectives and the obtained results.

### 6.2.1 Development of a Flexi-grid OBS Simulator

In order to study the behaviour of RWA algorithms over flexi-grid OBS networks, a simulator was required. This was deemed necessary as access to a physical OBS network, on which to evaluate the behaviour of algorithms was not available. Details on the operation of the simulator are given in Chapter 3. The simulator was designed and implemented using the process presented in Section 1.4. The process involved formulating the problem, to determine what capabilities are required of the simulator, what results will be recorded by the simulator, and what assumptions are being made.

It was decided that a simulation framework will be used, as opposed to building a simulator from scratch; hence a survey of three potential simulator frameworks to be used for the project was performed in Section 2.4.2. The framework considered most appropriate was then selected in Section 3.4. The simulator was then built according to the conceptual model given in Section 3.6. The simulator was verified by studying its operation to ensure there were no programming errors, and was validated by comparing the results obtained from the simulator to the analytical model presented in Section 2.4.3. The simulator has been made publicly accessible ${ }^{1}$ in other to allow other researchers in the field to use it in their research.

Finally, it was deemed important to validate that two of the basic assumptions of the research were accurate; the first being that flexi-grid offers better performance over fixed-grid transmission, and secondly, that non-linear impairments have more of an effect on flexi-grid transmissions than on fixed-grid transmissions. Both assumptions

[^3]were validated by the results obtained from the experiments performed in Section 3.9.

### 6.2.2 Improved ACO RWA Solution

The process of achieving this objective, was begun by studying previous ACO algorithms which had been applied to OBS networks (Section 2.6). Of the five algorithms reviewed, only two had been developed for flexi-grid OBS networks. The remaining three were developed with fixed grid in mind. The algorithms were found to have been evaluated under different scenarios, with different constraints imposed; hence no conclusions could be made about their relative performance.

In Chapter 4, two algorithms (ACOR and FSAC) were selected to be evaluated using the recently developed simulator. This was done in order to determine which of the algorithms might serve as a good base for future development. The algorithm which performed best was selected as the algorithm to be used in future steps. The evaluation process consisted of a parameter study to determine what the optimal parameters to be used for each algorithm in performance evaluations might be, and to understand the operation of the algorithms and the effects of their parameters on their performance. FSAC was found to perform better than ACOR in the performance evaluation. The evaluation allowed the determination of the mode of operation of the algorithms, and what their weak points were; hence showing opportunities for improvements.

Based on the observed weaknesses of the FSAC algorithm and insight into its functions, it was proposed that incorporating congestion measures into the FSAC algorithm might improve its performance. Three methods of measuring congestion on a network, and other modifications which might remedy other observed weak points within the operation of FSAC were suggested. The algorithms with congestion measures incorporated were then evaluated by performing a parameter study to understand the effects of the modifications and the modified algorithms performance and determine what the optimal parameters to be used for each algorithm in performance evaluations might be. Two of the methods of congestion measurement were found to improve on the performance of the FSAC algorithm.

### 6.3 Limitations

While the developed simulator is a large improvement over previous work, it cannot be considered to perfectly simulate the behaviour of real world networks. This is due to the possibility of the impairment equation which was used not being a perfect model of transmission impairments on a flexi-grid optical network; and the fact that packets were generated randomly with an exponential distribution, which had no bias placed towards certain destinations as may be the case in real world networks.

Although there are many possible network topologies, the evaluation of the algorithms was limited to two network topologies; hence the obtained results might not be a comprehensive representation of the potential and limitations of the algorithms.

### 6.4 Recommendation for Future Investigations

The CM-FSAC algorithm obtained from this research, only used congestion information. This was in order to allow the researcher determine in isolation from other possible information sources if congestion information was a viable way of improving ACO algorithms. However, future research could focus on introducing other information into the algorithm (such as path length, or a measure of fragmentation along a route) to investigate if this would improve performance. The impact of such added information on the performance of an algorithm, will most likely be more visible in a dynamic network setting, that is, a setting with bias placed towards certain destinations, and the possibility of nodes coming into and out of service due to faults, as might be the case in the real world. Hence, further research should be performed on dynamic networks and should as part of the research determine how to alter the influence of the various information available to the algorithm based on certain signals from the network.

A general question which should be answered in order to drive research in the field, and allow for more objective evaluation of algorithms is, "What is the theoretical limit of improvement to the BLP which might be achieved by an algorithm for a given network scenario?". Answering this question, would give a concrete target to be aimed for by researchers in the field, and would make the field more accessible to network operators who might be investigating implementing an algorithm on their network.

The simulator could be improved by integrating the INET framework in order to allow evaluation of algorithms in the presence of various network protocols; im-
plementing the JET reservation algorithm to determine if this has any effect on performance of the ACO algorithms; improving burst generation in the simulator, to allow for faster simulations; and improving the impairment model used.

### 6.5 Summary

The primary objective of this research was to improve the performance of ACO RWA algorithms over flexi-grid OBS networks with the WC constraint imposed, in the presence of impairments. To that end an OBS simulator was developed to allow evaluation of RWA algorithms on flexi-grid OBS networks with impairment modelling. Two pre-existing ACO algorithms were then evaluated, to understand their operation, and determine their performance. The best performing algorithm was then selected as the algorithm to be improved. Based on insights gained from the evaluation of the algorithms, incorporating congestion measures into the algorithm was considered to be a viable way of improving the algorithms performance. The results obtained showed that congestion awareness did improve the performance of OBS networks. Hence the primary objective of the research was achieved.

## References

Amaya, N., Irfan, M., Zervas, G., Banias, K., Garrich, M., Henning, I., ... Richardson, D. J. (2011). Gridless optical networking field trial: Flexible spectrum switching, defragmentation and transport of $10 \mathrm{G} / 40 \mathrm{G} / 100 \mathrm{G} / 555 \mathrm{G}$ over $620-$ km field fiber. 2011 37th European Conference and Exhibition on Optical Communication, 19(26), 1-3.
Asuero, A., Sayago, A., \& Gonzalez, A. (2006). The correlation coefficient: An overview. Critical reviews in analytical chemistry, 36(1), 41-59.
Auer, P., Cesa-Bianchi, N., \& Fischer, P. (2002). Finite-time analysis of the multiarmed bandit problem. Machine Learning, 47(2-3), 235-256.
Azodolmolky, S., Klinkowski, M., Marin, E., Careglio, D., Pareta, J. S., \& Tomkos, I. (2009). A survey on physical layer impairments aware routing and wavelength assignment algorithms in optical networks. Computer Networks, 53(7), 926944
Banks, J., Nelson, B. L., Carson, J. S., \& Nicol, D. M. (2010). Discrete-Event System Simulation. PrenticeHall international series in industrial and systems engineering, 640.
Battestilli, T., Perros, H., \& Carolina, N. (2003). An Introduction To Optical Burst Switching. IEEE Optical Communications, (8), 10-15.
Bjornstad, S., Gauger, C. M., Nord, M., Baert, E., Callegati, F., Careglio, D., ... Solé-Pareta, J. (2003). Optical Burst Switching and Optical Packet Switching. Advanced Infrastructure for Photonic Networks - Extended Final Report of COST Action 266, 115-154.
Blum, C. (2005). Ant colony optimization: Introduction and recent trends. Physics of Life Reviews, 2(4), 353-373.
Bluman, A. (2012). Elementary Statistics : A Step by Step Approach 8th edition.
Boiyo, D. K., Chabata, T., Kipnoo, R. E., Gamatham, R., Leitch, A., \& Gibbon, T. (2015). Crosstalk Penalties in Optical Fibre Transport for Next-Generation Flexible Spectrum Networks. In Southern africa telecommunication networks and applications conference (satnac) (September, pp. 21-24).

Chen, Y., Qiao, C., \& Yu, X. (2004). Optical burst switching: A new area in optical networking research. IEEE network, 18(3), 16-23.
Cohen, J. (1988). Statistical power analysis for the behavioral sciences (2nd edition). Erlbaum Associates, Hillsdale.
Dorigo, M. [M.], \& Di Caro, G. (1999). Ant colony optimization: a new metaheuristic. In Proceedings of the 1999 congress on evolutionary computationcec99 (cat. no. 99th8406) (Vol. 2, pp. 1470-1477)
Dorigo, M. [Marco], \& Gambardella, L. M. (1997). Ant colony system: a cooperative learning approach to the traveling salesman problem. Evolutionary Computation, IEEE Transactions on, 1 (1), 53-66.
Dorigo, M. [Marco], Maniezzo, V., \& Colorni, A. (1996). Ant system: Optimization by a colony of cooperating agents. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 26(1), 29-41.
Dorigo, M. [Marco], \& Stützle, T. (2001). An experimental study of the simple ant colony optimization algorithm. In 2001 wses international conference on evolutionary computation (ecffff01) (pp. 253-258).
Engelbrecht, A. P. (2007). Computational intelligence: an introduction. John Wiley $\&$ Sons.
Espina, F., Armendariz, J., García, N., Morató, D., Izal, M., \& Magaña, E. (2012). OBS network model for OMNeT++: a performance evaluation. In Proceedings of the 3rd international icst conference on simulation tools and techniques
Fei, X., Yoo, S. J. B., Yokoyama, H., \& Horiuchi, Y. (2005). Performance Comparison of Optical Burst and Circuit Switched Networks. In Optical fiber communication conference, 2005. technical digest. ofc/nfoec (Vol. 3, p. 3).
Fernandez-Palacios, J. P., López, V., Cruz, B., \& De Dios, O. G. (2014). Elastic optical networking: An operators perspective. In European conference on optical communication, ecoc.
Gerstel, O., Jinno, M., Lord, A., \& Yoo, S. J. B. (2012). Elastic optical networking: A new dawn for the optical layer? IEEE Communications Magazine, 50(2), 12-20.
Gravett, A. S., du Plessis, M. C., \& Gibbon, T. B. (2016). Hybridising Ant Colony Optimisation with a Upper Confidence Bound algorithm for Routing and Wavelength Assignment in an Optical Burst Switching network.
Gravett, A. S., du Plessis, M. C., \& Gibbon, T. B. (2017). A Distributed Antbased Algorithm for Routing and Wavelength Assignment in an Optical Burst Switching Flexible Spectrum Network with Transmission Impairments.
Guizani, M., Rayes, A., Khan, B., \& Al-fuqaha, A. (2010). NETWORK MODELING AND SIMULATION.

INET Framework. (2018). Retrieved April 2, 2018, from https://inet.omnetpp.org/ Institute of Communication Networks and Computer Engineering (IKR) - IKR Simulation and Emulation Library. (2017). Retrieved January 28, 2018, from http: //www.ikr.uni-stuttgart.de/IKRSimLib/
International Telecommunication Union - ITU-T. (2003). Spectral grids for WDM applications: CWDM wavelength grid. International Telecomunication Union, 12.

International Telecommunication Union - ITU-T. (2012). G. 694.1 (02/2012), Spectral grids for WDM applications: DWDM frequency grid. SERIES G.694.1, 1-16. Retrieved from http://www.itu.int/rec/T-REC-G.694.1-201202-I/en
Jinno, M., Takara, H., Kozicki, B., Tsukishima, Y., Sone, Y., \& Matsuoka, S. (2009). Spectrum-efficient and scalable elastic optical path network: Architecture, benefits, and enabling technologies. IEEE Communications Magazine, 47(11), 6673.

Kartalopoulos, S. V. [Stamatios V.]. (1999). Introduction to DWDM technology: Data in a rainbow.
Kartalopoulos, S. V. [Stamatis V.]. (2004). DWDM: Shaping the future communications network. IEEE Potentials, 23(4), 16-19.
Katehakis, M. N., \& Veinott, A. F. (1987). The Multi-Armed Bandit Problem: Decomposition and Computation. Mathematics of Operations Research, 12(2), 262-268
Kelly, F. P. (1986). Blocking probabilities in large circuit-switched networks. Advances in applied probability, 18(2), 473-505.
Kirci, P., \& Zaim, A. H. (2006). Just-in-tinie, just-enough-time and horizon signalling protocols on optical burst switches. Optica Applicata, 36(1).
Klinkowski, M. [Mirosław], Careglio, D., Solé-pareta, J., Marciniak, M., \& Member, S. (2009). Performance Overview of the Offset Time Emulated OBS Network Architecture. 27(14), 2751-2764.
Klinkowski, M. [Miroslaw], Pedro, J., Careglio, D., Pioro, M., Pires, J., Monteiro, P., \& Sole-Pareta, J. (2010). An overview of routing methods in optical burst switching networks. Optical Switching and Networking, 7(2), 41-53
Klinkowski, M. [Miroslaw], Ruiz, M., Velasco, L., Careglio, D., Lopez, V., \& Comellas, J. (2013). Elastic Spectrum Allocation for Time-Varying Traffic in FlexGrid Optical Networks. IEEE Journal on Selected Areas in Communications, 31 (1), 26-38.
Klinkowski, M. [Miroslaw], \& Walkowiak, K. (2011). Routing and spectrum assignment in spectrum sliced elastic optical path network. IEEE Communications Letters, 15(8), 884-886.

Law, A. M. (2003). How to conduct a successful simulation study. In Proceedings of the 35th conference on winter simulation: Driving innovation (pp. 66-70). Winter Simulation Conference.
Liu, X., Qiao, C., Yu, X., \& Gong, W. (2006). A fair packet-level performance comparison of obs and ocs. In Optical fiber communication conference (pp. 4850). Optical Society of America.

López, V., \& Velasco, L. (Eds.). (2016). Elastic Optical Networks.
Luke, G., Larsen, C. P., Ehrhardt, A., \& Jansen, H. (2005). Optical Commuinications. Teletronikk, 101.2.
Luo, H., Hu, G., \& Li, L. (2004). Performance evaluation of optical burst switching networks with limited number wavelength conversion. In Optical transmission, switching, and subsystems (Vol. 5281, pp. 96-102). International Society for Optics and Photonics.
Maier, M. (2008). Optical Switching Networks. Cambridge University Press.
Maier, M., \& Reisslein, M. (2008). Trends in optical switching techniques: A short survey. IEEE Network, 22(6), 42-47.
Merrifield, M., \& Haran, B. (2013). What is the maximum Bandwidth? Retrieved April 2, 2018, from https://www.youtube.com/watch?v=0OOmSyaoAt0
Montgomery, D. C., \& Runger, G. c. (2003). Applied Statistics and Probability for Engineers Third Edition.
Nag, A., Tornatore, M., \& Mukherjee, B. (2010). Optical network design with mixed line rates and multiple modulation formats. Journal of Lightwave Technology, 28(4), 466-475.
Ngo, S. H., Jiang, X., \& Horiguchi, S. (2006). An ant-based approach for dynamic RWA in optical WDM networks. Photonic Network Communications, 11(1), 39-48.

Nleya, B., \& Mutsvangwa, A. (2014). Qos considerations in obs switched backbone networks. Global Journal of Computer Science and Technology, 14 (5), 10.
ns-3. (2018). Retrieved April 2, 2018, from https://www.nsnam.org/
Omnet++ Discrete Event Simulator - Home. (2016). Retrieved April 2, 2018, from https://www.omnetpp.org
OMNET++ Simulation Manual. (2018). Retrieved April 2, 2018, from https:// www.omnetpp.org/doc/omnetpp/manual/
Pedro, J., Pires, J., \& Carvalho, J. P. (2009). Distributed Routing Path Optimization for OBS Networks based on Ant Colony Optimization. IEEE Global Telecommunications Conference, 1-7.
Ramaswami, R., Sivarajan, K. N., \& Sasaki, G. H. (2010). Optical Networks: A Practical Perspective.

Riley, G. F., \& Henderson, T. R. (2010). The ns-3 network simulator. In K. Wehrle, M. Güneş, \& J. Gross (Eds.), Modeling and tools for network simulation (pp. 15-34)
Rosberg, Z., Zukerman, M., \& White, J. (2003). Blocking probabilities of optical burst switching networks based on reduced load fixed point approximations. IEEE INFOCOM 2003. Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE Cat. No.03CH37428), 3(100), 2008-2018

Rouskas, G. N. (1999). Routing and wavelength assignment in optical wdm networks. Encyclopedia of Telecommunications.
Rouskas, G. N., \& Xu, L. (2004). Chapter 1: Optical Packet Switching. In Optical wdm networks: Past lessons and path ahead (pp. 1-19). Massachusetts: Kluwer.
Saleh, A. A. M., \& Simmons, J. M. [Jane M.]. (2012). All-optical networkingevolution, benefits, challenges, and future vision. Proceedings of the IEEE, 100(5), 1105-1117.

Shaw, J. (2004). Mathematical Principles of Optical Fibre Commuinications.
Shen, G., \& Yang, Q. (2011). From coarse grid to mini-grid to gridless: How much can gridless help contentionless? 2011 Optical Fiber Communication Conference and Exposition and the National Fiber Optic Engineers Conference, 1(100), 1-3.
Simmons, J. M. [J M]. (2014). Optical Network Design and Planning
Sokolowski, J. A., \& Banks, C. M. (2009). Principles of Modeling.
Sommer, J., \& Scharf, J. (2010). Ikr simulation library. In K. Wehrle, M. Güneş, \& J. Gross (Eds.), Modeling and tools for network simulation (pp. 61-68)

Stützle, T., \& Hoos, H. H. (2000). MAX-MIN Ant System. Future Generation Computer Systems, 16(8), 889-914.
Taillard, E. D. (1998). FANT: fast ant system.
Taylor, R. (1990). Interpretation of the Correlation Coefficient: A Basic Review. Journal of Diagnostic Medical Sonography, 6(1), 35-39.
Teng, J., \& Rouskas, G. N. (2003). A comparison of the jit, jet, and horizon wavelength reservation schemes on a single obs node. In Proc. of the first international workshop on optical burst switching.
Teng, J., \& Rouskas, G. N. (2005). Wavelength selection in obs networks using traffic engineering and priority-based concepts. IEEE Journal on Selected Areas in Communications, 23(8), 1658-1669.
Tomkos, I., Azodolmolky, S., Solé-Pareta, J., Careglio, D., \& Palkopoulou, E. (2014). A tutorial on the flexible optical networking paradigm: State of the art, trends, and research challenges. Proceedings of the IEEE, 102(9), 1317-1337.

Transmode. (2013). WDM the Transmode way. Retrieved April 2, 2018, from https: //www.infinera.com/wp-content/uploads/2015/09/WDM-TheTransmodeWay_ A.pdf

Triay, J., \& Cervelló-Pastor, C. (2010). An ant-based algorithm for distributed routing and wavelength assignment in dynamic optical networks. Selected Areas in Communications, ... 28(4), 542-552.
Varga, A. [Andras]. (2010). Omnet++. In K. Wehrle, M. Güneş, \& J. Gross (Eds.), Modeling and tools for network simulation (pp. 35-59)

Varga, A. [András], \& Hornig, R. (2008). AN OVERVIEW OF THE OMNeT++ SIMULATION ENVIRONMENT. Proceedings of the First International ICST Conference on Simulation Tools and Techniques for Communications Networks and Systems, (January 2008)

Verma, S., Chaskar, H., \& Ravikanth, R. (2000). Optical burst switching: A viable solution for terabit ip backbone. IEEE network, 14 (6), 48-53.
Wilcox, R. R. (2009). Basic statistics: Understanding conventional methods and modern insights. Oxford University Press on Demand.
Wright, P., Lord, A., \& Velasco, L. (2013). The network capacity benefits of flexgrid. In Optical network design and modeling (ondm), 2013 17th international conference on (pp. 7-12). IEEE.
Xia, T. J., Gringeri, S., \& Tomizawa, M. (2012). High-capacity optical transport networks. Communications Magazine, IEEE, 50(11), 170-178.
Yen, J. Y. (1971). Finding the K Shortest Loopless Paths in a Network. Management Science, 17(11), 712-716
Yu, Y., Zhang, J., Zhao, Y., Cao, X., Lin, X., \& Gu, W. (2013). The first singlelink exact model for performance analysis of flexible grid wdm networks. In National fiber optic engineers conference. Optical Society of America.
Zang, H., Jue, J. P., Mukherjee, B., et al. (2000). A review of routing and wavelength assignment approaches for wavelength-routed optical wdm networks. Optical networks magazine, 1(1), 47-60.

## Appendix A

## Results

Table A.1: Cases with significant results for performance comparison of ACOR, FSAC, and GSPR

|  |  | SIMPLE |  |  | NSFNET |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Algorithms |  | Low | Medium | High | Low | Medium | High |
| ACOR | FSAC | All | All | All | All | $200 \mathrm{GHz}, 400 \mathrm{GHz}$ | $300 \mathrm{GHz}, 400 \mathrm{GHz}$ |
| ACOR | GSPR | None | 300 GHz | $100 \mathrm{GHz}, 300 \mathrm{GHz}$ | All | 300 GHz | none |
| FSAC | GSPR | All | All | All | All | All | All |
| All means " $100 \mathrm{GHZ}, 200 \mathrm{GHz}, 300 \mathrm{GHz} "$ for SIMPLE, and " $200 \mathrm{GHz}, 300 \mathrm{GHz}, 400 \mathrm{GHz} "$ for NSFNET |  |  |  |  |  |  |  |

Table A.2: Cases with significant results for performance comparison of CM1, CM2, CM3, FSAC, ID-FSAC and GSPR

Table A.3: Results Obtained From Running Algorithms on the SIMPLE Topology at Low Loads.

| Time (s) | Algorithm | Spectrum Width | Mean | Lower Confidence | Upper <br> Confi- <br> dence | Median | Minimum | Maximum | Range | Std.Dev. | Coef.Var. |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 800 | FSAC | 100 | 0.06620 | 0.06531 | 0.06710 | 0.06596 | 0.06224 | 0.07442 | 0.01218 | 0.00240 | 3.62858 |
| 800 | FSAC | 200 | 0.04853 | 0.04776 | 0.04930 | 0.04877 | 0.04386 | 0.05218 | 0.00832 | 0.00206 | 4.24764 |
| 800 | FSAC | 300 | 0.03717 | 0.03638 | 0.03795 | 0.03735 | 0.03109 | 0.04098 | 0.00989 | 0.00209 | 5.63131 |
| 800 | GSPR | 100 | 0.05302 | 0.05292 | 0.05311 | 0.05304 | 0.05254 | 0.05362 | 0.00108 | 0.00026 | 0.48294 |
| 800 | GSPR | 200 | 0.05233 | 0.05227 | 0.05239 | 0.05235 | 0.05205 | 0.05259 | 0.00054 | 0.00016 | 0.30073 |
| 800 | GSPR | 300 | 0.05105 | 0.05099 | 0.05111 | 0.05109 | 0.05077 | 0.05146 | 0.00069 | 0.00017 | 0.32863 |
| 800 | ACOR | 100 | 0.05583 | 0.05344 | 0.05822 | 0.05535 | 0.04870 | 0.08091 | 0.03221 | 0.00640 | 11.45939 |
| 800 | ACOR | 200 | 0.05392 | 0.05177 | 0.05607 | 0.05326 | 0.04627 | 0.07580 | 0.02953 | 0.00576 | 10.68062 |
| 800 | ACOR | 300 | 0.05289 | 0.05027 | 0.05550 | 0.05160 | 0.04562 | 0.07872 | 0.03311 | 0.00699 | 13.22475 |
| 800 | CM1 | 100 | 0.05518 | 0.05450 | 0.05586 | 0.05547 | 0.05218 | 0.05829 | 0.00611 | 0.00181 | 3.28555 |
| 800 | CM1 | 200 | 0.03396 | 0.03323 | 0.03469 | 0.03381 | 0.03072 | 0.03786 | 0.00714 | 0.00195 | 5.73717 |
| 800 | CM1 | 300 | 0.02207 | 0.02146 | 0.02268 | 0.02212 | 0.01694 | 0.02402 | 0.00708 | 0.00163 | 7.40347 |
| 800 | ID-FSAC | 100 | 0.07033 | 0.06897 | 0.07170 | 0.06878 | 0.06506 | 0.07927 | 0.01422 | 0.00366 | 5.20278 |
| 800 | ID-FSAC | 200 | 0.04983 | 0.04827 | 0.05140 | 0.04972 | 0.04228 | 0.06152 | 0.01923 | 0.00418 | 8.39746 |
| 800 | ID-FSAC | 300 | 0.03636 | 0.03502 | 0.03770 | 0.03630 | 0.03089 | 0.04670 | 0.01581 | 0.00358 | 9.85798 |
| 800 | CM2 | 100 | 0.05646 | 0.05583 | 0.05710 | 0.05629 | 0.05319 | 0.06063 | 0.00744 | 0.00169 | 2.99313 |
| 800 | CM2 | 200 | 0.03600 | 0.03532 | 0.03669 | 0.03558 | 0.03215 | 0.04003 | 0.00787 | 0.00183 | 5.08603 |
| 800 | CM2 | 300 | 0.02323 | 0.02256 | 0.02391 | 0.02279 | 0.01990 | 0.02677 | 0.00686 | 0.00181 | 7.79303 |
| 800 | CM3 | 100 | 0.06448 | 0.06370 | 0.06526 | 0.06469 | 0.05957 | 0.06884 | 0.00927 | 0.00209 | 3.23836 |
| 800 | CM3 | 200 | 0.04465 | 0.04360 | 0.04569 | 0.04383 | 0.03932 | 0.05081 | 0.01149 | 0.00280 | 6.27307 |
| 800 | CM3 | 300 | 0.03133 | 0.03014 | 0.03253 | 0.03140 | 0.02687 | 0.03857 | 0.01170 | 0.00321 | 10.24414 |

Table A.4: Results Obtained From Running Algorithms on the SIMPLE Topology at Medium Loads.

| Time (s) | Algorithm | $\begin{aligned} & \text { Spectrum } \\ & \text { Width (GHz) } \end{aligned}$ | Mean | Lower Confidence |  | Median | Minimum | Maximu | Range | Std.Dev. | Coef.Var. |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 800 | FSAC | 100 | 0.24068 | 0.23895 | 0.24241 | 0.23934 | 0.23329 | 0.25270 | 0.01941 | 0.00462 | 1.91974 |
| 800 | FSAC | 200 | 0.16746 | 0.16502 | 0.16991 | 0.16805 | 0.15388 | 0.18293 | 0.02906 | 0.00655 | 3.90890 |
| 800 | FSAC | 300 | 0.16458 | 0.16286 | 0.16630 | 0.16425 | 0.15706 | 0.17333 | 0.01627 | 0.00461 | 2.80121 |
| 800 | GSPR | 100 | 0.25314 | 0.25306 | 0.25322 | 0.25311 | 0.25252 | 0.25358 | 0.00106 | 0.00021 | 0.08283 |
| 800 | GSPR | 200 | 0.25394 | 0.25388 | 0.25400 | 0.25394 | 0.25359 | 0.25423 | 0.00064 | 0.00016 | 0.06239 |
| 800 | GSPR | 300 | 0.25466 | 0.25462 | 0.25470 | 0.25465 | 0.25448 | 0.25489 | 0.00041 | 0.00011 | 0.04457 |
| 800 | ACOR | 100 | 0.26063 | 0.25300 | 0.26826 | 0.25477 | 0.24221 | 0.32333 | 0.08112 | 0.02005 | 7.69273 |
| 800 | ACOR | 200 | 0.26144 | 0.25505 | 0.26783 | 0.25808 | 0.24420 | 0.31538 | 0.07117 | 0.01712 | 6.54786 |
| 800 | ACOR | 300 | 0.26289 | 0.25577 | 0.27001 | 0.25832 | 0.24197 | 0.34217 | 0.10019 | 0.01908 | 7.25629 |
| 800 | CM1 | 100 | 0.24468 | 0.24296 | 0.24640 | 0.24516 | 0.23318 | 0.25482 | 0.02164 | 0.00460 | 1.88197 |
| 800 | CM1 | 200 | 0.16796 | 0.16550 | 0.17041 | 0.16751 | 0.15605 | 0.18117 | 0.02512 | 0.00658 | 3.91997 |
| 800 | CM1 | 300 | 0.16661 | 0.16417 | 0.16905 | 0.16719 | 0.15206 | 0.18009 | 0.02803 | 0.00653 | 3.91761 |
| 800 | ID-FSAC | 100 | 0.25339 | 0.25183 | 0.25495 | 0.25357 | 0.24456 | 0.26067 | 0.01611 | 0.00419 | 1.65255 |
| 800 | ID-FSAC | 200 | 0.17870 | 0.17617 | 0.18123 | 0.17818 | 0.16648 | 0.18908 | 0.02261 | 0.00678 | 3.79158 |
| 800 | ID-FSAC | 300 | 0.17776 | 0.17545 | 0.18007 | 0.17626 | 0.16783 | 0.18893 | 0.02110 | 0.00619 | 3.48349 |
| 800 | CM2 | 100 | 0.24346 | 0.24188 | 0.24504 | 0.24323 | 0.23457 | 0.25188 | 0.01731 | 0.00423 | 1.73913 |
| 800 | CM2 | 200 | 0.16561 | 0.16346 | 0.16776 | 0.16494 | 0.15710 | 0.17749 | 0.02039 | 0.00576 | 3.47700 |
| 800 | CM2 | 300 | 0.16579 | 0.16336 | 0.16822 | 0.16602 | 0.15287 | 0.18124 | 0.02837 | 0.00651 | 3.92688 |
| 800 | CM3 | 100 | 0.24463 | 0.24313 | 0.24613 | 0.24522 | 0.23750 | 0.25366 | 0.01616 | 0.00402 | 1.64353 |
| 800 | CM3 | 200 | 0.16922 | 0.16735 | 0.17109 | 0.16960 | 0.15568 | 0.17937 | 0.02369 | 0.00501 | 2.96056 |
| 800 | CM3 | 300 | 0.16241 | 0.16026 | 0.16456 | 0.16228 | 0.15179 | 0.17038 | 0.01859 | 0.00575 | 3.53973 |

Table A.5: Results Obtained From Running Algorithms on the SIMPLE Topology at High Loads.

| Time(s) | Algorithm | Spectrum Width (GHz) | Mean | Lower Confidence | Upper Confidence | Median | Minimum | Maximun | Range | Std.Dev. | Coef.Var. |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 800 | FSAC | 100 | 0.34212 | 0.34027 | 0.34398 | 0.34239 | 0.32918 | 0.35153 | 0.02235 | 0.00497 | 1.45363 |
| 800 | FSAC | 200 | 0.26687 | 0.26437 | 0.26936 | 0.26614 | 0.25116 | 0.28295 | 0.03179 | 0.00669 | 2.50748 |
| 800 | FSAC | 300 | 0.29987 | 0.29822 | 0.30152 | 0.29945 | 0.29128 | 0.30904 | 0.01776 | 0.00441 | 1.47225 |
| 800 | GSPR | 100 | 0.35502 | 0.35497 | 0.35508 | 0.35505 | 0.35475 | 0.35532 | 0.00057 | 0.00015 | 0.04198 |
| 800 | GSPR | 200 | 0.35372 | 0.35368 | 0.35376 | 0.35373 | 0.35347 | 0.35391 | 0.00044 | 0.00011 | 0.03063 |
| 800 | GSPR | 300 | 0.35245 | 0.35241 | 0.35249 | 0.35247 | 0.35219 | 0.35264 | 0.00044 | 0.00010 | 0.02815 |
| 800 | ACOR | 100 | 0.36246 | 0.35899 | 0.36593 | 0.36120 | 0.34899 | 0.37869 | 0.02970 | 0.00930 | 2.56543 |
| 800 | ACOR | 200 | 0.36454 | 0.35852 | 0.37056 | 0.36183 | 0.34674 | 0.41760 | 0.07086 | 0.01613 | 4.42540 |
| 800 | ACOR | 300 | 0.36169 | 0.35637 | 0.36701 | 0.36054 | 0.33921 | 0.41281 | 0.07360 | 0.01425 | 3.93969 |
| 800 | CM1 | 100 | 0.34473 | 0.34300 | 0.34645 | 0.34505 | 0.33520 | 0.35645 | 0.02125 | 0.00462 | 1.33901 |
| 800 | CM1 | 200 | 0.26918 | 0.26627 | 0.27210 | 0.26860 | 0.25727 | 0.28804 | 0.03076 | 0.00780 | 2.89915 |
| 800 | CM1 | 300 | 0.29932 | 0.29695 | 0.30169 | 0.29788 | 0.28819 | 0.31621 | 0.02802 | 0.00635 | 2.12183 |
| 800 | ID-FSAC | 100 | 0.35161 | 0.34973 | 0.35348 | 0.35181 | 0.33925 | 0.36050 | 0.02125 | 0.00502 | 1.42677 |
| 800 | ID-FSAC | 200 | 0.27754 | 0.27511 | 0.27998 | 0.27725 | 0.26695 | 0.29167 | 0.02471 | 0.00640 | 2.30428 |
| 800 | ID-FSAC | 300 | 0.30940 | 0.30757 | 0.31122 | 0.30918 | 0.30189 | 0.32083 | 0.01894 | 0.00489 | 1.57916 |
| 800 | CM2 | 100 | 0.34606 | 0.34405 | 0.34807 | 0.34535 | 0.33573 | 0.35576 | 0.02003 | 0.00519 | 1.50017 |
| 800 | CM2 | 200 | 0.26875 | 0.26536 | 0.27213 | 0.26868 | 0.25842 | 0.28295 | 0.02453 | 0.00611 | 2.27424 |
| 800 | CM2 | 300 | 0.29880 | 0.29676 | 0.30085 | 0.29736 | 0.29003 | 0.31060 | 0.02057 | 0.00496 | 1.65863 |
| 800 | CM3 | 100 | 0.34663 | 0.34330 | 0.34995 | 0.34845 | 0.33512 | 0.35705 | 0.02193 | 0.00690 | 1.98960 |
| 800 | CM3 | 200 | 0.26866 | 0.26493 | 0.27239 | 0.26908 | 0.24110 | 0.28631 | 0.04520 | 0.00981 | 3.65215 |
| 800 | CM3 | 300 | 0.29923 | 0.29733 | 0.30113 | 0.30005 | 0.28966 | 0.31237 | 0.02271 | 0.00508 | 1.69817 |

Table A.6: Results Obtained From Running Algorithms on the NSFNET Topology at Low Loads.

| Time (s) | Algorithm | Spectrum Width (GHz) | Mean | Lower Confidence | Upper Confidence | Median | Minimum | Maximum | Range | Std.Dev. | Coef.Var. |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 800 | FSAC | 200 | 0.06992 | 0.06941 | 0.07043 | 0.06971 | 0.06804 | 0.07306 | 0.00501 | 0.00137 | 1.95684 |
| 800 | FSAC | 300 | 0.06866 | 0.06817 | 0.06914 | 0.06853 | 0.06627 | 0.07247 | 0.00619 | 0.00130 | 1.89980 |
| 800 | FSAC | 400 | 0.06592 | 0.06527 | 0.06656 | 0.06587 | 0.06201 | 0.06933 | 0.00732 | 0.00172 | 2.60220 |
| 800 | GSPR | 200 | 0.05099 | 0.05091 | 0.05106 | 0.05097 | 0.05060 | 0.05143 | 0.00082 | 0.00020 | 0.39042 |
| 800 | GSPR | 300 | 0.05186 | 0.05177 | 0.05195 | 0.05184 | 0.05128 | 0.05226 | 0.00099 | 0.00023 | 0.44514 |
| 800 | GSPR | 400 | 0.05146 | 0.05140 | 0.05153 | 0.05142 | 0.05116 | 0.05181 | 0.00065 | 0.00018 | 0.35253 |
| 800 | ACOR | 200 | 0.05442 | 0.05164 | 0.05719 | 0.05337 | 0.04479 | 0.07338 | 0.02859 | 0.00743 | 13.64853 |
| 800 | ACOR | 300 | 0.05707 | 0.05378 | 0.06035 | 0.05416 | 0.04706 | 0.08207 | 0.03501 | 0.00880 | 15.42373 |
| 800 | ACOR | 400 | 0.05594 | 0.05422 | 0.05766 | 0.05588 | 0.04589 | 0.06412 | 0.01823 | 0.00460 | 8.22857 |
| 800 | CM1 | 200 | 0.05436 | 0.05405 | 0.05467 | 0.05442 | 0.05306 | 0.05623 | 0.00317 | 0.00083 | 1.52924 |
| 800 | CM1 | 300 | 0.05165 | 0.05121 | 0.05209 | 0.05181 | 0.04943 | 0.05396 | 0.00453 | 0.00119 | 2.29574 |
| 800 | CM1 | 400 | 0.04765 | 0.04737 | 0.04794 | 0.04747 | 0.04640 | 0.04942 | 0.00301 | 0.00077 | 1.61620 |
| 800 | ID-FSAC | 200 | 0.08279 | 0.08184 | 0.08373 | 0.08294 | 0.07772 | 0.08727 | 0.00955 | 0.00253 | 3.05832 |
| 800 | ID-FSAC | 300 | 0.07757 | 0.07683 | 0.07832 | 0.07760 | 0.07320 | 0.08176 | 0.00856 | 0.00200 | 2.57605 |
| 800 | ID-FSAC | 400 | 0.07112 | 0.07043 | 0.07181 | 0.07100 | 0.06661 | 0.07619 | 0.00958 | 0.00184 | 2.59336 |
| 800 | CM2 | 200 | 0.05723 | 0.05687 | 0.05759 | 0.05730 | 0.05481 | 0.05882 | 0.00401 | 0.00095 | 1.66774 |
| 800 | CM2 | 300 | 0.05427 | 0.05401 | 0.05453 | 0.05422 | 0.05294 | 0.05566 | 0.00271 | 0.00070 | 1.29173 |
| 800 | CM2 | 400 | 0.04999 | 0.04967 | 0.05032 | 0.04998 | 0.04806 | 0.05187 | 0.00382 | 0.00086 | 1.71805 |
| 800 | CM3 | 200 | 0.07755 | 0.07681 | 0.07829 | 0.07771 | 0.07251 | 0.08125 | 0.00874 | 0.00198 | 2.55400 |
| 800 | CM3 | 300 | 0.07179 | 0.07115 | 0.07242 | 0.07157 | 0.06884 | 0.07492 | 0.00608 | 0.00171 | 2.38195 |
| 800 | CM3 | 400 | 0.06620 | 0.06556 | 0.06685 | 0.06608 | 0.06248 | 0.06972 | 0.00725 | 0.00173 | 2.60934 |

Table A.7: Results Obtained From Running Algorithms on the NSFNET Topology at Medium Loads.

| Time (s) | Algorithm | Spectrum <br> Width (GHz) | Mean | Lower <br> Confi- <br> dence | Upper <br> Confi- <br> dence | Median | Minimum MaximumRange | Std.Dev. | Coef.Var. |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 800 | FSAC | 200 | 0.21417 | 0.21339 | 0.21495 | 0.21417 | 0.20977 | 0.21949 | 0.00972 | 0.00210 |  |
| 800 | FSAC | 300 | 0.19982 | 0.19875 | 0.20090 | 0.19994 | 0.19443 | 0.20503 | 0.01060 | 0.00288 |  |
| 800 | FSAC | 400 | 0.18095 | 0.18022 | 0.18167 | 0.18081 | 0.17617 | 0.18456 | 0.00839 | 0.00190 | 1.43990 |
| 800 | GSPR | 200 | 0.20371 | 0.20365 | 0.20377 | 0.20370 | 0.20343 | 0.20417 | 0.00074 | 0.00017 | 0.08298 |
| 800 | GSPR | 300 | 0.20561 | 0.20555 | 0.20567 | 0.20560 | 0.20524 | 0.20599 | 0.00075 | 0.00016 |  |
| 800 | GSPR | 400 | 0.20213 | 0.20208 | 0.20218 | 0.20211 | 0.20191 | 0.20252 | 0.00061 | 0.00014 |  |
| 800 | ACOR | 200 | 0.20451 | 0.19925 | 0.20978 | 0.20146 | 0.18558 | 0.24388 | 0.05830 | 0.01409 | 6.88891 |
| 800 | ACOR | 300 | 0.20076 | 0.19641 | 0.20512 | 0.19838 | 0.18279 | 0.22398 | 0.04120 | 0.01166 | 5.80773 |
| 800 | ACOR | 400 | 0.20623 | 0.20069 | 0.21176 | 0.20624 | 0.18428 | 0.24869 | 0.06441 | 0.01481 | 7.18249 |
| 800 | CM1 | 200 | 0.19720 | 0.19677 | 0.19763 | 0.19729 | 0.19427 | 0.19921 | 0.00494 | 0.00115 | 0.58306 |
| 800 | CM1 | 300 | 0.18368 | 0.18317 | 0.18418 | 0.18378 | 0.17996 | 0.18619 | 0.00623 | 0.00135 | 0.73752 |
| 800 | CM1 | 400 | 0.16545 | 0.16497 | 0.16592 | 0.16538 | 0.16365 | 0.16943 | 0.00579 | 0.00127 | 0.76786 |
| 800 | ID-FSAC | 200 | 0.22041 | 0.21950 | 0.22132 | 0.22014 | 0.21451 | 0.22418 | 0.00967 | 0.00244 | 1.10840 |
| 800 | ID-FSAC | 300 | 0.20677 | 0.20554 | 0.20800 | 0.20611 | 0.20026 | 0.21379 | 0.01354 | 0.00330 | 1.59528 |
| 800 | ID-FSAC | 400 | 0.18979 | 0.18849 | 0.19109 | 0.18978 | 0.18254 | 0.19589 | 0.01335 | 0.00349 | 1.83941 |
| 800 | CM2 | 200 | 0.19982 | 0.19936 | 0.20029 | 0.19979 | 0.19734 | 0.20234 | 0.00501 | 0.00124 | 0.62085 |
| 800 | CM2 | 300 | 0.18534 | 0.18487 | 0.18581 | 0.18543 | 0.18272 | 0.18850 | 0.00579 | 0.00125 | 0.67568 |
| 800 | CM2 | 400 | 0.16630 | 0.16578 | 0.16682 | 0.16627 | 0.16372 | 0.16922 | 0.00551 | 0.00140 | 0.83961 |
| 800 | CM3 | 200 | 0.21274 | 0.21195 | 0.21353 | 0.21291 | 0.20812 | 0.21597 | 0.00785 | 0.00211 | 0.99013 |
| 800 | CM3 | 300 | 0.19865 | 0.19779 | 0.19950 | 0.19859 | 0.19436 | 0.20627 | 0.01191 | 0.00228 | 1.14967 |
| 800 | CM3 | 400 | 0.18123 | 0.18042 | 0.18204 | 0.18114 | 0.17574 | 0.18449 | 0.00875 | 0.00218 | 1.20060 |

Table A.8: Results Obtained From Running Algorithms on the NSFNET Topology at High Loads.

| Time (s) | Algorithm | Spectrum <br> Width (GHz) | Mean | Lower <br> Confi- <br> dence | Upper <br> Confi- <br> dence | Median | Minimum | Maximum | Range | Std.Dev. | Coef.Var. |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 600 | FSAC | 200 | 0.34751 | 0.34676 | 0.34826 | 0.34719 | 0.34218 | 0.35088 | 0.00869 | 0.00201 | 0.57867 |
| 600 | FSAC | 300 | 0.32201 | 0.32078 | 0.32323 | 0.32240 | 0.31426 | 0.32721 | 0.01296 | 0.00328 | 1.01785 |
| 600 | FSAC | 400 | 0.29709 | 0.29598 | 0.29820 | 0.29699 | 0.29119 | 0.30337 | 0.01217 | 0.00298 | 1.00258 |
| 600 | GSPR | 200 | 0.35376 | 0.35370 | 0.35382 | 0.35376 | 0.35346 | 0.35409 | 0.00064 | 0.00016 | 0.04600 |
| 600 | GSPR | 300 | 0.35138 | 0.35135 | 0.35142 | 0.35138 | 0.35120 | 0.35163 | 0.00043 | 0.00010 | 0.02805 |
| 600 | GSPR | 400 | 0.34959 | 0.34955 | 0.34963 | 0.34959 | 0.34942 | 0.34982 | 0.00040 | 0.00010 | 0.02740 |
| 600 | ACOR | 200 | 0.35409 | 0.34771 | 0.36046 | 0.35321 | 0.32366 | 0.39001 | 0.06635 | 0.01707 | 4.82008 |
| 600 | ACOR | 300 | 0.34798 | 0.34342 | 0.35255 | 0.34873 | 0.32661 | 0.37263 | 0.04603 | 0.01222 | 3.51062 |
| 600 | ACOR | 400 | 0.34983 | 0.34364 | 0.35601 | 0.34678 | 0.32307 | 0.39344 | 0.07037 | 0.01657 | 4.73672 |
| 600 | CM1 | 200 | 0.34223 | 0.34174 | 0.34271 | 0.34233 | 0.33978 | 0.34470 | 0.00492 | 0.00131 | 0.38180 |
| 600 | CM1 | 300 | 0.31838 | 0.31764 | 0.31912 | 0.31829 | 0.31393 | 0.32197 | 0.00804 | 0.00198 | 0.62095 |
| 600 | CM1 | 400 | 0.29488 | 0.29399 | 0.29577 | 0.29461 | 0.28916 | 0.29896 | 0.00980 | 0.00239 | 0.80989 |
| 600 | ID-FSAC | 200 | 0.35908 | 0.35814 | 0.36002 | 0.35877 | 0.35340 | 0.36312 | 0.00972 | 0.00252 | 0.70133 |
| 600 | ID-FSAC | 300 | 0.33545 | 0.33399 | 0.33690 | 0.33634 | 0.32807 | 0.34178 | 0.01370 | 0.00361 | 1.07665 |
| 600 | ID-FSAC | 400 | 0.31253 | 0.31113 | 0.31392 | 0.31249 | 0.30572 | 0.32322 | 0.01750 | 0.00367 | 1.17407 |
| 600 | CM2 | 200 | 0.34302 | 0.34244 | 0.34360 | 0.34282 | 0.34035 | 0.34598 | 0.00563 | 0.00154 | 0.45009 |
| 600 | CM2 | 300 | 0.31672 | 0.31589 | 0.31754 | 0.31671 | 0.31172 | 0.32096 | 0.00925 | 0.00220 | 0.69464 |
| 600 | CM2 | 400 | 0.29065 | 0.28978 | 0.29151 | 0.29137 | 0.28537 | 0.29575 | 0.01038 | 0.00231 | 0.79445 |
| 600 | CM3 | 200 | 0.34940 | 0.34845 | 0.35034 | 0.34956 | 0.34408 | 0.35466 | 0.01058 | 0.00244 | 0.69768 |
| 600 | CM3 | 300 | 0.32559 | 0.32453 | 0.32665 | 0.32495 | 0.31927 | 0.33231 | 0.01304 | 0.00284 | 0.87247 |
| 600 | CM3 | 400 | 0.30298 | 0.30145 | 0.30452 | 0.30232 | 0.29727 | 0.30917 | 0.01190 | 0.00372 | 1.22728 |

## Appendix B

## Statistical Methods

A brief description of the statistical methods used in the course of the research is given in the following sections.

## B. 1 Pearson Correlation Coefficient

The Pearson correlation coefficient is a measure of the strength of the linear relationship between two quantitative variables. The values obtained from this test range from -1 to 1 , with -1 suggesting strong negative correlation and 1 suggesting a strong positive correlation. A correlation of 0 indicates no relationship. More information on the calculation and interpretation of the Pearson correlation coefficient can be found in Bluman (2012)and Wilcox (2009)

## B. 2 Mann-Whitney U Test

The Mann-Whitney U test (also known as the Wilcoxon rank-sum test or WilcoxonMannWhitney test) is a non-parametric test (it does not make any assumptions about the populations distribution) that is used to determine if the means of a variable for two independent groups are significantly different from each other. The null hypothesis of the test is that the distribution of both samples are equal, and the alternative hypothesis is that they are different. This test can be used when the sample size of both groups is greater than or equal to 10 . More information on the calculation and interpretation of the Mann-Whitney U test can be found in Bluman (2012), Wilcox (2009) and Montgomery and Runger (2003)

## Appendix C

## CM-FSAC Pseudo-Code

Pseudo-code for the CM-FSAC algorithm is given below in terms of the activities of the edge nodes (Algorithm 2) and the activities of the core nodes (Algorithm 1)

```
Algorithm 1 Core node
    \{For lifetime of network\}
    loop
        if Arriving message is BCP then
            Send BCP Traversal Acknowledgement
            Perform reservation
            Forward to next node
            Start BCP failure timer
        end if
        if Arriving message is Burst then
            if Reservation exists then
                Forward to next node
            end if
        end if
        if Arriving message is successful EndToEndAck then
            Collect information using procedure of CM1 (Section 5.2)
            Forward to next node
        end if
        if Arriving message is failed EndToEndAck then
            Cancel reservation
            Forward to next node
        end if
        if Failure timer timeout then
            Cancel reservation
            Send failure EndToEndAck to source
        end if
    end loop
```

```
Algorithm 2 Edge node
Require: \(\alpha_{1}, \alpha_{2}, \gamma, \epsilon\), Number of Entries(NoE)
    \{Initialise RWA\}
    for all possible destinations do
        Initialise candidate route list \(\mathcal{K}_{s d}\)
        for \(i=1\) to \(N o E\) do
            Compute initial entries
            Create new routing tuple \(\zeta\)
            \(\varpi \leftarrow\) random wavelength center within spectrum range
            \(r^{s d} \leftarrow\) random route from \(\mathcal{K}_{s d}\)
            \(\tau \leftarrow U(0,1)\)
            \(\sigma \leftarrow 1\)
            Add \(\zeta\) to \(\Xi_{s d}\)
        end for
    end for
    \{For lifetime of network\}
    loop
        if Burst to be transmitted then
            Do RWA 3
        end if
        if Successful EndToEndACK received then
            Update congestion measure for all entries to \(d\) which use route \(r^{s d}\)
            Update success counter for entry
        end if
        if Failed EndToEndACK received then
            Update failed counter for entry
        end if
        if Burst received then
            Send successful EndToEndAck
        end if
    end loop
```

```
Algorithm 3 RWA
    \(a \leftarrow U(0,1)\)
    if \(a<\alpha_{1}\) then
        Select \(\hat{\zeta}\) using equation (5.2)
        Send BCP and burst using \(\hat{\zeta}\)
    else if \(a<\left(\alpha_{1}+\alpha_{2}\right)\) then
        Select \(\hat{\zeta}\) using equation (5.3)
        Send BCP and burst using \(\hat{\zeta}\)
    else
        Create new routing tuple \(\zeta\)
        \(\varpi \leftarrow\) random wavelength center within spectrum range
        \(r^{s d} \leftarrow\) random route from \(\mathcal{K}_{s d}\)
        \(\tau \leftarrow U(0,1)\)
        \(\sigma \leftarrow 1\)
        replace lowest performing entry in \(\Xi_{s d}\)
        Send BCP and burst using \(\zeta\)
    end if
    Return
```


## Appendix D

## Scatter Plots of Data Obtained in Parameter Studies

## D. 1 NSFNET

D.1.1 High


Figure D.1: Scatter plots of data obtained in parameter study of FSAC at high load on the NSFNET topology


Figure D.2: Scatter plots of data obtained in parameter study of ID-FSAC at high load on the NSFNET topology


Figure D.3: Scatter plots of data obtained in parameter study of CM1 at high load on the NSFNET topology


Figure D.4: Scatter plots of data obtained in parameter study of CM2 at high load on the NSFNET topology


Figure D.5: Scatter plots of data obtained in parameter study of CM3 at high load on the NSFNET topology


Figure D.6: Scatter plots of data obtained in parameter study of ACOR at high load on the NSFNET topology

## D.1.2 Medium



Figure D.7: Scatter plots of data obtained in parameter study of FSAC at medium load on NSFNET topology


Figure D.8: Scatter plots of data obtained in parameter study of ID-FSAC at medium load on NSFNET topology


Figure D.9: Scatter plots of data obtained in parameter study of CM1 at medium load on NSFNET topology


Figure D.10: Scatter plots of data obtained in parameter study of CM2 at medium load on NSFNET topology


Figure D.11: Scatter plots of data obtained in parameter study of CM3 at medium load on NSFNET topology


Figure D.12: Scatter plots of data obtained in parameter study of ACOR at medium load on NSFNET topology

## D.1.3 Low



Figure D.13: Scatter plots of data obtained in parameter study of FSAC at low load on NSFNET topology


Figure D.14: Scatter plots of data obtained in parameter study of ID-FSAC at low load on NSFNET topology


Figure D.15: Scatter plots of data obtained in parameter study of CM1 at low load on NSFNET topology


Figure D.16: Scatter plots of data obtained in parameter study of CM2 at low load on NSFNET topology


Figure D.17: Scatter plots of data obtained in parameter study of CM3 at low load on NSFNET topology


Figure D.18: Scatter plots of data obtained in parameter study of ACOR at low load on NSFNET topology

## D. 2 SIMPLE

## D.2.1 High



Figure D.19: Scatter plots of data obtained in parameter study of FSAC at high load on SIMPLE topology


Figure D.20: Scatter plots of data obtained in parameter study of ID-FSAC at high load on SIMPLE topology


Figure D.21: Scatter plots of data obtained in parameter study of CM1 at high load on SIMPLE topology


Figure D.22: Scatter plots of data obtained in parameter study of CM2 at high load on SIMPLE topology


Figure D.23: Scatter plots of data obtained in parameter study of CM3 at high load on SIMPLE topology


Figure D.24: Scatter plots of data obtained in parameter study of ACOR at high load on SIMPLE topology

## D.2.2 Medium



Figure D.25: Scatter plots of data obtained in parameter study of FSAC at medium load on SIMPLE topology


Figure D.26: Scatter plots of data obtained in parameter study of ID-FSAC at medium load on SIMPLE topology


Figure D.27: Scatter plots of data obtained in parameter study of CM1 at medium load on SIMPLE topology


Figure D.28: Scatter plots of data obtained in parameter study of CM2 at medium load on SIMPLE topology


Figure D.29: Scatter plots of data obtained in parameter study of CM3 at medium load on SIMPLE topology


Figure D.30: Scatter plots of data obtained in parameter study of ACOR at medium load on SIMPLE topology

## D.2.3 Low



Figure D.31: Scatter plots of data obtained in parameter study of FSAC at low load on SIMPLE topology


Figure D.32: Scatter plots of data obtained in parameter study of ID-FSAC at low load on SIMPLE topology


Figure D.33: Scatter plots of data obtained in parameter study of CM1 at low load on SIMPLE topology


Figure D.34: Scatter plots of data obtained in parameter study of CM2 at low load on SIMPLE topology


Figure D.35: Scatter plots of data obtained in parameter study of CM3 at low load on SIMPLE topology


Figure D.36: Scatter plots of data obtained in parameter study of ACOR at low load on SIMPLE topology

## Appendix E

## IFIP PEMWN 2017

This paper presented the design of a OBS simulator which is capable of simulating the operation of OBS on Flex-spectrum networks, in the presence of the WC constraint and physical layer impairments. The simulator is validated using an analytical model, and it is found that results obtained from the simulator closely match the results obtained from the analytical model. Finally, a study to determine the effects of flexi-grid and non-linear impairments on the performance of OBS networks is performed.

This paper was accepted for presentation at the 6th IFIP International Conference on Performance Evaluation and Modeling in Wired and Wireless Networks (IFIP PEMWN 2017).

# Implementation and Validation of an Omnet++ Optical Burst Switching Simulator 

Joshua Oladipo*, Mathys C. duPlessis*, Timothy B. Gibbon ${ }^{\dagger}$<br>*Department of Computing Sciences<br>${ }^{\dagger}$ Department of Physics<br>Nelson Mandela Metropolitan University, P.O. Box 77000, Port Elizabeth 6031, South Africa<br>${ }^{1}$ joshua.oladipo@nmmu.ac.za<br>${ }^{2}$ mc.duplessis@nmmu.ac.za<br>${ }^{3}$ tim.gibbon@nmmu.ac.za


#### Abstract

In this paper, we present a newly developed simulator based on the Omnet++ framework, for simulating Dense Wavelength Division Multiplexed, optical burst switched networks in the presence of optical impairments. The simulator is validated against a reduced link load model for optical burst switched networks, and simulations are performed in order to determine the effects of flexi-grid and impairments modelling on simulation results. We find that flexi-grid substantially reduces the burst loss probability on the simulated network, and that impairments modelling has a noticeable effect on simulation results, particularly in flexi-grid scenarios.


Index Terms-Optical Burst Switching, Omnet++, Network Simulation, Flexi-grid, Dense Wavelength Division Multiplexing, Optical Impairments

## I. Introduction

Optical Burst Switching (OBS) is a proposed optical switching technology which is seen as having the potential to bridge the technological gap between Optical circuit Switching (currently used technology) and Optical Packet Switching (future proposed technology) on Dense Wavelength Division Multiplexed (DWDM) networks; by improving the bandwidth utilisation of networks. This is required, in order to allow supporting the forecasted increased pressure on core networks due to global increases in the use of high bandwidth applications and number of persons with access to the internet, by allowing more efficient usage of available bandwidth[1].

A major obstacle to the deployment of OBS is the lack of Routing and Wavelength Assignment (RWA) algorithms which attain the Quality of Service requirements of network operators; as well as a poor understanding of the behaviour of various protocols (such as TCP and VoIP) over OBS networks. In order to remedy these issues, various researchers have investigated multiple avenues; this is typically done using a simulator, built by the researcher to determine the potential performance of newly designed algorithms. However, researchers tend not to release the code for their simulators or provide enough implementation details to allow easy replication. This reduces the credibility of published results. Also, most simulations seem to ignore the effects of optical impairments within the system, reducing the real world applicability of their results. Due to all of the above, a situation is created where there are no credible comparison of dynamic OBS RWA algorithms. In this paper, we present
a new general purpose OBS simulator which may remedy all of the above. We then present the outcome of validating the results obtained by the simulator against a reduced link load model in the presence of the wavelength continuity constraint, and determine the effects of flexi-grid and modelling of optical impairments in simulations.

The rest of the paper is structured as follows, Section II] gives a brief introduction to OBS, Section III describes the simulator's operation, Section IV introduces the reduced link load model which will be used for validation of the simulator, simulation results are presented in Section V , and finally, we give our conclusions in Section VI

## II. Optical Burst Switching

In an OBS network, packets are aggregated into bursts, at network edge nodes, according to their destination. The aggregation process may be time based, where a burst is only sent after a preset time; size based, where a burst is only sent when it reaches a given maximum length; or hybrid, where both time and size are taken into account[2].

When a burst is ready to be sent, the node sends a packet, known as a Burst Control Packet (BCP) on a dedicated channel called the control channel towards the destination of the burst, in order to reserve the required spectral bandwidth and setup the switching matrix at intermediate nodes. The burst is then sent after a given offset time. The design of the network might require that the wavelength continuity constraint is maintained, that is, a burst must travel from source to destination using the same spectrum; else the core nodes would be equipped with costly wavelength converters, allowing the burst to be sent on whatever spectrum is available on arrival to the node.

The bandwidth reservation and switching in intermediate nodes may be performed in two ways; Just-In-Time (JIT), where the intermediate nodes perform the bandwidth reservation and switching immediately after they receive the BCP[3]; and Just-Enough-Time (JET) where the BCP contains a time value, indicating when the burst is to be expected, in order to allow the intermediate nodes to perform bandwidth reservation and switching at the last possible moment[1]].

Routing might be performed at each node or might already
have been performed by the source edge node[4]. A burst may be lost during transmission due to an unsuccessful bandwidth reservation attempt, because of contention at an intermediate node; or due to linear and non-linear impairments which make it unreadable at the receiving node.

## III. Simulator Design

The simulator's design consists of three main modules; the Edge node, Core Node and Fibre. Each module encapsulates a given aspect of the operation of an OBS network. The simulator currently supports JIT scheduling, all three forms of burst assembly mentioned in section $\Pi$, modelling fixed and flexi-grid DWDM networks, and networks where the wavelength continuity constraint is imposed. The simulator's design was based on the design of an OBS simulator by Espina et al[5]. However, their design has been heavily modified and re-implemented to allow better flexibility and extensibility. Major changes include the implementation of a K-Shortest path algorithm to aid the study of RWA algorithms, and the encapsulation of the physical aspects of the network into a single module (the Fibre module) leading to better modelling of WDM networks, better segregation of roles among all the modules and the removal of redundant modules.

In this section, a brief overview of the operation of the simulator is given, followed by a detailed description of the operation of each module. Omnet++ generated Images showing the structure of modules are provided.

## A. Operational Overview

The primary role of the simulator is to facilitate the credible implementation and evaluation of OBS RWA algorithms. Towards this end, the simulator must be capable of simulating the generation and transmission of burst; route traffic based on an implemented routing algorithm; simulate DWDM systems, optical impairments and other optical phenomenon; and provide a mechanism for configuration of networks.

Edge nodes are the source and destination of bursts, and are responsible for the aggregation of packets into burst, routing of the burst, and the sending of the BCP and burst at the appropriate times (based on wavelength availability and burst offset). The chief responsibility of core nodes is the forwarding of messages from edge nodes and other core nodes towards their final destination. However, if the implemented routing algorithm is distributed, it may be implemented here. The implementation of routing algorithms usually requires access to a list of shortest paths between a node and other edge nodes, hence, every node is equipped with a list of K shortest paths between the node and every edge node (where $K \in \mathbb{N}$ ). The shortest paths are calculated at the start of the simulation using Yen's K-Shortest path algorithm[6].

After a message is sent from a node, the message then traverses the network along a fibre. If the message is a BCP, the message attempts to reserve the wavelength channel assigned to its corresponding burst before it is sent, or discarded if the reservation fails. The network can support sending burst using fixed grid or flexi-grid, as defined by the International

Telecommunications Union[7]. In the fixed grid scheme, the channel bandwidth spacing is usually set to 50 or 100 GHz , and a burst is sent on a single channel, while in the flexi-grid scheme, the channel bandwidth spacing is set to 12.5 GHz , and a burst may be sent on an arbitrarily sized bandwidth channel, made up of the concatenation of a number of the 12.5 GHz channels. In order to assign an appropriate width of flexi-grid channels to a transmission (Assuming On-Off keying modulation), the following relationship is used

$$
\Delta F * \Delta T=1
$$

Where $\Delta F$ is the channel bandwidth, and $\Delta T$ is the bit duration.

Immediately after a message traverses a fibre, The loss of power (in dB ) experienced by a transmitted signal across a length of fibre due to linear and non-linear impairments is calculated using [8]

$$
\text { Penalty }=A L+k L \sum_{i \epsilon T_{f} \backslash\{s\}} \frac{B_{s} 10^{\frac{P_{i}}{10}}}{B_{i} 10^{\frac{P_{s}}{10}}\left(f_{i}-f_{s}\right)}
$$

where constant $\mathrm{k}=4.78$ (Assuming On-Off keying modulation), A is the attenuation constant in $\mathrm{dB} / \mathrm{km}, \mathrm{L}$ is the fibre length in $\mathrm{km}, T_{f}$ is the set of signals traversing the fibre between the start and end time of the signal, $B_{s}$ is the bit rate of the signal, $B_{i}$ is the bit rate of the signal causing the interference, $P_{i}$ is the intensity of the signal causing the interference in $\mathrm{dBm}, P_{s}$ is the intensity of the signal in dBm , $\left(f_{i}-f_{s}\right)$ is the difference between the central frequencies of the interfering signal and the signal in GHz.

## B. Omnet++

The simulator was built using the Omnet++ framework[9], which is an open-source, Discrete Event Simulator framework that can be used to build simulations for different scenarios (not just communication networks). Omnet++ uses a component based architecture, which allows for easier development of a simulation as key aspects of the model can be identified and implemented in separate modules. Modules may be nested in other modules to aggregate their behaviours and communication between modules is done via messages passed between modules. Development of modules is done using C++ and a module description language call NED, which is used to define the structure, parameters and connections of modules. Omnet++ is well documented which makes the API easy to learn. It also has a sizeable active community as well as multiple open-source, community written simulation models (most notably the INET framework which models various network protocols). Omnet++ is provided on an academic license, however, a commercial version is also provided under the name Omnest, which has better support as well as features not available in Omnet++.

## C. Edge Node

The edge node (Figure 1) is modelled as a router with an OBS interface, which may then be connected to a packet generator (The packet generator currently employed is obtained


Fig. 1. Edge Node Components
from the INET framework). The edge node also has a KShortest path table which is populated with the K shortest paths between the current node and every other edge node within the simulated network.

The OBS interface consists of an assembler, disassembler and routing table. The disassembler receives messages destined for the current node and processes them as required, the routing table can be extended to fit the routing algorithm being modelled. The assembler consist of a dispatcher which sends the incoming packets to the appropriate packet burstifier depending on its destination. When a packet burstifier satisfies the conditions for burst generation, it sends the burst to the router, which generates the BCP, performs RWA and calculates the offset between the BCP and Burst, and finally sends them off towards their destination.

## D. Core Node

The core node (Figure 2) consists of 5 modules, the input module, core control logic module, optical cross connect module, K-Shortest path table and core node routing table. The core nodes chief responsibility is forwarding messages from edge nodes and other core nodes towards their final destination. However, if the routing algorithm being tested is a distributed routing algorithm, it can be implemented in the core control logic module.

The input module determines if the message received is a control message or burst and forwards them either to the optical cross connect or core control logic respectively. The core control logic deals with creating a reservation for the appropriate output port for the message. The optical cross connect component is meant to model the operation of an optical cross connect, switching messages to the right output port as has been reserved. The core node routing table and K-Shortest path table are similar to their counter parts in the edge node (except that the K-Shortest path table consists of K routes between the current node to every edge node) and may be used when implementing a routing algorithm at the core node.


Fig. 2. Core Node Module Layout

## E. Fibre

The fibre module (Figure 3) consist of three components; two Mux components representing either end of a fibre link, and a fibre table which records transmission times for use in the calculation of physical layer impairments. The fibre module is meant to encapsulate all the details of sending data over a fibre link; such as the reservation of a light path, the transmission and subsequent traversal of data over the fibre and the application of impairments and other physical layer phenomenon such as amplification, wavelength conversion etc. Separating the data transmission aspect from the nodes, allows for provision of a method to aid calculation of impairments; and independence of the algorithms implemented on the node from the physical layer design of the network, leading to improved extensibility and re-usability of modules.

When a message arrives at either end of the fibre to be sent to the other end, the following happens; if the message is a control message and the dedicated control channel is free, it is sent over the dedicated control channel and then sent on to the node if the message is readable; else if the dedicated control channel is not free, the control message is queued. If the message is a BCP, the wavelength required by it's burst is reserved if possible; the BCP is then sent to the other side. If the BCP is unreadable on the other side of the fibre or the


Fig. 3. Fibre Module Layout
wavelength reservation is unsuccessful on the other side, the reservation is not accepted and the BCP is discarded. if the message received is a burst and the required reservation has been made, it is sent on; else the burst is lost.

On sending a message the sending side records an entry in the fibretable citing the start time, end time, power and spectrum of the message it sent. On the other side, when a message is received, all entries between a given time is requested of the fibretable and used to calculate the impairment loss suffered by the received message. If the message's optical power is less than the set receiver sensitivity, and it is a control message, then it is discarded; else, if the message is a burst it is sent on to the next node (due to the transparent nature of OBS).

Aside from the components stated above, the simulator also has a component called a combi-node which consists of an edgenode and corenode put together. This is to allow data transfer from the edgenode into the network, while still allowing the corenode to fulfil its forwarding duties. This structure was created in order to improve the ease with which mesh networks can be built using the simulator. The simulator also consist of various message definition objects meant to represent BCPs, Bursts, and other control messages.

## IV. Reduced Link Load Model

The simulator was validated by comparing simulation results against analytical results obtained by using a reduced link load fixed-point approximation model for OBS networks presented in [10]. Using the model, the steady state blocking probability of each link is approximated by considering only the reduced offered load caused by the blocking on other links in the network. Blocking events in each link are assumed to occur independently from blocking events in other links.

Assuming that burst are offered to route $r$ according to a poisson process with rate $\lambda_{r}$; then the blocking probability of a link $j$, with $N_{j}$ wavelength channels and offered load $\rho_{j}$, on a network with full wavelength conversion provided at each link, is given by the Erlang-B formula

$$
B_{j}=H\left(\rho_{j}, N_{j}\right)=\frac{\rho_{j}^{N_{j}} / N_{j}!}{\sum_{k=0}^{N_{j}} \rho_{j}^{k} / k!}
$$

However, this can be simplified for a network which maintains the wavelength continuity constraint to [11]

$$
B_{j}=H\left(\rho_{j}, N_{j}\right)=\frac{\rho_{j}}{1+\rho_{j}}
$$

$\rho_{j}$ is given by

$$
\rho_{j}=\mu_{j}^{-1} \sum_{r \in \mathcal{R}: j \in r} \lambda_{r} \prod_{i=1}^{J}\left(1-I(i, j, r) \cdot B_{i}\right)
$$

where $\mathcal{R}$ is the set of all routes on the network, $\mu_{j}$ is the service rate of burst in link $j$, measured in bursts per time unit, $J$ is the number of links on the network and $I(i, j, r)$ equals one if $i, j \in r$ and link $i$ strictly, but not necessarily immediately precedes link $j$ on route $r$; or zero otherwise. The link blocking probabilities can then be obtained by the method of successive substitution. For any given vector of blocking probabilities $\mathbf{B}$, define the transformation vector $T(\mathbf{B})=\left(T_{1}(\mathbf{B}), T_{2}(\mathbf{B}), \ldots, T_{J}(\mathbf{B})\right)$ by

$$
T_{j}(\mathbf{B})=H\left(\mu_{j}^{-1} \sum_{r \in \mathcal{R}: j \in r} \lambda_{r} \prod_{i=1}^{J}\left(1-I(i, j, r) \cdot B_{i}\right), N_{j}\right)
$$

The successive substitution method begins with an initial blocking probability vector $\mathbf{B}^{\mathbf{0}}=\mathbf{1}$, and repeatedly computes $\mathbf{B}^{\mathbf{n}}=T\left(\mathbf{B}^{\mathbf{n}-\mathbf{1}}\right)$ for $n=1,2,3, \ldots$, until $\mathbf{B}^{\mathbf{n}}$ is sufficiently close to $\mathbf{B}^{\mathbf{n - 1}}$. The approximate blocking probability of bursts offered to route $r$ may then be obtained using the following expression

$$
B(r)=1-\prod_{i \in r}\left(1-B_{i}\right)
$$

and the blocking probability of an arbitrary burst is given by

$$
B=\frac{1}{\Lambda} \sum_{r \in \mathcal{R}} \lambda_{r} . B(r)
$$

where $\Lambda=\sum_{r \in \mathcal{R}} \lambda_{r}$

## V. Simulation Results

In this section we compare results obtained from the simulator against results predicted by the reduced link load model. We then go on to investigate the effects of flexi-grid and application of the optical impairments model given in Section III on a network's performance.

## A. Simulator Validation

The 7 node network along with the routes on which data burst were sent is given in Figure 4 The wavelength continuity constraint was imposed, with the wavelength used for each burst selected randomly, and optical impairments disabled. The scheduling scheme used was the JIT protocol, hence, for the reduce link load approximation $\mu_{j}$ was set to $\mu_{j}=1 /\left(E\left(S_{j}\right)+E\left(D_{j}\right)\right)$, where $E\left(S_{j}\right)$ is the mean burst service time in link $j$ and $E\left(D_{j}\right)$ is the mean offset time between BCPs and bursts in link j [10]. The transmission speeds used for burst and control channels was 10 Gbps .


Fig. 4. Validation Network Topology, with routes used in validation


Fig. 5. Graphs showing the average burst loss probability vs load, for the simulations and approximations

Packets were generated using exponentially distributed inter-arrival times, with the mean of the distribution set to $2 \mathrm{~ms}, 1.5 \mathrm{~ms}, 1 \mathrm{~ms}$ and 0.5 ms ; for a total of four load scenarios (increasing the mean interarrival time, decreases the load). The size of the generated packets were exponentially distributed with the mean set at 1.5 KB . The burst assembly scheme used was size based, with the maximum burst size being 15 kB . Each loading scenario was tested on the network with 2, 4 and 6 burst transmission channels, with each load/network combination run 30 times. The experiments were run for a simulation time of a 200 s in order to ensure that steady state was reached.

The burst loss ratio obtained for the various scenarios are presented in Figure 5 From the graphs, it can be seen that the burst loss probability decreases with decreasing load and increasing number of transmission channels; which is consistent with expectations. It can also be clearly observed that the simulated results (legend entries that end in '-sim') closely match the approximated results (legend entries that end in '-approx'), hence, we conclude that the design and implementation of the simulator provides a good representation of an OBS network.

## B. Effect of Flexi-grid and Optical Impairments

We ran simulations on the 14 node network given in Figure 6. with the wavelength continuity constraint imposed, in order


Fig. 6. NSFNET Topology (12]. Link lengths in km

TABLE I
Burst Loss Probabilities for various scenarios

| Grid type | Impairments | Load $(\mu s)$ | Burst Loss |
| :--- | :---: | :---: | :---: |
|  |  | 50 | 0.2825 |
|  |  | 40 | 0.33186 |
|  | Enabled | 30 | 0.40209 |
| Fixed |  | 20 | 0.50995 |
|  |  | 10 | 0.69779 |
|  |  | 50 | 0.25286 |
|  | Disabled | 40 | 0.29959 |
|  |  | 30 | 0.36726 |
|  |  | 20 | 0.47475 |
|  |  | 10 | 0.67442 |
|  |  | 50 | 0.2591 |
|  |  | 40 | 0.29956 |
| Flexi |  | 30 | 0.3565 |
|  |  | 20 | 0.44329 |
|  |  | 10 | 0.59352 |
|  |  | 50 | 0.22532 |
|  |  | 40 | 0.26126 |
|  |  |  |  |
|  |  | 30 | 0.31251 |
|  |  | 20 | 0.39193 |
|  |  | 10 | 0.5343 |

to determine the effects of flexi-grid and optical impairments on simulation results obtained. The wavelength used for each burst was selected randomly, and the scheduling scheme used was the JIT protocol. Bursts were sent between every edge node pair, using the shortest paths between node pairs.

Packets were generated using exponentially distributed inter-arrival times, with the mean of the distribution set to $50 \mu s, 40 \mu s, 30 \mu s, 20 \mu s$ and $10 \mu s$; for a total of five load scenarios (increasing the mean interarrival time, decreases the load). The size of the generated packets were exponentially distributed with the mean set at 1.5 KB . The burst assembly scheme used was size based, with the maximum burst size being 15 kB . Each loading scenario was applied to the network with the optical impairments enabled and disabled, using $16,50 \mathrm{GHz}$ fixed grid burst transmission channels, and $64,12.5 \mathrm{GHz}$ flexi-grid burst transmission channels. The experiments were run 30 times for a simulation time of a 200 s in order to ensure that steady state was reached.

Each burst was sent on a channel with a randomly assigned transmission speed, such that a specified percentage of all burst sent over the network traversed the network at a given speed. The transmission speeds and percentage of burst that


Fig. 7. Graphs showing the average burst loss probability vs load, for various fixed and flexi-grid scenarios
travelled over them are $10 \mathrm{Gbps} 40 \%$, $20 \mathrm{Gbps} 30 \%$, 30 Gbps $20 \%$ and $40 \mathrm{Gbps} 10 \%$. The transmission speeds of all control channels were set to 40 Gbps .

The burst loss ratio obtained for the various scenarios are presented in Table 1 and Figure 7 . From the graphs, three major observation can be made. Firstly, the burst loss probability for the fixed grid scenarios are higher than those of the flexi-grid scenarios. This confirms the assertions by [13] and [14] that flexi-grid improves bandwidth utilisation of the network, in comparison to fixed grid.

Secondly, the difference between the burst loss probability values of the scenarios where impairments are applied and where they are not, are higher for the flexi-grid scenarios (an average difference of $4.5 \%$ ) than those of the fixed grid scenarios (an average difference of $3.1 \%$ ). This implies that optical impairments have a larger impact on flexi-grid transmission than they do fixed-grid. This is consistent with expectations, due to the fact that, fixed grid channel widths were designed with ample guard-bands to limit the effect of impairments [13][14], and the number of fixed grid channels present in a network is always constant, unlike in the flexi-grid scenarios.

And thirdly, the effects of impairments, increase the higher the load for the flexi-grid scenarios, while it stays relatively constant for the fixed grid scenarios. For instance, the difference between the burst loss probability, when impairments are enabled and when they are not for the fixed grid at load $10 \mu s$ and $50 \mu s$ are $2.33 \%$ and $2.96 \%$, respectively. While for flexi-grid they are $5.9 \%$ and $3.38 \%$, respectively. This implies that, while it may be safe to discount impairments for fixed grid transmission, they cannot be neglected in flexi-grid scenarios, as the effect of impairments cannot be quantified by a constant burst loss probability. Once again, this could be attributed to the guard-bands built into fixed grid channels and the static nature of fixed-grid. All three observations confirm the validity of the simulator, as its behaviour is consistent with the literature.

## VI. Conclusion

In this paper, we have presented a new OBS simulator which we believe will be of immense value to the OBS research community, and have also presented our validation process and the results obtained from it. Our comparison of the results obtained from the simulator against those obtained from the reduced link load analytical model suggest that the simulator adequately fulfils its purpose as an OBS simulator. The experiments on flexi-grid networks, demonstrated that the simmulator behaves consistently with theorethical expectations, hence it can be concluded that the simulator accurately reflects the functioning of real world networks.

A copy of the simulator can be obtained at https://github.com/JoshuaOladipo/Flexi_Obsmodules.

## REFERENCES

[1] C. Qiao and M. Yoo, "Optical Burst Switching (OBS) - A New Paradigm for an Optical Internet," Journal of High Speed Networks, vol. 8, no. 716, pp. 69-84, 1999.
[2] T. Battestilli, H. Perros, and N. Carolina, "An Introduction To Optical Burst Switching," IEEE Optical Communications, no. August, pp. 1015, 2003.
[3] J. Y. Wei and R. I. Mcfarland, "Just-In-Time Signaling for WDM Optical Burst Switching Networks," vol. 18, no. 12, pp. 2019-2037, 2000.
[4] M. Klinkowski, J. Pedro, D. Careglio, M. Pioro, J. Pires, P. Monteiro, and J. Sole-Pareta, "An overview of routing methods in optical burst switching networks," Optical Switching and Networking, vol. 7, no. 2, pp. 41-53, 2010. [Online]. Available: http://dx.doi.org/10.1016/j.osn. 2010.01.001
[5] F. Espina, J. Armendariz, N. García, D. Morató, M. Izal, and E. Magaña, "OBS network model for OMNeT++: a performance evaluation," in Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques. ICST, 2012.
[6] J. Y. Yen, "Finding the k shortest loopless paths in a network," management Science, vol. 17, no. 11, pp. 712-716, 1971.
[7] International Telecommunication Union - ITU-T, "G.694.1 (02/2012), Spectral grids for WDM applications: DWDM frequency grid," SERIES G.694.1, pp. 1-16, 2012. [Online]. Available: http://www.itu.int/rec/ T-REC-G.694.1-201202-I/en
[8] D. K. Boiyo, T. Chabata, R. E. Kipnoo, R. Gamatham, A. Leitch, and T. Gibbon, "Crosstalk Penalties in Optical Fibre Transport for Next-Generation Flexible Spectrum Networks," in Southern Africa Telecommunication Networks and Applications Conference (SATNAC), no. September, 2015, pp. 21-24.
[9] "OMNeT++ Discrete Event Simulator - Home." [Online]. Available: https://omnetpp.org/
[10] Z. Rosberg, H. L. Vu, and M. Zukerman, "Performance Analyses of Optical Burst-Switching Networks," vol. 21, no. 7, pp. 1187-1197, 2003.
[11] H. Luo, G. Hu, and L. Li, "Performance evaluation of optical burst switching networks with limited number wavelength conversion ", vol. 5281, no. 2002, pp. 96-101.
[12] A. S. Gravett, M. C. du Plessis, and T. B. Gibbon, "Hybridising ant colony optimisation with a upper confidence bound algorithm for routing and wavelength assignment in an optical burst switching network," in Computational Intelligence (SSCI), 2016 IEEE Symposium Series on. IEEE, 2016, pp. 1-8.
[13] P. Wright, A. Lord, and L. Velasco, "The network capacity benefits of Flexgrid." Ondm, no. c, pp. 7-12, 2013. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs\{_\}all.jsp?arnumber=6524911
[14] O. Gerstel, M. Jinno, A. Lord, and S. J. B. Yoo, "Elastic optical networking: A new dawn for the optical layer?" IEEE Communications Magazine, vol. 50, no. 2, pp. 12-20, 2012.

Joshua Oladipo received his BSc(Hons) in Computer science from NMMU in 2015, and is currently an MSc Computer science student at NMMU, working on developing intelligent routing algorithms for OBS networks.


[^0]:    ${ }^{1}$ More information on the Centre for Broadband Communication can be found at http://broadband.mandela.ac.za

[^1]:    ${ }^{1}$ A brief introduction to the Pearson correlation coefficient can be found in Appendix B

[^2]:    ${ }^{2} \mathrm{~A}$ brief introduction to the Mann-Whitney U test can be found in Appendix B

[^3]:    ${ }^{1} \mathrm{~A}$ copy of the simulator can be obtained at https://github.com/bigThinking/Flexi_Obsmodules

