INVESTIGATION OF PHYSICALLY AWARE ROUTING AND WAVELENGTH ASSIGNMENT (RWA) ALGORITHMS FOR NEXT GENERATION TRANSPARENT OPTICAL NETWORKS

by

Timothy Allen Hahn

A dissertation submitted in partial fulfillment
of the requirements for the degree

of

Doctor of Philosophy

in

Computer Science

MONTANA STATE UNIVERSITY
Bozeman, Montana

April 2010

APPROVAL

of a dissertation submitted by

Timothy Allen Hahn

This dissertation has been read by each member of the dissertation committee and has been found to be satisfactory regarding content, English usage, format, citations, bibliographic style, and consistency, and is ready for submission to the Division of Graduate Education.

Dr. Brendan Mumey

Approved for the Department of Computer Science

Dr. John Paxton

Approved for the Division of Graduate Education

Dr. Carl A. Fox

STATEMENT OF PERMISSION TO USE

In presenting this dissertation in partial fulfillment of the requirements for a doctoral degree at Montana State University, I agree that the Library shall make it available to borrowers under rules of the Library. I further agree that copying of this dissertation is allowable only for scholarly purposes, consistent with "fair use" as prescribed in the U.S. Copyright Law. Requests for extensive copying or reproduction of this dissertation should be referred to ProQuest Information and Learning, 300 North Zeeb Road, Ann Arbor, Michigan 48106, to whom I have granted "the exclusive right to reproduce and distribute my dissertation in and from microform along with the non-exclusive right to reproduce and distribute my abstract in any format in whole or in part."

Timothy Allen Hahn

April 2010

DEDICATION

I dedicate this dissertation to my beautiful and loving wife Nikki. Nikki has been my rock and my strength. She has been there to love me, support me, and encourage me through every trial. She is always there for me. She is my inspiration. She makes me want to be a better man and husband.

I love you, respect you, and admire you. I could have never done this without you. I look forward to spending the rest of my life with you.

I also dedicate this dissertation to my future family. Even though our children have yet to arrive in this world, you are already in our thoughts and prayers. I hope that I give you someone to look up to and someone that you can be proud of.

ACKNOWLEDGEMENTS

I am in great debt to those who assisted in my graduate studies at Montana State University. This dissertation would have never been possible without their tremendous contributions.

First off, I would like to express my sincerest appreciation for my graduate advisor, Dr. Brendan Mumey. He has provided me with excellent guidance and has greatly increased the quality of my work. He has been extremely patient with me and supported me throughout the entire process. I am very fortunate to have Dr. Mumey as my graduate advisor.

Dr. Richard Wolff has been a tremendous asset on my graduate committee. He has provided me with valuable insight, especially with the physics related questions. His comments led to a significant improvement in my papers and this dissertation.

It has also be a pleasure working with Dr. Jian Tang, Dr. John Sheppard, and Dr. Marc Giullian. I do not think most graduate students are fortune to have such a great committee.

Several students at Montana State University have helped. Wenhao Lin provided me with an excellent foundation to start with. Andrew Albers led the development of RAPTOR's GUI. Trent Jackson and Sriharsha K. Pavan always provided thoughtful discussion in our optical research group. I think that I will miss the weekly meetings.

Nikki has spent many hours reviewing my dissertation. Sometimes, I think she has probably spent more time reading my dissertation than anyone. This was a very tedious process for her I am sure, but she was always willing to help. In many ways, I think her name should be on the title page and diploma too.

Finally, I would like to acknowledge the one who makes all things possible. God has followed me, strengthened me, and encouraged me throughout my life. My prayer is that he will walk with you in your life as he has walked with me in my life.

Funding Acknowledgment

This work was kindly supported by the National Science Foundation under Nets Award Number 0624874 and the Benjamin Fellowship. However, any opinions, findings, conclusions, or recommendations expressed herein are those of the author(s) and do not necessarily reflect the views of the NSF or the Benjamin Fellowship.

TABLE OF CONTENTS

TABLE OF CONTENTS – CONTINUED

TABLE OF CONTENTS – CONTINUED

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

LIST OF FIGURES – CONTINUED

Figure Page

LIST OF FIGURES – CONTINUED

Figure Page

LIST OF FIGURES – CONTINUED

Figure Page

LIST OF ALGORITHMS

Algorithm Page

ABSTRACT

Optical networks form the foundation of today's information infrastructure. Current generation optical networks consist largely of point-to-point electronically transmitted links which switch between nodes and repeaters. There is a trend in optical networking to move from the current generation opaque networks toward transparent networks. Transparent networks use only optical devices, eliminating the costly need for OEO conversions. Unfortunately, transparent networks present a unique challenge in maintaining acceptable signal quality levels. This research is an investigation of RWA algorithms in transparent optical networks.

We present RAPTOR, a custom built discrete event program to simulate optical networks. RAPTOR uses its physically aware modules to accurately calculate three of the dominant physical impairments. RAPTOR is fast and multi-threaded. We introduce several new performance metrics. RAPTOR enables us to study transparent optical networks in a unique and realistic manner.

We conduct an extensive performance analysis of existing RWA algorithms. We explore many different traffic models, traffic loads, signal quality, and network topologies in a comprehensive fashion. We directly compare the leading RWA algorithms in a manner has not been done before.

We studied new RWA algorithms in two fields: Dynamic Programming and Ant Colony Optimization. Our new Dynamic Programming based algorithm has the best overall performance in most scenarios. It is flexible and adapts well to all network conditions we studied. It shows good promise for future optical networks.

CHAPTER 1
INTRODUCTION

This dissertation is a presentation of my research in optical networking algorithms. The dissertation begins with some background information and then presents my results and analysis.

1.1 Context and Motivation

Optical networks form the foundation of today's information infrastructure. The main advantages of optical networks, as outlined in [5], are:

1. Huge bandwidth (over 50 Tbps (Terabits per second))
2. Low signal attenuation (as low as 0.2 dB/km)
3. Immunity from electromagnetic interference
4. High security of signal due to no electromagnetic radiation
5. No crosstalk or interference between fibers in a single cable
6. Low signal distortion
7. Low material usage, small space requirements, and cost-effectiveness
8. Low power requirements

The advantages enumerated above give optical networking an advantage over both traditional copper-wired and wireless networks. Thus, long distance communications are dominated by optical networking.

Optical networking has been around for several decades. In the 1960s, the initial purpose of fiber optics was the transmission of images through a bundle of glass fibers. [6] These fibers were extremely lossy (loss $>$ 1000 dB/km). Around 1970, the loss rates of glass fibers were reduced to below 20 dB/km. [7] Finally, in 1979 the fiber fabrication process was improved to lower the loss rates to just 0.2 dB/km [8], which is still the standard today. This improvement lead to the first generation of large scale optical networks in the 1980s.

Optical networks are capable of broadcasting multiple channels on a single fiber using a technique known as *Wavelength Division Multiplexing* (WDM). Transmission systems using up to 160 wavelengths, each running at 10 Gbps are becoming standard. This gives each optical fiber a capacity of 1.6 Tbps. Networks with additional wavelengths and higher bit rates are being researched, bringing the bandwidth closer to the theoretical limit of 50 Tbps.[9]

Current generation optical networks consist mainly of point-to-point links which switch between nodes and repeaters. All intermediate nodes convert the incoming optical signal into an electric one, process it, and then convert it back into an optical signal. This processes is called an *OEO conversion*. Networks which use OEO conversions are called *opaque*.

There is a trend in optical networking to move from opaque networks toward *transparent* networks. Transparent networks use optical switches to eliminate the need for OEO conversions. The reason for this transition is simple: electronics capable of handling Terabits of data per second are prohibitively expensive. Transparent networks are also more flexible, as they can handle multiple modulation formats, protocols, and data rates.

Unfortunately, the movement toward transparent networks creates a new problem. Traditionally, optical networks had bit error rates so low that link error performance

was never a large concern. Networks depended upon the periodic OEO conversions to restore the signal quality through the 3R functions (reshaping, retiming, and regeneration). With transparent optical networks, the signal stays in the optical domain for the entire lightpath. This distinction allows physical impairments to accumulate along the lightpath.

For this reason, future networks must consider signal quality when choosing a *Route and Wavelength Assignment* (RWA). However, minimizing the blocking probability is also important. RWA algorithms which consider signal quality are said to be *physically aware*. Most of the RWA algorithms used today do not consider signal quality. The few algorithms which do consider signal quality are relatively simple. New physically aware RWA algorithms are necessary to efficiently manage future optical networks.

There are many optical network simulation tools available, but none of them meet our needs. Many of the general network simulation software tools do not support optical networks well, if at all. Most do not model physical impairments, so they are not able to model optical networks in a realistic manner. The few that do model physical impairments do so for small point-to-point networks only. They are very slow, making it difficult to simulate networks of a reasonable size. Some of these packages are rather expensive.

To carefully analyze existing RWA algorithms and evaluate new algorithms using a wide array of performance metrics, it is necessary for us to build an optical network simulator.

1.2 Contents

The rest of this dissertation is organized as follows. Chapter 2 presents an overview of optical networking. Chapter 3 discusses Routing and Wavelength Assignment in detail. Chapter 4 presents RAPTOR, our custom-built, discrete-event, optical network simulator. Chapter 5 presents our new approach for solving the RWA problem based upon dynamic programming. Chapter 6 discusses Ant Colony Optimization and presents our ACO based approaches for RWA. Chapter 7 presents our comprehensive analysis and performance evaluation of RWA algorithms. Finally, Chapter 8 is the conclusion.

CHAPTER 2
OPTICAL NETWORKING BACKGROUND

This chapter presents a summary of the background information relevant to optical networking. The first section discusses the enabling technologies of an optical network. The next section discusses the dominant physical impairments in optical networks. The final section presents metrics used to quantify the signal quality in optical networks.

2.1 Enabling Technologies

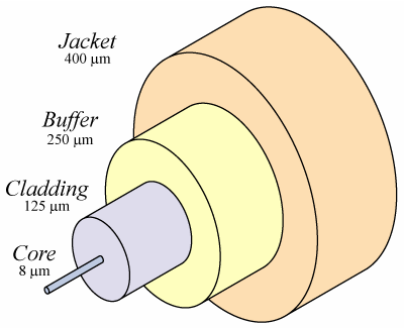
Today's optical networks are made possible by the use of several key optical technologies. This section discusses the main components of optical networks: fiber, transmitters, receivers, amplifiers, and switches.

2.1.1 Optical Fiber

Optical fiber is a thin filament of glass which acts as a *waveguide*. A waveguide is simply a transparent physical medium that allows the propagation of electromagnetic waves, such as light. Optical fibers, as shown in Figure 2.1.1 consist of two main layers: a core and a cladding. The buffer and jacket protect the fiber from damage, but are not part of the waveguide. Due to *total internal reflection*, light can propagate inside a fiber with little loss.

Light travels at the speed of $c \approx 3$ x 10^8 m/s in a vacuum, but at slower speeds through other transparent materials. If c_{mat} is the speed of light for a given material, the ratio of the speed of light in a vacuum to the speed of light in the material is known as the material's *refractive index* (n) and is modeled by $n_{mat} = c/c_{mat}$. Since

Figure 2.1: Optical Fiber [1]



$n_{mat} \approx 1.5$ for glass, the speed of signal propagation in a fiber is roughly 2 x 10^8 (3 x 10^8 / 1.5).

When light travels from a material of one refractive index to a material with a different refractive index, the angle of the light that is transmitted into the second material depends on the refractive indexes of the two materials and the angle at which the light strikes the interface between the two materials. Equation 2.1, known as Snell's Law, shows this relationship

$$n_a \sin\theta_a = n_b \sin\theta_b \tag{2.1}$$

where n_a and n_b are the refractive indexes of the two materials, θ_a the angle of incidence, and θ_b is the angle of light in the second material. However, if $n_a > n_b$ and θ_a is greater than the critical angle, all of the light rays are reflected back into the medium of a, causing total internal reflection.

The angle at which total internal reflection will take place is known as the *critical angle*. To find the critical angle for a given optical fiber, we only need to consider the refractive indexes of the core and cladding components of the optical fiber.

We can rewrite Snell's law into the form of Equation 2.2.

$$\sin\theta_{\text{clad}} = \frac{n_{\text{core}}}{n_{\text{clad}}} sin\theta_{\text{core}} \tag{2.2}$$

And from that, we derive Equation 2.3 which shows the critical angle. [2]

$$\theta_{\text{crit}} = \sin^{-1} \frac{n_{\text{clad}}}{n_{\text{core}}} \tag{2.3}$$

Attenuation in optical fiber reduces the signal power as the signal propagates over a distance. The receiver sensitivity determines the minimum power required to properly detect the signal, and thus, the maximum spacing between the transmitter and receiver or amplifier. Let $P(L)$ be the power of the optical signal at distance L from the transmitter and α the attenuation constant of the fiber, the attenuation is given by Equation 2.4 [10]

$$P(L) = 10^{-\alpha L/10} P(0) \tag{2.4}$$

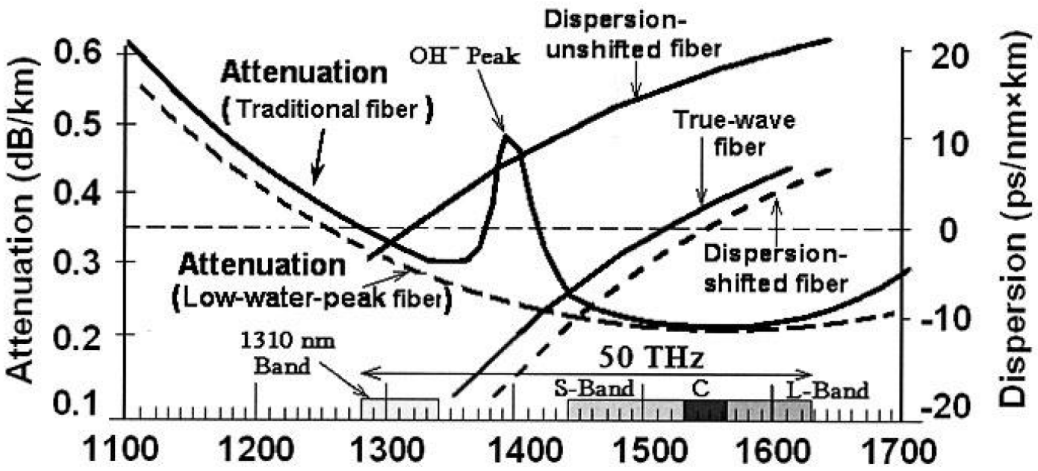
For a typical fiber, centered at roughly 1310 nm is a 200 nm region with attenuation of less than 0.3 dB/km. Centered at 1550 are three bands with attenuation as low as 0.2 dB/km. These three bands are named the *S-band* (1460-1530 nm), *C-band* (1530-1560 nm), and *L-band* (1560-1630 nm). Combined, these windows provide a theoretical limit of 50 THz of bandwidth. [2].

Most of the loss in good fibers is caused by Rayleigh scattering. However, centered around roughly 1400 nm there is a peak in attenuation caused by the hydroxyl ion (OH^-) impurities in the fiber. Figure 2.2 shows the attenuation of traditional fiber.

Dispersion is the spreading out of a pulse duration as it traverses through the optical fiber. If the pulse widens enough, it can interfere with neighboring pulses on the fiber causing intersymbol interference. For this reason, dispersion can limit the bit rate on fiber-optic channels.

Dispersion is often addressed by combining the usage of Non-Zero Dispersion-Shifted Fibers (NZDSF) with small positive dispersion rates and Dispersion Compensating Fiber (DCF) with large negative dispersion rates. If the correct length of

Figure 2.2: Fiber Attenuation and Dispersion [2]



each type of fiber is used, the total dispersion can be nearly zero. The dispersion characteristics of common types of fiber are shown in Figure 2.2.

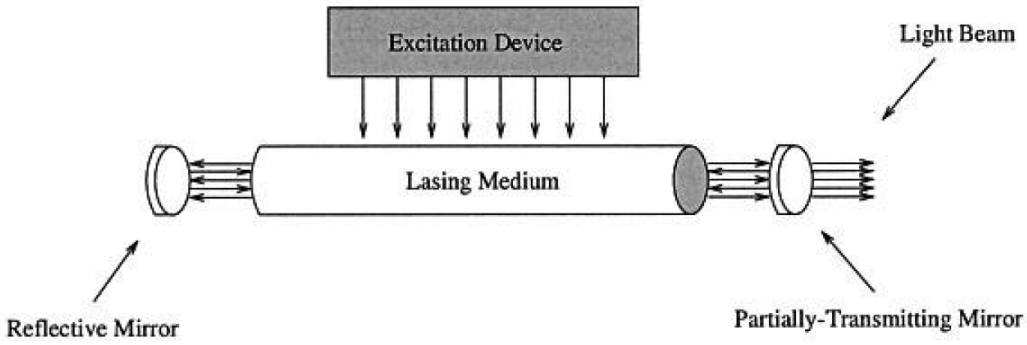
2.1.2 Optical Transmitters

Most optical networks use lasers to transmit optical signals. *Laser* is an acronym for Light Amplification by Stimulated Emission of Radiation. Stimulated Emission allows a laser to produce intense high-powered beams of coherent light. [2]

In every atom, there are a number of discrete levels of energy that an electron can have. They are typically referred to as states. Atoms which are stable have electrons at the lowest possible energy levels. Atoms move to the excited state by absorbing energy. Excited atoms are typically unstable, and usually move quickly back to the stable state by releasing a *photon*, a particle of light.

Some materials are *quasi-stable* and capable of staying in an excited state for longer periods of time. If enough energy is applied to such materials, a *population*

Figure 2.3: General Structure of a Laser [2]



inversion occurs, which means that there are more excited electrons than stable electrons. This inversion allows the material to emit more light that it absorbs.

Figure 2.3 shows the structure of a typical laser. There are two mirrors and a cavity, a lasing medium inside of the cavity, and an excitation device. The excitation device applies a current to the lasing medium, which is composed of a quasi-stable material. When a current is applied to the lasing medium and when an electron in the lasing medium drops to the stable state, a photon of light is emitted. The photon will reflect off the mirrors of the cavity and enter the lasing medium again.

Stimulated emission occurs when a photon passes very closely to an excited electron. The photon causes the electron to release its energy and return to the ground state. The electron releases another photon with the same direction and coherence of the stimulating photon. The excitation device will continue to excite the atoms in the lasing medium and the light will increase in intensity as the mirrors feed the photons back and forth.

One of the mirrors is partially transmitting, meaning some of the photons will exit the cavity in a narrowly focused beam of light. The frequency of the photon emitted depends on its change in energy levels. Equation 2.5 shows the relationship,

$$f = \frac{E_i - E_f}{h} \tag{2.5}$$

where f is the frequency of the photon, E_i is the initial state of the electron, E_f is the stable state of the electron, and h is Plank's constant. Many frequencies are possible, however, only a single frequency, determined by the cavity length, is emitted from the laser.

To transmit data across an optical fiber, the information has to be modulated, or encoded, into a laser signal. The preferred method of binary modulation is usually Amplitude Shift Keying (ASK), also known as On Off Keying (OOK). [2] Under ASK, a lower power level level represents a "0" bit and a higher power level represents a "1" bit.

The simplest way to modulate a laser using ASK is to simply turn the laser on and off. This technique is known as *direct modulation*. This leads to a signal chirping, where the amplitude and frequency variate. The preferred approach in high bit rate systems (greater than 2 Gbps) is to use an external modulator, a device which blocks or passes light depending on the current applied to it.

2.1.3 Optical Receivers

Photodetection is the process used by an optical receiver to convert the incoming optical stream into a digital stream. There are two main methods to achieve this: direct detection and coherent detection.

In systems using *direct detection*, a photodetector converts the incoming optical signal into a stream of electrons. This is accomplished through the use of PN photo-

diodes and PIN photodiodes. The electric signal is then amplified and tested against a threshold. Whether a bit is a "0" or "1" is dependent upon whether the stream is above or below the threshold. Simply put, the decision is made based on whether or not enough light is present during the bit duration.

In systems using *coherent detection*, phase information is used to encode and detect signals. This is accomplished through the use of a monochromatic laser as a local oscillator. The incoming optical stream and the signal from the oscillator are combined, resulting in a signal at the difference frequency. This difference signal is then amplified and photodetected.

Coherent detection is able to receive weaker signals under noisier conditions. However, under most optical systems it is too difficult to maintain the phase information required for coherent detection, which limits the performance of coherent detection systems. [11]

2.1.4 Optical Amplifiers

Optical signals can propagate long distances (around 80 km) before they need amplification. However, optical networks, particularly those with long links, can benefit from optical amplifiers.

Optoelectronic amplification is known as $3R$ (reamplification, reshaping, and retiming). This requires costly OEO (Optical to Electrical to Optical) conversions, however, it does provide optimal signal quality. Due to the required OEO conversions, transparent optical networks do not use optoelectronic amplifiers. In WDM (Wavelength Division Multiplexing) systems using 3R, each wavelength needs to be separated prior to amplification and recombined before transmission.

All-optical amplification is much different from optoelectronic amplification. All-optical amplification boosts only the power of the signal, but does not otherwise

restore the shape or timing of the signal. This is known $1R$ (reamplification). All-optical amplification is completely data transparent and requires no knowledge of the underlying protocols, data rates, or modulation formats.

Erbium Doped Fiber Amplifiers (EDFAs) are the most common all-optical amplifiers. Figure 2.4 shows the structure of a typical EDFA. At one end of the EDFA, a laser transmits a strong signal at a lower wavelength. The pump wavelength and original data signal are then coupled together and enter a region of erbium doped fiber. The pump signal excites the doped atoms to a higher energy level and allows the data signal to stimulate the excited atoms to release photons through stimulated emission. This effect amplifies the data signal.

Most EDFAs use pump signals with a wavelength of either 980 nm or 1480 nm. The 980 nm pump signal has shown gain efficiencies around 10 dB/mW, while the 1480 nm pump signal shows gain efficiencies around 5 dB/mW. Typical gains are on the order of 25 dB. [2]

One limitation with optical amplification is the unequal gain spectrum. Even though optical amplifiers provide a gain across a wide range of wavelengths, they do not amplify all wavelengths equally. A multiwavelength optical signal passing through a series of EDFAs will result in the power of the wavelengths being uneven. The gain spectrum of a typical EDFA is shown in Figure 2.5.

EDFAs introduce noise in the form of Amplified Spontaneous Emission (ASE). EDFAs chained together will further amplify ASE noise (and all other noise in general). Thus, the number of EDFAs in a path is often the largest contributor to poor signal quality.

EDFAs also introduce transients each time a connection is added or dropped. This is a result of the sudden increase or decrease of EDFA input power. These transients can be reduced by using a technique known as *power shaping*. [12]

Figure 2.4: Erbium Doped Fiber Amplifier

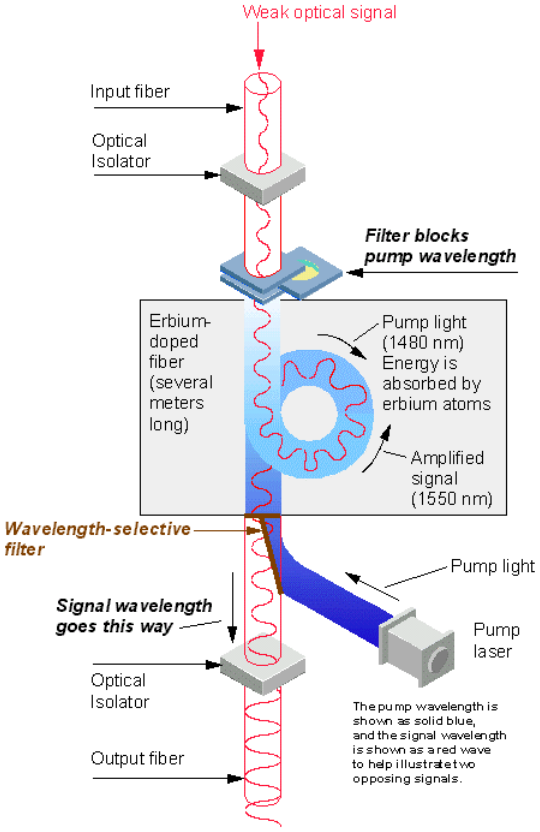
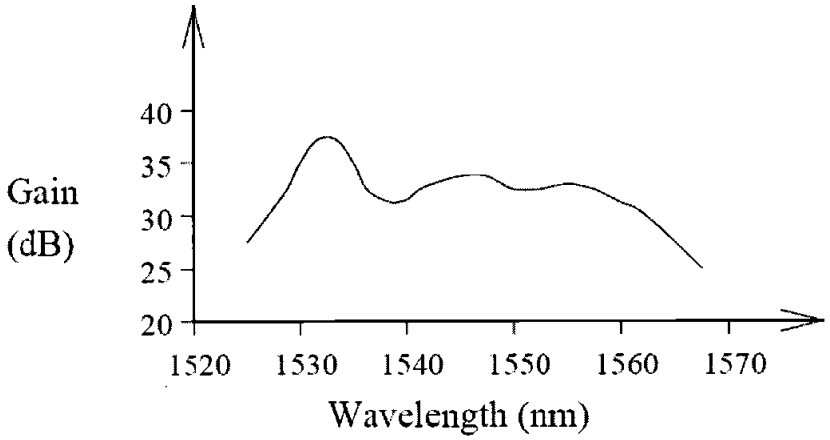


Figure 2.5: EDFA Gain [3]



2.1.5 Optical Switches

The basic function of an optical switch is to switch signals on a wavelength from an input fiber to the correct output fiber. Through the use of optical switches, optical fibers from different links can be connected and a lightpath established.

As wavelength converters are generally unavailable, the input wavelength and output wavelength must be identical. This is the reason behind the wavelength continuity constraint of RWA. The wavelength continuity constraint is the primary difference between the general network routing problem and RWA. It is the reason that RWA is NP hard, while the general network routing problem is not.

There are multiple types of optical switches available. While electronic switching is still used in opaque optical networks, transparent optical networks use optical switches exclusively. Three types of optical switches are Optical Add-Drop Multiplexer, Micro-Electro Mechanical Systems, and Wavelength Routing Switches.

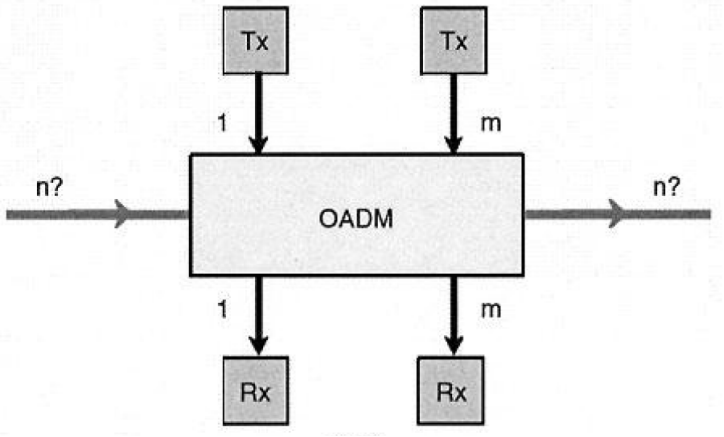
2.1.5.1 Optical Add-Drop Multiplexer:

Optical Add-Drop Multiplexers (OADMs) are devices that provide the ability to add and drop channels in the network. They accomplish this without affecting traffic that is transmitted transparently through the node and, thus, do not require OEO conversions. This introduces significant cost savings in the network. Figure 2.6 shows the generic architecture of an OADM.

They are two types of OADMs: static and reconfigurable. In static OADMs, the add/drop channels are predetermined and can only be adjusted by manually rearranging the OADM. In reconfigurable OADMs, the add/drop channels can be dynamically reconfigured as required by the needs of the network.

The Reconfigurable OADMs also fall into two categories: partly-reconfigurable and fully-reconfigurable. Partly-reconfigurable OADMs are typically used in linear

Figure 2.6: Generic OADM Architecture [2]

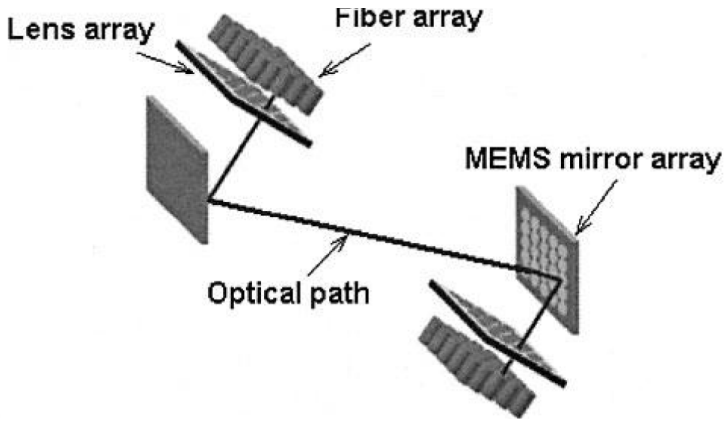


network configurations while fully-reconfigurable OADMs are typically used in ring networks and networks requiring fault-tolerance.

2.1.5.2 Micro-Electro Mechanical Systems: Micro-Electro Mechanical Systems (MEMS) based switches are distinguished by being based upon mirrors, membranes, or planar moving waveguides. MEMS based upon mirrors are a popular choice as it is suitable for compact, large-scale switching fabric. This architecture can scale to input and output port counts of over one thousand. Another advantage of this architecture is the minimal degradation of the optical signal-to-noise ratio.

Figure 2.8 shows the basic configuration of a MEMS optical switch. The optical signals from the input port are switched independently by the gimbal-mounted MEMS mirrors with a two-axis title control. The mirror focuses the light onto the optical fiber of the output port. The switch can be reconfigured by controlling the tilt angle of each mirror.

Figure 2.7: A MEMS optical switch schematic diagram [2]

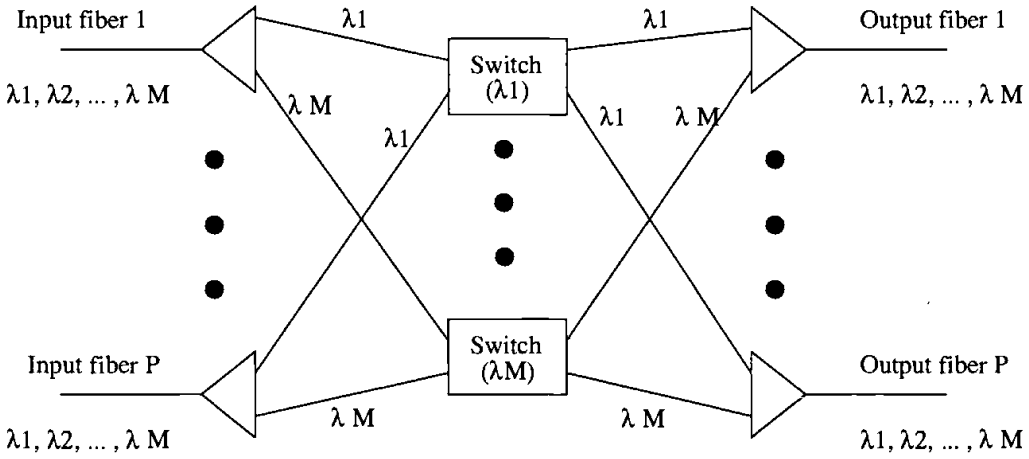


MEMS provide compact and stable optical crossconnect switches for simple, fast, and flexible wavelength applications in today's optical networks. [13] They provide very low crosstalk, wavelength insensitivity, polarization sensitivity, and scalability.

2.1.5.3 Wavelength Routing Switch: Wavelength-Routing Switches (WRS), sometimes referred to as a Wavelength Selective Crossconnect (WSXC), use photonic switches inside of the routing element. A typical WRS is shown in Figure 2.8.

A WRS has P incoming fibers and P outgoing fibers. All fibers have a capacity of M channels. The incoming wavelengths for each fiber are separated using a grating demultiplexer. The outputs of the demultiplexer are then directed to an array of MP x P optical switches. All signals on a given wavelength are directed to the same switch. The switched signals are then directed to multiplexers associated with the output ports. The signals are multiplexed together before sending them onto the output fiber.

Figure 2.8: A PxP WRS with M Wavelengths [2]



Networks built from WRSs can be constructed from components available today. [2] The WRS switch can be reconfigured each time a connection is added or dropped, thus they are extremely flexible.

2.2 Physical Impairments

While optical fibers have extremely low attenuation rates, Wavelength Division Multiplexing still leads to many optical phenomenons that degrade the signal quality in optical networks. This section discusses some of the physical impairments in optical networks.

The list of physical impairments presented here is not an exhaustive list. Optical switches and OADMs, for example, also introduce noise. However, the impairments presented in this section include the dominant causes of noise in transparent optical networks.

2.2.1 Amplified Spontaneous Emission

Amplified Spontaneous Emission (ASE) is noise produced by spontaneous emission when a laser gain medium is pumped to produce a population inversion. The power spectral density of ASE noise is

$$S(f) = 2n_{sp}(G-1)hf \tag{2.6}$$

where n_{sp} is the spontaneous emission factor, h is Planck's constant, f is the frequency, and G is the amplification gain. ASE noise is usually treated as white noise. [4]

ASE accumulates in optical spans that have multiple optical amplifiers between regenerators. Transparent networks do not use regenerators, so the number of optical amplifiers in an end-to-end path can be rather high. For this reason, ASE is often a dominant contributor to noise in all-optical networks.

2.2.2 Self Phase Modulation

In an optical fiber, the index of refraction depends slightly on the optical intensity of signals propagating through the fiber. [14] In other words, the phase of the light at the receiver is dependent upon the phase of light sent by the transmitter, the optical intensity, and the length of the fiber.

Self Phase Modulation (SPM) is caused by variations in the power of the optical signal. SPM causes variations in the phase of the signal. The phase shift introduced by SPM is given by the following equation [2]

$$\phi_{NL} = n_2 k_0 L|E|^2 \tag{2.7}$$

where n_2 is the nonlinear coefficient for the index of refraction, $k_0 = 2\pi/\lambda$, L is the length of the fiber, and $|E|^2$ is the optical intensity. SPM leads to a performance

degradation since the receiver relies on phase information. SPM also leads to a broadening of the pulse.

2.2.3 Cross Phase Modulation

Cross Phase Modulation (XPM) is a shift in the phase of a signal caused by the change in intensity of a signal co-propagating on the fiber at a different wavelength. XPM leads to asymmetric spectral broadening and may also affect the pulse shape in the time domain. [2]

Although XPM limits the performance of fiber-optic networks, there are some advantages of XPM as well. XPM can be used to modulate a pump signal at one wavelength from a modulated signal on a different wavelength, in effect, performing a wavelength conversion.

2.2.4 Four Wave Mixing

Four Wave Mixing (FWM) is an intermodulation distortion signal. When three wavelengths (λ_1, λ_2, and λ_3) interact in a nonlinear medium, such as an optical fiber, they create a fourth wavelength (λ_4) due to the scattering of the incident photons producing a fourth photon. In terms of frequency, inputs f_1, f_2, and f_3 will produce FWM on f_0 if they satisfy this relationship.

$$f_0 = f_1 \pm f_2 \pm f_3 \tag{2.8}$$

Two approaches to mitigating FWM are using fiber with a high local dispersion (such as SMF, NZDSF) or unequally spacing the channels. [15]

FWM can be used to perform an all-optical wavelength conversion. In optical networks, this technique is seldom used. Thus, most RWA algorithms assume that wavelength converters are not available.

2.2.5 Polarization Mode Dispersion

Polarization Mode Dispersion (PMD) is a complex optical effect that occurs in single-mode optical fibers. Since optical fibers are not perfectly round and free from all stress, the two perpendicular polarizations of the original transmitted signal travel at different speeds. The difference in arrival times between the two polarizations is known as PMD. The mean time differential due to PMD can be modeled by the following equation

$$\sigma_t \approx \Delta\beta_1\sqrt{2l_cL} \equiv D_p\sqrt{L} \tag{2.9}$$

where D_p is the PMD parameter, typically in the range of 0.1 to 1 $ps/\sqrt{km}$, and L is the length of the fiber. [16]

The effects of PMD can lead to bit errors at the receiver, especially with higher bit rate transmissions over long distances. [17]

2.2.6 Stimulated Raman Scattering

Stimulated Raman Scattering (SRS) is caused by the interaction of the optical signal with molecular vibrations of the fiber. A portion of the signal traveling at each frequency is downshifted across a region of lower frequencies. The light generated at the lower frequencies is called the *Stokes wave*. In a silica fiber, the Stokes wave has a maximum gain at a frequency around 13.2 THz less than the optical signal. [2]

The portion of the power that is transfered to the Stokes wave grows rapidly when the optical signal power is increased. At high power levels, nearly all of the power will be transfered to the Stokes wave.

In WDM systems, the shorter-wavelength channels lose a portion of their power to the higher wavelength channels within the Raman gain spectrum. The loss can be

minimized by keeping the power on each channel below a certain threshold. In [18], power levels below 3 mW are shown to minimize the effects of SRS.

2.2.7 Stimulated Brillouin Scattering

Stimulated Brillouin Scattering (SBS) is similar to SRS, except the shift in frequency is caused by sound waves rather than molecular vibrations. [14] Other characteristics of SBS are: 1) the Stokes wave propagates in the opposite direction of the input signal, 2) it occurs at low input powers for wide pulses (greater than 1 μs), and 3) is negligible for short pulses (less than 10 ns). [19] The intensity of the scattered light is much greater in SBS, but the frequency range of SBS is much lower than SRS.

In WDM systems, SBS can be mitigated by ensuring that the signal power is below a threshold. SBS can also induce crosstalk between channels. However, the narrow gain bandwidth of SBS makes the crosstalk relatively easy to avoid.

2.3 Quality Measurement

Traditionally, signal quality was measured by the Optical Signal to Noise Ratio (OSNR) or the Bit Error Rate (BER). BER and OSNR measurements are not sufficient for optical networks, as optical networks have BERs below 10^{12}. A measurement of nearly error-free BER can take hours or even days. [20]

In optical networks, quality is usually measured via the Q-factor. The Q-factor is calculated as

$$Q = 10 \log_{10} \frac{I_1 - I_0}{\sigma_1 + \sigma_0} \tag{2.10}$$

where I_0 and I_1 are the photo-current received at the destination and σ_0 and σ_1 is the standard deviation at the destination when a "1" and "0" are transmitted. [21] Once

the Q-factor is calculated, the BER can be approximated using the formula below. This formula makes a Gaussian noise assumption. [22]

$$BER \approx 0.5 erfc \frac{Q}{\sqrt{2}} \tag{2.11}$$

To avoid making assumptions on data modulation and their respective vulnerability to noise, we use the Q-factor exclusively in our research. Quality thresholds and connection quality are both measured in terms of their Q-factor.

CHAPTER 3

ROUTING AND WAVELENGTH ASSIGNMENT

This chapter contains a detailed discussion on Routing and Wavelength Assignment (RWA). The first section formally defines the RWA problem. The next section details the complexity of the problem. The remainder of the chapter presents several known algorithms for solving RWA.

3.1 RWA Problem Definition

The general objective of the Routing and Wavelength Assignment (RWA) problem is to maximize the number of established connections on an optical network. Each connection request must be given a route and wavelength. The wavelength must be consistent for the entire path, unless the usage of wavelength converters is assumed. Two connections requests can share the same optical link, provided a different wavelength is used.

RWA is a problem for WDM optical networks in general. The formal definition and complexity proofs presented here apply to both opaque and transparent optical networks.

The RWA problem can be formally defined in a Integer Linear Program (ILP). The ILP formulation given here is taken from [23].

$$\textbf{Maximize} : C_0(\rho, q) = \sum_{i=1}^{N_{sd}} m_i \tag{3.1}$$

subject to

$$m_i \geq 0, integer, i = 1, 2, ..., N_{sd} \tag{3.2}$$

$$c_{ij} \in 0, 1, i = 1, 2, ..., P, j = 1, 2, ..., W \tag{3.3}$$

$$C^T B \leq 1_{W \times L} \tag{3.4}$$

$$m \leq 1_W C^T A \tag{3.5}$$

$$m_i \leq q_i \rho, i = 1, 2, ..., N_{sd} \tag{3.6}$$

Integer Linear Programming is known to be NP Complete, in contrast to the more general Linear Programming which can be solved polynomially. The purpose of the ILP is to present a formal definition of the RWA problem, not a method to solve the problem (although many have attempted to do so [24]).

N_{sd} is the number of source-destination pairs, while m_i is the number of connections established for each source-destination pair. L is the number of links and W is the number of wavelengths. P is the set of paths to route connections. $A : P \times N_{sd}$ is a matrix which shows which source-destination pairs are active, $B : P \times L$ is a matrix which shows which links are active, and $C : P \times W$ is a route and wavelength assignment matrix.

Equation 3.1 represents the total number of connections in the network. Equation 3.2 ensures that the number of connections per source destination pair is a non-negative integer. Equation 3.3 limits the C matrix values to either 1 (active) or 0 (inactive). Equation 3.4 ensures that each wavelength is used only once. Equations 3.5 and 3.6 ensure that the number of established connections are less than or equal to the requested connections.

Note that the above formulation assumes that the traffic demands are known *a priori*. This type of problem is known as Static Lightpath Establishment (SLE). We are more interested in the Dynamic Lightpath Establishment (DLE) where traffic demands are not known ahead of time. This requires a dynamic algorithm to adapt to varying traffic requests over time.

The above formulation also does not consider the signal quality. We are interested in solving the more complicated physically aware RWA problem, where algorithms consider signal quality and attempt to find a route and wavelength with satisfactory quality.

3.2 RWA Complexity

It is shown in the subsections below that the SLE problem is NP complete. Thus it is unlikely that an polynomial algorithm exists for optimally solving the SLE problem. Given that the physically aware DLE problem is at least as difficult as SLE, there is little hope for an efficient algorithm to optimally solve physically aware DLE.

Note that many of the optical impairments are nonlinear, so a standard shortest path algorithm can't be used to solve them optimally even if we know the exact state of the network. This is often not a safe assumption, so solutions need to be efficient using only limited network information.

Given the complexity of RWA, there are two general methodologies for approximating the problem. One method is splitting the problem into two subproblems. The route is calculated first and a wavelength is assigned second. Four types of route selection are Fixed Path Routing, Fixed Alternate Routing, Adaptive Routing, and Fault Tolerant Routing. The second approach is to consider both route selection and wavelength assignment jointly.

3.2.1 Integral Multi-Commodity Flow

The original NP Completeness proof for the SLE problem involved a reduction to the integral multi-commodity flow (IMCF) problem in 1976. The IMCF problem was

proved to be NP complete in [25] when the number of commodities is greater than or equal to two.

It is easy to see how the SLE problem can be reduced to the IMCF problem, and vice versa. The traffic requests of SLE correspond to the requested flows of IMCF. The edges in SLE each support a finite integer of channels, while the edges in IMCF have a capacity that is a finite integer. The formal proof is omitted.

While the IMCF problem is NP Complete, the general Multi-Commodity Flow (MCF) problem can be solved efficiently using linear programming methods. [26] One method of approximating SLE is to solve the MCF problem and then use a technique called path stripping to convert the real edge flows into integral edge flows. [2]

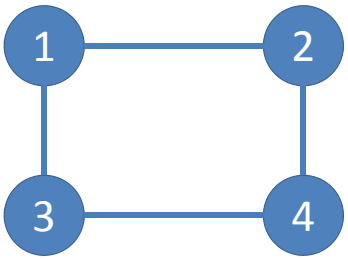
3.2.2 n-graph Colorability

The SLE problem can also be reduced to the n-graph-(vertex)-colorability problem. The reduction, presented here, is taken from [27].

First, let us show that solving the n-graph-(vertex)-colorability problem would also solve the SLE problem. Let us create an undirected graph $G_{SLE}(V_{SLE},E_{SLE})$. A vertex $v \in V_{SLE}$ is created for each lightpath. Two verticies, v_1, $v_2 \in V_{SLE}$ have an edge between them $e \in E_{SLE}$ if and only if the respective lightpaths of v_1 and v_2 have at least one link in common. Any coloring of V_{SLE} with n colors such that no two adjacent verticies share the same color would also define a wavelength assignment W where no two lightpaths having a link in common would share a lightpath. Thus, a feasible coloring in V_{SLE} is also a feasible wavelength assignment, solving SLE.

To complete the proof, we need to show that solving SLE would also solve the n-graph-(vertex)-colorability problem. First, a polynomial time algorithm is given that translates any coloring problem into a network and a set of lightpath demands. Given a graph $G_C(V_C,E_C)$, let us create a graph $G_{SLE}(V_{SLE},E_{SLE})$ as follows:

Figure 3.1: n-colorability graph

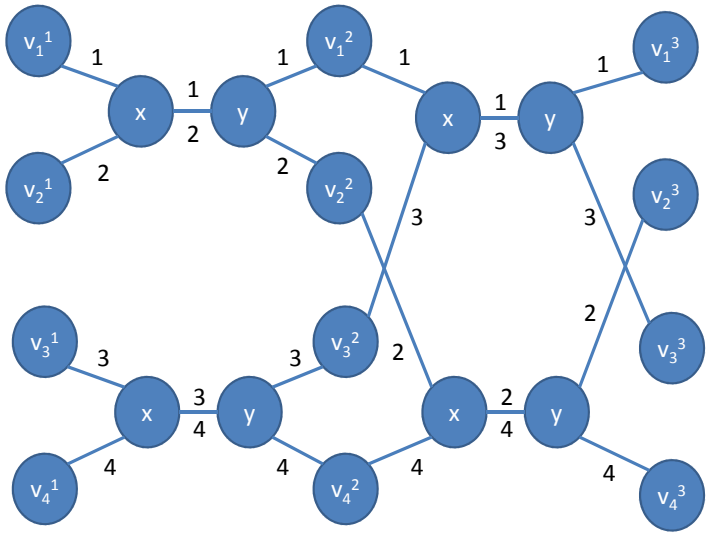


1. create a vertex $v_i^0 \in V_{SLE}$ for all verticies $v \in V_C$
2. for each edge $e = i \rightarrow j \in E_C$:
   - create four new verticies: x, y, v_i^k, $v_j^l \in V_{SLE}$
   - create five new directed edges: $v_i^{k-1} \rightarrow x$, $v_j^{l-1} \rightarrow x$, $x \rightarrow y$, $y \rightarrow v_i^k$, $y \rightarrow v_j^l \in E_{SLE}$
   - attach a label i to edges going from/to v_i and $x \rightarrow y$
   - attach a label j to edges going from/to v_j and $x \rightarrow y$

Each vertex $v_i^k \in V_{SLE}$ corresponds to the kth replication of the node $i \in V_C$. As an example, a simple four node G_C is included in Figure 3.1. The corresponding G_{SLE} is shown in Figure 3.2. The lightpath demand set L is defined by the $|V_C|$ lightpaths for which lightpath i requires the usage of all links having the label i. Note that the complexity of the transformation is O($|E_C|$).

To complete the proof, one must show that any solution to G_{SLE} with n wavelengths implies that the chromatic number of G_C is less than or equal to n. This follows immediately from the construction of the graph. If the lightpaths can be established then a function must exist which assigns wavelengths to each lightpath such

Figure 3.2: Translation to SLE



that no lightpaths sharing a link are assigned the same wavelength. Two lightpaths share a link if and only if the respective nodes in G_C are adjacent. This implies the existence of a function which assigns colors to each node in V_C, such that no two adjacent nodes are assigned the same color.

3.3 Fixed Path Routing

Fixed Path Routing is the simplest approach to finding a lightpath. The same fixed route for a given source and destination pair is always used. Typically this path is computed ahead of time using a shortest path algorithm, such as Dijkstra's Algorithm. While this approach is very simple, the performance is usually not sufficient. If resources along the fixed path are in use, future connection requests will be blocked even though other paths may exist.

The SP-1 (Shortest Path, 1 Probe) algorithm is an example of a Fixed Path Routing solution. This algorithm calculates the shortest path using the number of optical switches as the cost function. A single probe is used to establish the connection using the shortest path. The running time is the cost of Dijkstra's algorithm: $O(n^2 + m)$, where m is the number of edges and n is the number of edges. The running time is just a constant if the paths are precomputed.

1. Set the cost of each link L using $cost(L) = 1$
2. Return the minimal cost path using a shortest path(s) algorithm

Algorithm 3.1: Shortest Path Algorithm (SP)

This definition of SP-1 uses the hop count as the cost function. The SP-1 algorithm could be extended to use different cost functions, such as the number of EDFAs. In practice, there is little difference in results between the two cost functions.

3.4 Fixed Alternate Routing

Fixed Alternate Routing is an extension of Fixed Path Routing. Instead of having just one fixed route for a given source and destination pair, several routes are stored. The paths can be attempted in a serial or parallel fashion. For each connection request, the source node attempts to find a connection on each of the paths. If all of the paths fail, then the connection is blocked. If multiple paths are available, only one of them would be utilized.

The SP-p (Shortest Path, p Probes, $p > 1$) algorithm is an example of Fixed Alternate Routing. This algorithm calculates the p shortest paths using the number of optical routers as the cost function. The running time using Yen's algorithm [28] is $O(pn(m + n \log n))$ where m is the number of edges, n is the number of routers,

and p is the number of paths. The running time is a constant factor if the paths are precomputed.

In some cases, having as few as 2 alternate routes leads to better performance than Fixed Routing with full wavelength switching. [29].

3.5 Adaptive Routing

The major issue with both Fixed Path Routing and Fixed Alternate Routing is that neither algorithm takes into account the current state of the network. If the predetermined paths are not available, the connection request will become blocked even though other paths may exist. Another disadvantage is Fixed Path Routing and Fixed Alternate Routing are both not physically aware. For these reasons, most of the research in RWA is currently taking place in Adaptive algorithms. Six examples of Adaptive Routing are LORA, PABR, IA-BF, IA-FF, QM, and AQoS.

Adaptive algorithms fall into two categories: traditional and physically-aware. Traditional adaptive algorithms do not consider signal quality, however, physically aware adaptive algorithms do.

3.5.1 Traditional Adaptive RWA

The Lexicographical Routing Algorithm (LORA) algorithm was proposed in [30]. The main idea behind LORA is to route connection requests away from congested areas of the network, increasing the probability that connection requests will be accepted. This is accomplished by setting the cost of each link to be

$$cost(l) = \beta^{usage(l)} \tag{3.7}$$

where β is parameter that can be dynamically adjusted according to the traffic load and $usage(l)$ is the number of wavelengths in use on link l. A standard shortest path

algorithm can then be used to find the shortest path(s). This requires each optical switch to broadcast recent usage information periodically. Note that LORA does not consider any physical impairments.

When β is equal to one, the LORA algorithm is identical to the SP algorithm. Increasing the value of β will increase the bias toward less used links. The optimal value of β can be calculated using the well-known hill climbing algorithm. In [4], the optimal values of β were between 1.1 and 1.2.

1. Determine the appropriate value of β according to the current network traffic load
2. Set the cost of each link L using $cost(L) = \beta^{U_l}$
3. Return the minimal costs paths using a shortest path(s) algorithm

Algorithm 3.2: Lexicographical Routing Algorithm (LORA)

3.5.2 Physically Aware Adaptive RWA

The Physically Aware Backward Reservation Algorithm (PABR) is an extension of LORA. [31] PABR is able to improve performance in two ways: by considering physical impairments and through improved wavelength selection. As PABR searches for an optical path, paths with an unacceptable signal quality due to linear impairments are pruned. In other words, PABR can be formalized as

$$\textbf{Minimize} : \sum_{l \in pathP} \beta^{usage(l)} \tag{3.8}$$

subject to

$$\sum_{l \in pathP} ASEnoise(l) < threshold \tag{3.9}$$

Note that PABR only considers ASE noise. The algorithm's basic idea, however, could be extended to include all linear impairments, such as Polarization Mode Dispersion.

The nonlinear impairments, on the other hand, are difficult to estimate because they can not be represented as a sum of individual edge costs. Nonlinear impairment calculations also require information not available to a distributed algorithm.

PABR also considers signal quality when making the wavelength selection. It accomplishes this by removing from consideration all wavelengths with an unacceptable signal quality level. The approach is called Quality First Fit and it is discussed in the following section.

It should also be noted that both LORA and PABR can be implemented with either single-probing or multi-probing. The maximum number of probes p is denoted as LORA-p or PABR-p. With single-probing, only one path is selected by the route selection. With multi-probing, multiple paths are attempted serially or in parallel, increasing the probability of connection success.

1. **while** $\mathcal{H}$ is not empty **do**
2. P = FirstElement($\mathcal{H}$)
3. **if** P==D **then**
4. **if** number of paths in the result buffer < required number **then**
5. put the path from S to D in the result buffer
6. **else**
7. **return** result buffer
8. **end if**
9. **else**
10. **for** each adjacent node a_i of P **do**
11. **if** $a_i \notin$ S $\rightarrow$ P && ASE(S $\rightarrow a_i$) < threshold **then**
12. a_i.parent = P
13. a_i.cost = P.cost + $\beta^{usage(P->a_i)}$
14. insert a_i into $\mathcal{H}$
15. **end if**
16. **end for**
17. **end if**
18. **end while**

Algorithm 3.3: Physically Aware Backward Reservation Algorithm (PABR)

The Impairment Aware Best Fit (IA-BF) algorithm was proposed in [32]. IA-BF is a centralized algorithm requiring global complete knowledge of the network's resources. IA-BF uses serial multi-probing. The shortest available path and wavelength are attempted first, and upon failure, the second shortest available path and wavelength are attempted. This process continues until a successful path and wavelength have been found or all wavelengths have been attempted.

The multi-probing approach will allow IA-BF to significantly outperform both PABR-1 and LORA-1 in terms of blocking probability. However, as the number of probes increases to 4, the performance gap between the algorithms shrinks. In most scenarios, IA-BF has the lowest blocking probability. However, the serial nature of IA-BF leads to longer connection setup times.

1. **for** each wavelength w **do**
2. cost(w) = shortest path in w
3. **end for**
4. m = index with minimum value in cost
5. send probe message using wavelength m to destination to estimate signal quality
6. **if** response is acceptable **then**
7. reserve network resources and establish connection
8. **else**
9. **if** additional wave available **then**
10. Goto Step 4
11. **else**
12. Connection Request Fails
13. **end if**
14. **end if**

Algorithm 3.4: Impairment Aware Best Fit (IA-BF)

Impairment Aware First Fit (IA-FF) is a simple extension of IA-BF. [32] Instead of picking the wavelengths in terms of the minimum cost, the wavelengths are selected in order according to their index. IA-BF tends to outperform IA-FF under most scenarios.

1. Initialize w to 1.
2. Apply shortest path algorithm to find a path P_w. If no path is found, increment w and repeat step. If all wavelengths have been searched, stop and goto step 5.
3. Attempt to setup the connection using P_w.
4. If successful, then begin transmission. Otherwise, increment w and goto step 2.
5. Connection is blocked

Algorithm 3.5: Impairment Aware First Fit (IA-FF)

Quality Measurement (QM) was proposed in [20]. The algorithm is unique in the fact that it uses the Q-factor degradation as the link cost. The cost of the i_{th} link is calculated by this formula

$$D_i = \frac{\sum_{j=1}^{N_i} 10 \log[Q_{i,j}^{(s)}/Q_{i,j}^{(d)}]}{N_i} \tag{3.10}$$

where N_i is the number of lightpaths on the i_{th} link, $Q_{i,j}^{(s)}$ and $Q_{i,j}^{(d)}$ are the Q-factor measurements of the j_{th} lightpath at the source and destination nodes of the i_{th} link, respectively. The repeated Q-factor estimations are computationally very expensive, so QM is not as scalable as the other RWA algorithms.

In situations where wavelength blocking dominates quality blocking, QM performs very poorly. This is due to QM's single objective of minimizing Q-factor degradation, which ignores the availability of wavelengths. [20]

1. Initialize link costs.
2. Use a standard shortest path algorithm to compute the shortest paths.
3. If the paths are acceptable, setup the connection and update the link costs.
4. If the path is not acceptable, the link costs do not need to be updated.
5. When the connection is destroyed, link costs will again need to be updated.

Algorithm 3.6: Quality Measurement (QM)

Adaptive Quality of Service (AQoS) was proposed in [20] as an extension to QM. AQoS works similarly to QM, but with minor changes to avoid the wavelength availability issues. This algorithm is unique in that each node maintains two counters:

N_{BER} and N_{wave}. The purpose of each counter is to determine which issue is a bigger factor in blocking: Path and wavelength availability or Quality requirements. The algorithm chooses routes differently based upon the dominant issue.

1. Initialize link costs.
2. Initialize both N_{BER} and N_{wave} for all nodes.
3. For all connection requests, consider N_{BER} and N_{wave}.
4. If $N_{BER} \geq N_{wave}$, select the route that gives the least Q-degradation (i.e. use QM).
5. Else if $N_{BER} < N_{wave}$, calculate the k shortest paths and choose the path with the most available wavelengths.
6. If the request is successful, accept it. Update the link weights.
7. If the request is not successful, block it. Update either N_{BER} or N_{wave}.
8. When the connection is finished, update the link weights.

Algorithm 3.7: Adaptive Quality of Service (AQoS)

Both QM and AQoS are single probing algorithms. ALT-QM and ALT-AQoS are multi-probing extensions in which multiple probes are selected. To be consistent with our naming convention, we will refer to these algorithms as QM-p and AQoS-p.

3.6 Fault Tolerant Routing

In some optical networks, there is a desire to setup multiple edge-disjoint routes to provide protection against link and node failures in the network. [29, 33] Fixed alternate routing provides one method of setting up primary and secondary routes, provided that the routers are link disjoint.

The problem is more difficult for adaptive routing schemes. One way of providing fault tolerance is to setup the primary route as usual. A secondary or backup route can then be setup using the same scheme but with the primary route's edges removed from the graph.

Another scheme of providing fault tolerance is restoration. Under restoration, if the primary path fails, the same routing algorithm is used to search for a secondary path. Since one has not been reserved, it may be the case that a secondary path is not available. Thus there is a trade-off with restoration: no network resources are reserved for secondary links and fault tolerance is not guaranteed. Restoration will result in a lower blocking probability.[2]

3.7 Wavelength Assignment

Two of the most common methods for wavelength assignment are First Fit (FF) and Random Fit (RF). First Fit chooses the available wavelength with the lowest index. Random Fit, sometimes called Random Pick (RP), determines which wavelengths are available and then chooses randomly amongst them. The complexity of both algorithms is $O(w)$, where w is the number of wavelengths.

The general consensus is that First Fit outperforms Random Fit with regard to blocking probability due to its ability to tightly pack wavelengths together. Random Fit spreads connections across wavelengths, which results in a higher average signal quality.

First Fit can be modified slightly to order wavelengths by their frequency separation, not their index. This is called First Fit with Ordering (FFwO). Two other approaches are Least Quality (LQ) and Most Quality (MQ), where the wavelength with the Least/Most quality is used.

An extension of several wavelength algorithms was proposed in [4] which considers signal quality. Quality First Fit (Q-FF) and Quality Random Pick (Q-RP) eliminate from consideration wavelengths which have an unacceptable signal quality. A similar technique could be used for Quality First Fit with Ordering (Q-FFwO). The com-

plexity of these algorithms is higher though, as up to w calls to estimate the Q-factor are required. Some networks may not have the ability of information to make this calculation.

There are several other wavelength assignment algorithms: Least Used, Most Used, Min Product, Least Loaded, Max Sum [34], and Relative Capacity Loss [35]. Most Used outperforms Least Used significantly, and slightly outperforms First Fit [35]. Min Product, Least Loaded, Max Sum, and Relative Capacity Loss all try to choose a wavelength that minimizes the probability that future requests will be blocked.

A significant disadvantage of these algorithms is that they require a significant communication and computation overhead, making them practical only in a centralized SLE situation. DLE networks typically use First Fit.

3.8 Joint Routing and Wavelength Assignment

An alternate approach to selecting a route and wavelength separately is to consider them jointly. These approaches tend to be more theoretical and not very practical. As this is a NP-complete problem, any exact solution is likely not possible. The approximation techniques usually are not very useful either, as they require centralized control and, usually, predefined traffic demands. One joint approach, using multicommodity flows, has already been referenced. Two additional joint approaches are ILP formulation and Island Hopping.

The ILP formulation listed previously in this chapter can be solved using a traditional ILP solver. This is typically done by temporarily relaxing the integer constraints, solving the problem optimally, and converting the real solution to an integer

solution. Additional constraints can be added and the process is repeated indefinitely using a branch and bound approach.

Another approach presented in [36] is called Island Hopping and Path Coloring. This paper does not attempt to solve RWA but gives solutions for two related problems: fiber minimization (MinFib) and hop minimization (MinHop).

It is often the case that multiple fibers exist between two optical nodes, but only a subset of the fibers are actually in use. This is due to economics: the cost of laying each additional fiber after the first is minimal, but the cost of deploying the electronics necessary for each fiber is high. The goal of MinFib is thus to minimize the total number of fibers required to meet a traffic demand. MinFib has only limited applicability to RWA.

Optical switches are categorized by a degree which measures their ability to switch optical signals [37]. If the degree of the switch is too low, then an OEO conversion is required for some connections. MinHop is an algorithm that segregates the network into transparent islands. Each transparent island can switch optical signals without an OEO conversion, but each hop between islands would require an OEO conversion. The goal of MinHop is to minimize the number of hops, i.e. OEO conversions.

In [36], several proofs are given to show the complexity of MinHop using a graph theory approach. While it is trivial to $O(n)$ approximate MinHop, it is difficult to do much better. With the increasing degree of optical switches, this problem is becoming less important. Future networks will have full switching capabilities, so OEO conversions will not be necessary. [37]

CHAPTER 4

RAPTOR

RAPTOR (Route Assignment Program for Transparent Optical Routes) is a custom-built, discrete-event, transparent optical network simulator. RAPTOR is physically aware and capable of modeling large optical networks with up to 1000 wavelengths. There are roughly 11,000 lines of source code.

This Chapter discusses RAPTOR in detail. First, we discuss the necessity of RAPTOR. The next section presents the Physically Aware Modules. The following section discusses the simulation architecture and important classes of RAPTOR. The next section presents the configurable inputs of RAPTOR. The final section presents some results from RAPTOR, as an example of its capabilities.

4.1 Motivation Behind RAPTOR

There is a need for a software tool capable of modeling and simulating physically aware transparent optical networks to determine the performance of existing RWA algorithms. There are many optical network simulation tools available, but none of them met our needs completely.

SIMON is a physically aware optical network simulator presented in [38]. SIMON's quality estimation modules are very simple, focusing only on noise introduced by optical switches. The paper is vague in general and provides no presentation of results or conclusions from SIMON. The paper does not appear to be cited by another paper.

Many of the general network simulation software tools, such as Opnet and ns2, do not support optical networks well, if at all. The few simulators that are physically aware, such as PhotoSS and OptSim, are only capable of modeling tiny networks with

point-to-point links. None of these software packages are capable of modeling large scale physically aware transparent optical networks.

To make matters worse, these tools are (except for ns2 and SIMON) very expensive. Little support is provided for extending the packages to suit one's needs. As we are interested in adding new statistics to complete an extensive performance analysis of existing RWA algorithms, it became clear that we needed to write the software package ourselves. RAPTOR is an integral part of the research presented in this dissertation.

4.2 Physically Aware Modules

RAPTOR currently models three physical effects: Amplified Spontaneous Emission (ASE), Four Wave Mixing (FWM), and Cross Phase Modulation (XPM). ASE, FWM, and XPM are considered because they have the most significant influence on signal quality. [21] These modules are a key advantage of RAPTOR as they allow for a realistic modeling of a transparent optical network where the effects of the dominant physical impairments can be observed. Each of the models are described in detail in this section.

Extensive analysis and simulation in [4] has shown that the modules presented here present an accurate estimation of both FWM and XPM. I modified the code for optimization purposes, but took great care to ensure that the accuracy of the modules was not effected.

Both the XPM and FWM modules assume the usage of NZDSF (Non-Zero-Dispersion Shifted Fiber) and DCF (Dispersion Compensating Fiber) to compensate for dispersion. The DCF portion of the fiber is omitted from the XPM and FWM

calculations, as the nonlinear effects in DCF were shown to be negligible in [39] due to the large attenuation in the preceding NZDSF.

Both FWM and XPM use a nonlinear sliding window to limit the complexity of the calculation. In other words, active channels outside of the nonlinear window are not considered. If the window size is large enough, this assumption has little effect on the results as the nearer active channels dominate the total noise contribution. A typical value for the window size is 40 channels, 20 on each side of the center frequency. With a 50 GHz channel spacing, this represents a windows size of 2 THz. If the channel spacing is decreased, the window size should be increased.

4.2.1 Amplified Spontaneous Emission

The noise from Amplified Spontaneous Emission (ASE) is often a dominant factor in the signal quality. [2] An input parameter specifies the amount of noise that each amplifier contributes. The simple calculation below is used to calculate the ASE noise on frequency f for an end-to-end path.

$$\sigma^2_{ASE} = \text{NumberOfAmps} * \text{ASEPerAmp}(f) \tag{4.1}$$

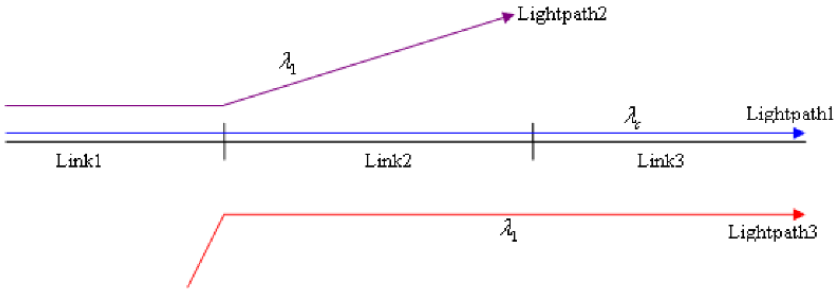
The ASEPerAmp can be computed using Equation 4.2

$$\text{ASEPerAmp}(f) = 2P_{ch}F_n hf(G-1)B_o \tag{4.2}$$

where P_{ch} is the channel power, F_n is the EDFA noise figure, h is Planck's constant, f_c is the channel frequency, G is the EDFA gain, and B_o is the optical bandwidth of the channel.

The values of ASEPerAmp for each frequency are computed once at program initialization and stored in an array to avoid repeating the same calculation.

Figure 4.1: Active Coherent Segments [4]



4.2.2 Cross Phase Modulation

The first step in computing the end-to-end Cross Phase Modulation (XPM) is the division of the lightpath into *active coherent segments*. An active coherent segment is a portion of the lightpath which is active using the same connection. For example, in Figure 4.1, Link 1 forms one active coherent segment, while Links 2 and 3 together form a second active coherent segment.

Note that the active coherent segment is a function of the center wavelength (λ_c) and an active wavelength inside the nonlinear sliding window (λ_1). The nonlinear sliding window is necessary to bound the complexity of the nonlinear computations for optical networks with a high degree of WDM. In practice, this has little impact upon the results if the window size is large enough since the nearest channels are the biggest contributors of noise.

Once the end-to-end path is divided into active coherent segments for each of the wavelengths inside of the window, the FWM can be calculated using Equation 4.3.

$$\sigma^2_{XPM} = \sum_{f_i \in W(f_c)} \sum_{\alpha}^{ACS} \text{XPMterm}(spans(\alpha), f_c, f_i) \tag{4.3}$$

In Equation 4.3, f_i is a channel inside the nonlinear sliding window of f_c that when active will cause XPM on f_c. α is an active coherent segment which has a length of $spans(\alpha)$.

The XPMterm calculation is taken from [4]. The noise due to XPM can be calculated by Equation 4.4

$$\sigma^2_{XPMterm} = \frac{1}{2\pi} \sum_{j \in pumps} \int_{-\infty}^{\infty} |H_i(w)|^2 PSD_i(w) dw \tag{4.4}$$

where $PSD_i(w)$ is the power spectrum density and $H_i(w)$ is the XPM transfer function of the channel i. The details of both functions can be found in [4].

The XPM calculations require integration and are computationally expensive. For this reason, the XPM calculations are computed once at program initialization and stored in a matrix. The XPMterm function is implemented in Matlab, which is compiled into a dynamic library and called from RAPTOR.

4.2.3 Four Wave Mixing

The first step in computing Four Wave Mixing (FWM) is to precompute the combinations of wavelengths that will cause FWM. This is done once as the program is initialized and the results are stored in a list to avoid repeating the same calculation.

To calculate the FWM for a given path, the list of combinations is checked. If all of the wavelengths in a particular combination are active, the path is divided into active coherent segments. FWMterm() is then called for each active coherent segment.

$$\sigma^2_{FWM} = \sum_{f_i, f_j, f_k \in W(f_c)}^{FWM} \sum_{\alpha}^{ACS} \text{FWMterm}(spans(\alpha), f_c, f_i, f_j, f_k) \tag{4.5}$$

In Equation 4.5, f_i, f_j, and f_k represent a combination of channels inside the nonlinear sliding window of f_c, that when active, will cause FWM on f_c. α is an active coherent segment which has a length of $spans(\alpha)$.

The FWMterm description is taken from [4]. The power of the FWM noise generated is

$$\sigma^2_{FWMterm} = |\gamma\frac{d}{3}A_m(0)A_n(0)A^*_p(0)L_{eff}|^2|\sum_{n=1}^{N} e^{j(k-1)(\phi_m+\phi_n-\phi_p-\phi_c)}|^2 \tag{4.6}$$

where d is the degeneracy factor and A_m, A_n, and A_p are the complex amplitudes of f_i, f_j, and f_k respectively. L_{eff} is calculated using Equation 4.7.

$$L_{eff} = \frac{e^{(j\Delta K_c-\alpha)^z}-1}{j\Delta K_c-\alpha} \tag{4.7}$$

ΔK_c is computed using Equation 4.8,

$$\Delta K_c = 2\pi\frac{\lambda_c^2}{c}(f_m-f_c)(f_n-f_c)[D-\frac{\lambda_c^2}{c}(\frac{f_m+f_n}{2}-f_c)S] \tag{4.8}$$

where D and S are the dispersion parameter and dispersion slope of the fiber.

In Equation 4.6, $|\gamma\frac{d}{3}A_m(0)A_n(0)A^*_p(0)L_{eff}|^2$ is the power of the FWM noise generated by one span and $|\sum_{n=1}^{N} e^{j(k-1)(\phi_m+\phi_n-\phi_p-\phi_c)}|^2$, the $link\ factor$, manifests the accumulation of noise along the entire link.

The link factor can be rewritten as $|\frac{1-e^{jN\Delta\phi}}{1-ej\Delta\phi}|^2$. The $\Delta\phi$ term is computed using Equation 4.9.

$$\Delta\phi = 2\pi\frac{\lambda_c^2}{c}(f_m-f_c)(f_n-f_c)[-\frac{\lambda_c^2}{c}(\frac{f_m+f_n}{2}-f_c)(S_{NZDSF}L_{NZDSF}+S_{DCF}L_{DCF})] \tag{4.9}$$

The FWM calculations represent a significant portion of RAPTOR's running time (between 5% and 15%). The combinatorial explosion of FWM possibilities makes it difficult to store the results as we did for XPM. Thus, each time the Q-factor is recalculated, the FWM noise is also recalculated.

4.2.4 Q-Factor

One simplifying assumption is that the impairments are independent. In other words, we assume that:

$$\sigma^2_{total} = \sigma^2_{ASE} + \sigma^2_{FWM} + \sigma^2_{XPM} \tag{4.10}$$

RAPTOR uses the total noise to calculate the Q-factor using Equation 4.11.

$$Q = 10 \log_{10} \frac{P}{\sigma_{total}} \tag{4.11}$$

RAPTOR uses the Q-factor to measure quality thresholds and connection quality.

4.3 Simulation Architecture

RAPTOR (Routing Assignment Program for Transparent Optical Routes) is a custom-built, C++ discrete-event optical network simulator. There are approximately 11,000 lines of source code. RAPTOR works on both the Windows and Linux platforms.

The heart of RAPTOR consists of a priority queue of events. The priority queue ensures that events are handled in the correct order. When a workstation is activated, it generates *Connection Requests*. Connections are then setup using the *Create Connection Probes* and *Create Connection Confirmations*, while they are destroyed using the *Destroy Connection Probe*. RAPTOR's implementation of a priority queue is called the EventQueue.

RAPTOR currently simulates the following RWA algorithms: DP-RWA, ACO-RWA, $\mathcal{MM}$-ACO-RWA, IA-BF, IA-FF, LORA, PABR, SP, QA SP, AQoS, and QM. DP-RWA is presented in Chapter 5. ACO-RWA and $\mathcal{MM}$-ACO-RWA are discussed in Chapter 6. The user can choose to simulate any subset of the algorithms at one

time. The program is flexible so additional algorithms can easily be added in the future.

The distribution of the workstations, and thus the traffic, can be uniform, random with a uniform distribution, or user-specified. The distribution of each connection request's destination can be uniform, distance weighted, or inverse distance weighted.

The number of wavelengths per fiber is also configurable. RAPTOR is fast enough to allow simulations of up to 1000 wavelengths per fiber. The methodology for calculating noise due to impairments is described in the previous Section.

RAPTOR can be run as either a console application or with a graphical interface (GUI). The GUI has proven to be an effective tool in analyzing algorithms and looking for improvements.

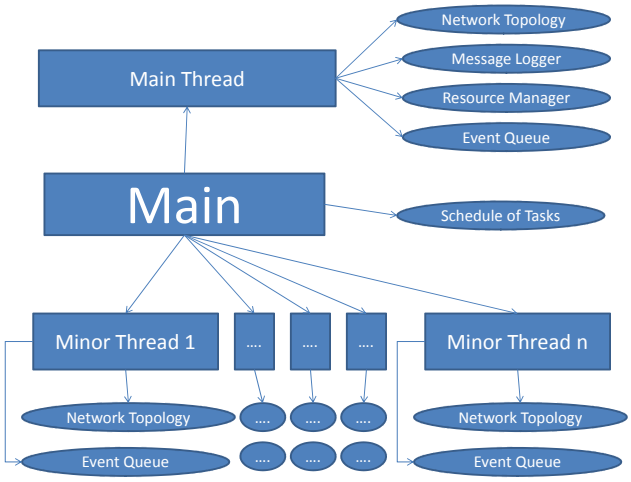
As with any C++ application, RAPTOR begins with a function call to Main. This function only serves two purposes: 1) generate a schedule of tasks and 2) manage the schedule of tasks. The second goal is accomplished by tasking an instance of the Thread class with a specific task, starting a chain of events. The Thread class, along with the other important classes of RAPTOR, are described in the following sections.

4.3.1 Thread Class

RAPTOR creates p threads, where p is an input parameter. The purpose of each Thread is to independently complete a *task* from the schedule. A task is simply a specified Routing Algorithm, Wavelength Assignment Algorithm, Probe Style (Single, Serial, or Parallel), and Offered Load. The Threads run simultaneously through the usage of the POSIX threads library.

A pointer to each Thread is stored in a dynamically sized array. The first Thread instance is the *Main Thread*. The Main Thread has access to the Network Topology through the Router, Edge, and Workstation classes, as well as the ResourceManager,

Figure 4.2: RAPTOR Parallel Architecture



EventQueue, and Message Logger classes. Each of these classes is discussed in detail in the following sections.

If $p > 1$, RAPTOR will create $p - 1$ *Minor Threads*. Each Minor Thread has its own Network Topology and EventQueue. However, it uses the ResourceManager and MessageLogger from the Main Thread to avoid unnecessary instances of those classes. The parallel architecture of RAPTOR is shown in Figure 4.2.

Table 4.1 shows the speedup factor with various numbers of threads. The number of threads was cut off at 8 as the simulation was run on an 8 core machine. Further multi-threading gains could likely be achieved on a machine with additional cores, but we could not find such a machine.

Once a Thread is initialized with a task, the Thread runs by removing an Event from the EventQueue and handling it. Each Event has its own function associated with it, which in turn, usually generates more Events. For example, the Connection-RequestEvent will likely create a CreateConnectionProbeEvent. The Thread runs until all of the Events have been processed.

Table 4.1: RAPTOR Parallel Performance

Threads	Total Run Time (sec)	Speedup
1	2290.7	1.000
2	1187.7	1.929
3	827.3	2.769
4	667.0	3.434
5	553.0	4.142
6	494.0	4.637
7	443.0	5.171
8	398.0	5.756

4.3.2 ResourceManager Class

ResourceManager has three main functions: 1) compute the signal quality, 2) compute a path from a source to a destination, and 3) run the appropriate Wavelength Assignment algorithm. The signal quality calculation has already been presented in detail, so this section will focus on the last two functions.

All of the Routing Algorithms presented thus far (SP, QA SP, LORA, PABR, IA, QM, AQoS) are implemented using a standard k-shortest path calculation. If k, the number of probes, is equal to one, Dijkstra'a Algorithm is used. For cases where $k > 1$, Yen's Algorithm is used. [28] The running time of Yen's Algorithm is $O(kn(m + n \log n))$ where m is the number of edges, n is the number of routers, and k is the number of paths. This is a factor of kn times slower than Dijkstra's running time, $O(m + n \log n)$.

DP-RWA, ACO-RWA, and $\mathcal{MM}$-ACO-RWA, which will be presented later, do not use a shortest path algorithm to calculate the route.

PABR is unique as it places an additional constraint that keeps the number of EDFAs in a path below a threshold. Such paths can be found by incrementally

increasing k until enough satisfactory paths have been found. Another approach would be to use Dynamic Programming.

The Wavelength Assignment algorithms are also implemented in ResourceManager. Each algorithm takes a list of candidate wavelengths and selects one of them. The implementation of each algorithm is straight-forward, so the code description has been omitted.

4.3.3 EventQueue Class

The EventQueue is used to store Events. Each Thread has its own EventQueue. The EventQueue contains an instance of the C++ Standard Template Library (STL) priority_queue. The priority_queue stores its elements in a partial ordering, such that the first element is always the highest (a max priority_queue) or lowest (a min priority_queue).

Each Event has a time property that specifies exactly when the Event should occur. The EventQueue is a min priority_queue, so the first element always has the lowest time. This requires the $<$ operator for the Event structure to be overwritten. This ordering allows events to be processed in the correct order.

The STL implementation of priority_queue uses a balanced binary search tree. This allows both insertions and deletions to run in just $O(\log n)$ time.

4.3.4 MessageLogger Class

The MessageLogger is used to write Events to a log file. One log file is sufficient for all of the Threads, so there is only one instance of the MessageLogger. The system time, and thread index are written to the log file along with the Event information.

There are two options for the log file: 1) standard and 2) detailed. The standard log file only writes the final statistics to the log file (blocking probability, average

quality, etc). This is sufficient in most cases. The detailed log file will record the setup and destruction of connections, along with path and quality information. This is useful for debugging purposes.

4.3.5 Router Class

The Router class stores pointers to its adjacent Edges. While the term Router is used for convenience, a more accurate term is an Optical Switch. Router provides an interface to access the edges via their destination or their index. This interface is used to periodically update the Edge usage (for PABR and LORA), QM Degradation (for AQoS and QM), and the Q-factor stats (if the Q-factor stats flag is set).

The Router class also processes the probe messages that are used to setup and destroy connections. This involves forwarding probe messages toward their destination Router and processing the probe message when it arrives at the destination.

4.3.6 Edge Class

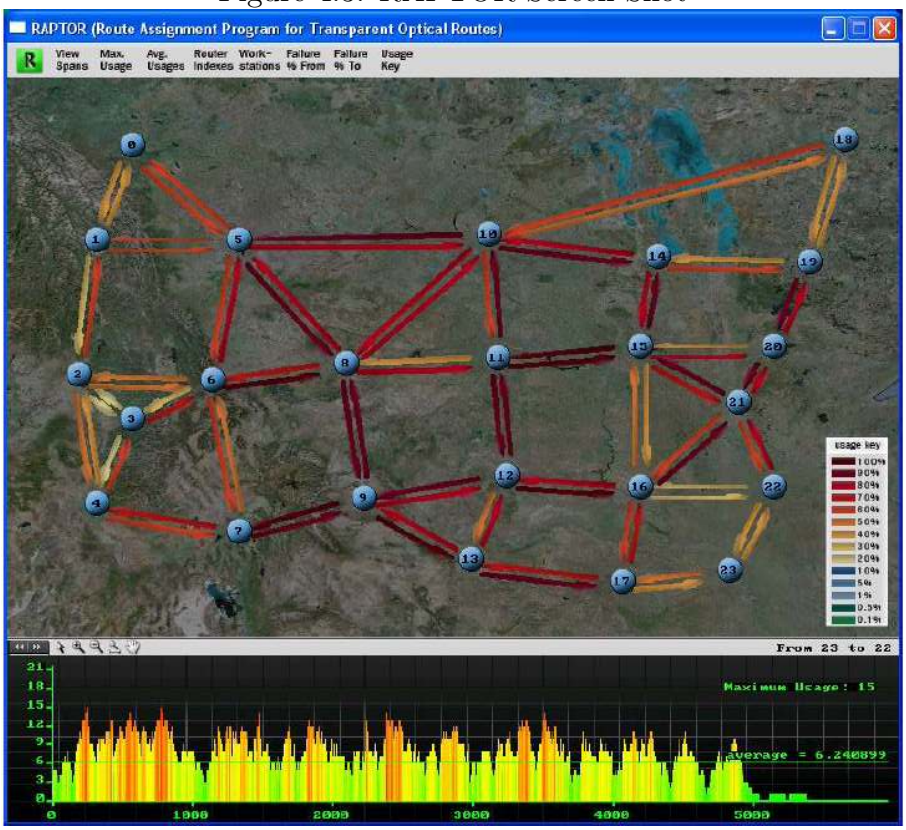
The Edge class is used mainly as a container to store information about the Edges. This includes the information about the physical Edge, such as the source router, destination router, and the number of spans. This also includes an array with the current status information (EDGE_FREE or EDGE_USED).

The Edge class also contains the source code used to update the Edge usage (for PABR and LORA), QM Degradation (for AQoS and QM), and the Q-factor stats (if the Q-factor stats flag is set).

4.3.7 Workstation Class

The Workstation class is used solely to generate traffic on the network, when the Workstation is active. Each Workstation is attached to a parent Router.

Figure 4.3: RAPTOR Screen Shot



4.3.8 GUI

RAPTOR has an optional graphical user interface (GUI). The GUI was developed primarily by Andrew Albers, an undergraduate student in the Computer Science Department at Montana State University.

The GUI requires the usage of the Allegro library. Precompiler directives (#ifdef RUN_GUI and #endif) were added around the GUI specific portions. This allows the user to choose between running the GUI version and the Console version. The directives also remove the GUI overhead from the Console version.

The GUI version provides a couple additional features. First, as the simulation is running progress bars appear for each Thread. This can be convenient for large

simulations which take days to run. The GUI version also tracks some additional usage information, that is stored in a text file. This file can then be used to review the simulation. An example of this is given in Figure 4.3.

The optical links are color coded based upon their utilization, from green (low utilization) to red (high utilization). In Figure 4.3, the interior links each have high utilization rates, serving as a bottleneck preventing new connections from being added. The visualization of hot spots in the network improve our understanding of RWA algorithms.

The GUI also has a temporal aspect to it. There is a graph at the bottom the GUI that is used to show the link utilization of a particular edge over time. From the example given, we can conclude that the traffic increased from zero very suddenly, remained roughly constant, and then tailed off at the end of the simulation.

There is also a topology builder, that can be used to view, build, and modify network topologies.

One drawback is that the GUI version takes roughly 35% longer to run than the console version.

4.4 Simulation Inputs

The user can configure RAPTOR through various input parameters. The important parameters are discussed in this section.

4.4.1 Algorithm Parameters

The Algorithms parameters file specifies which algorithms to run. Each line in the file should have the following structure: "RA=a,WA=b,PS=c,QA=d,RUN=e".

a specifies the Routing Algorithm (SP, LORA, PABR, QM, AQoS, IA, DP-RWA, ACO-RWA, $\mathcal{MM}$-ACO-RWA). b specifies the Wavelength Algorithm (FF, FFwO, BF, RP, Q-FF, Q-FFwO, Q-RP, LQ, MQ). In general, any combination of RA and WA can be specified. There are two exceptions, in that IA must have either FF or BF as a WA and BF can only be used with IA.

c specifies the probe style (SINGLE, SERIAL, PARALLEL). d specifies whether the algorithm is quality aware (0 for false or 1 for true). A quality aware algorithm will estimate connection quality at setup and block connections with insufficient quality. This is meaningful only for non quality aware wavelength algorithms (FF, FFwO, BF, RP) as the quality aware wavelength algorithms require testing the signal quality (Q-FF, Q-FFwO, Q-RP, LQ, MQ).

e specifies whether the algorithm should be scheduled or ignored. (0 for ignored, 1 for scheduled). It may be useful to turn certain lines of the input file off for debugging purposes. While this can be accomplished by deleting the line entirely, often keeping the line and just ignoring it is more convenient.

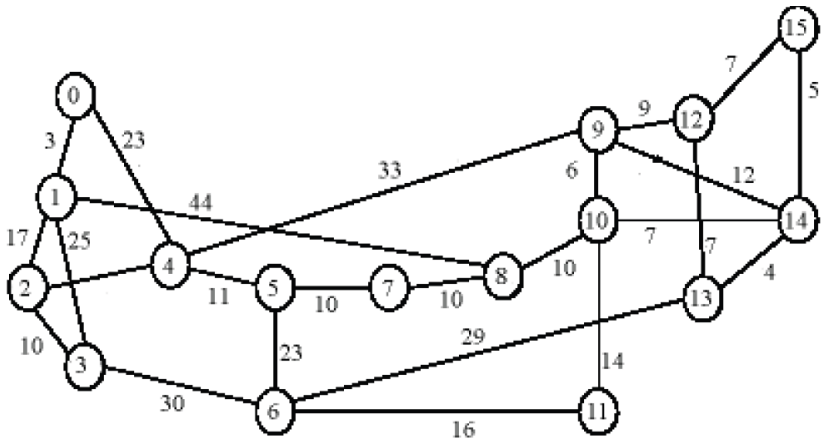
Unless otherwise specified, the Quality First Fit (Q-FF) wavelength algorithm is used for the physically aware algorithms while First Fit (FF) is used for the non physically aware algorithms. The maximum number of probes per connection request is set to 4.

4.4.2 Topology Parameters

The Topology parameters file specifies the network topology to simulate. The file is divided into two segments, a Router section and an Edge section.

The Router section must be placed in the file first. While the term Router is used for convenience, a more accurate description is an Optical Switch. Each Router is specified on a single line using the syntax "Router=x,y", where x and y are the

Figure 4.4: NSF Network Topology



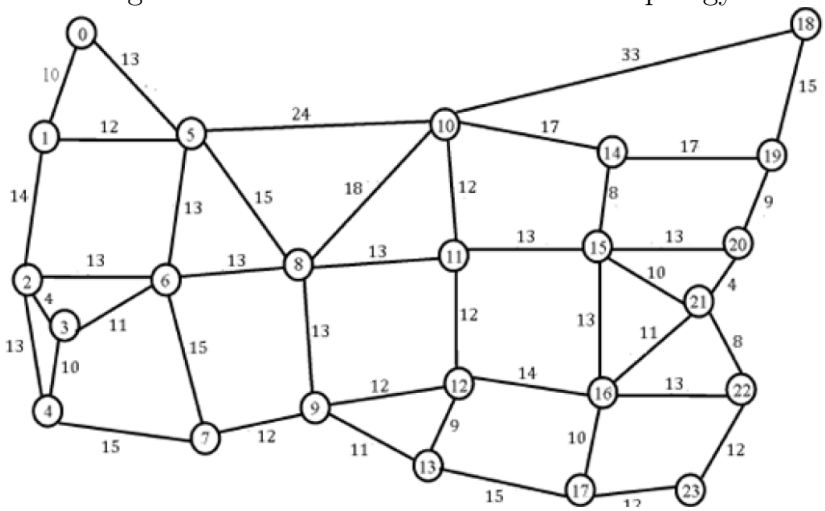
coordinates of the Router. The coordinates are used only for the GUI's display of the routers.

The Edge section is placed after the Router section. Each Edge is specified on a single line using the syntax "Edge=$r1$,$r2$,s", where $r1$ and $r2$ are Router indexes and s is the number of spans (EDFAs). All edges are assumed to be bidirectional, so both Edges $r1 \rightarrow r2$ and $r2 \rightarrow r1$ are created.

There are many common network topologies implemented in RAPTOR. The NSF network topology is given in Figure 4.4. The NSF network is a 16 node, 25 edge network. One drawback of the NSF network is the lack of connectivity between the edges. This reduces the viability of alternate paths, making it difficult to route requests away from congested areas in the network.

The UC Davis Mesh network topology was first proposed in [32]. The UC Davis Mesh network topology is given in Figure 4.5. For simplicity, the network will be named the Mesh network. This is a 24 node, 43 edge network. The Mesh network has a higher degree of connectivity.

Figure 4.5: UC Davis Mesh Network Topology



Other network topologies include a standard x-x Mesh. RAPTOR has been tested with 6x6, 8x8, and 10x10 Mesh files.

4.4.3 Quality Parameters

All of the Quality parameters are included in Table 4.2. The arrival_interval and duration specify the traffic load. Since arrival_interval = duration, each workstation adds 1 Erlang of offered load on average. To adjust the traffic load, the user can either add workstations or adjust the arrival_interval and duration parameters.

Note that both the nonlinear sliding window size W and the number of wavelengths (λ) parameters specify just half of the size of the window and number of wavelengths. The size of the nonlinear window is actually $2W$, or W channels on each side of the channel under consideration. The number of wavelengths is actually $2\lambda + 1$, in other words, λ channels on each side of the center frequency.

The Q-factor_factor f is used to calculate the threshold of acceptable quality. Upon initialization, RAPTOR calculates the maximum-minimum distance between routers in spans. This distance is used to calculate the ASE noise of that distance,

Table 4.2: Quality Parameters

Quality Parameter	Value	Description
arrival_interval	250 seconds	Mean inter arrival time for each connection
duration	250 seconds	Mean duration time for each connection
nonlinear_halfwin	20 channels	Half of the nonlinear sliding window size
halfwavelenth	10 - 640	Half of the total number of wavelengths
fc	193.1 THz	Frequency of the center channel
f_step	50 GHz	Frequency spacing between channels
channel_power	1 mW	Power per channel
NZDSF L	80 km	Length of each NZDSF fiber span
NZDSF alphaDB	0.25 dB/km	Attenuation of the NZDSF
NZDSF D	4 ps/(km*nm)	Dispersion coefficient of the NZDSF
NZDSF S	0.08 ps/km*nm^2)	Dispersion slope of the NZDSF
NZDSF γ	2 1/(km*W)	Nonlinear coefficient of NZDSF
Q-factor_factor	0.95	Used to calculate the Q-factor threshold
ASEPerEDFA	8e-10 W	ASE noise per EDFA
usage_update	0.0625 - 4 seconds	Interval between usage updates
gui_update	6 seconds	Interval between GUI updates
β	1.1 - 1.2	Parameter for PABR
refractive_index	1.5	Refractive Index of the optical links
q_factor_stats	0 or 1	Track connection quality (1=yes, 0=no)
detailed_log	0 or 1	Keep a detailed log (1=yes, 0=no)

and then the Q-factor Q, assuming no nonlinear impairments. The Q-factor threshold is set to Qf. Values of f below 1 increase the viability of alternate paths by lowering the Q-factor threshold.

4.4.4 Other Parameters

Some parameters are passed to RAPTOR via the command line. These parameters are not included in an input file as they are frequently changed. Leaving them as command line parameters allows scripts to run RAPTOR with a minimal number of input files.

The first two command line arguments are the network topology and wavelengths per fiber. The following network topologies are supported: NSF, Mesh, Mesh6x6, Mesh8x8, and Mesh10x10. The following numbers are valid for the number of wave-

lengths per fiber: 21, 41, 81, 161, 321, 641, 1281. The user can create a custom network topology if desired.

The next command line parameter is the random seed. This number is used to seed the random number generator for the destination router of connections, the duration of connections, and the arrival interval of connections. This allows a user to run several random seeds, usually for the purpose of averaging the results.

The fourth command line parameter is the number of iterations. In most cases each RWA algorithm is run several times with varying levels of traffic demands to produce a graph. The number of intervals is typically set to 10, to allow for 10 data points per algorithm.

The final command line parameter is the maximum number of probes. This is used only for the multi-probing algorithms which will attempt up to p probes in either a parallel or serial fashion. Typical values for p are between 2 and 8.

4.5 Simulation Outputs

RAPTOR outputs many metrics which can be used to evaluate the performance of RWA algorithms. *Overall blocking* probability is the percentage of connections that are denied. There are 3 types of blocking: collision blocking, resource blocking, and quality blocking. Overall blocking is the sum of the 3 types.

Collision blocking occurs in a distributed algorithm where two different connection requests attempt to utilize the same resource. *Resource blocking* occurs when an RWA algorithm is unable to find a free route and wavelength. *Quality blocking* occurs in a physically aware algorithm when a path and wavelength are free, but they have insufficient signal quality.

RAPTOR also outputs the average number of probes per connection request and the average request delay time. In a centralized algorithm, the probes per request will always be 1. The request delay time is the time elapsed between the message sent to the network controller and the response.

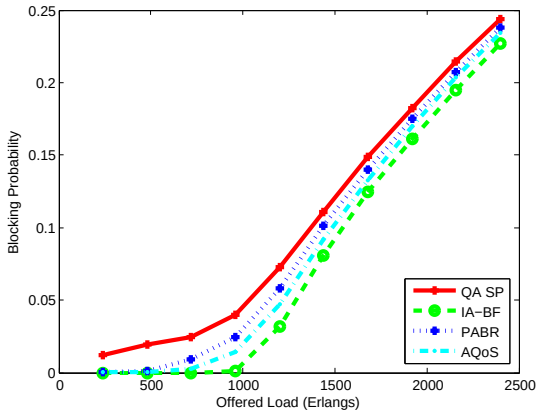
In distributed algorithms, probe messages are required to ensure the path selected is actually available. Single probing algorithms will attempt only one path, while multi-probing algorithms will attempt multiple paths. The paths can be attempted in a serial or parallel style. The request delay time is the time elapsed between sending the first probe message and receiving the final probe response.

Many statistics are tracked for connections that are successfully setup. RAPTOR outputs the average path length, in terms of both hops (optical switches) and spans (EDFAs). The average amount of ASE, FWM, and XPM per connection at the time of setup is also tracked. These three metrics are unique to RAPTOR.

The user can opt to allow connection quality stats as well. This can be computationally expensive, especially as the number of wavelengths per fiber increases. If this option is enabled, the quality of a connection is tracked over its duration. Each time a connection c is added or dropped on the network, the quality of all connections using at least one of the edges in c is recomputed.

This allows RAPTOR to output several important performance metrics. The *Average Q Factor* is defined as the average of the average Q factor over time for each of the successful connections. The *Percent Time Below Threshold* is the average time a connection spends below the quality threshold. While physically aware algorithms test the quality threshold as connections are setup, as additional connections are added the signal quality can drop below the threshold. Both the Average Q Factor and Percent Time Below Threshold are unique to RAPTOR.

Figure 4.6: Overall Blocking - RAPTOR



RAPTOR also computes the time spent computing the path and selecting a wavelength. For applications where network delay is important or connection requests have a short duration, this is an important metric.

4.6 Simulation Results

To demonstrate the capabilities of RAPTOR, we consider the Mesh network. All edges are assumed to be bi-directional and have a capacity of 81 wavelengths. The QA SP, PABR, AQoS, and IA-BF algorithms are considered.

All of the results presented here are the average of 5 random seeds: 12210, 24239, 38979, 50877, and 54075.

The Overall Blocking probability is shown in Figure 4.6. SP has the worst performance in terms of blocking probability, as SP is limited to using the same fixed route for all connection requests. IA-BF outperforms PABR and AQoS, exploiting its knowledge of the complete state of the network. PABR, despite being a completed distributed algorithm, outperforms AQoS, which requires centralization.

Figure 4.7: Quality Blocking - RAPTOR

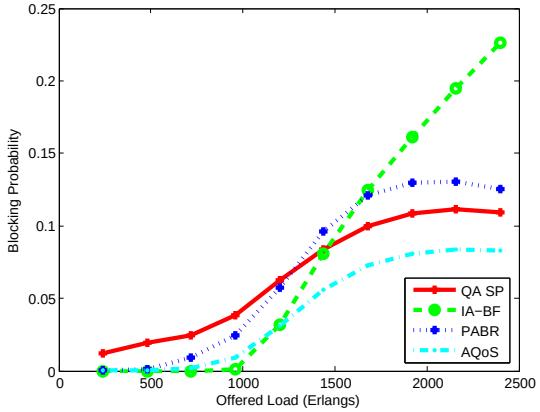


Figure 4.8: Resource Blocking - RAPTOR

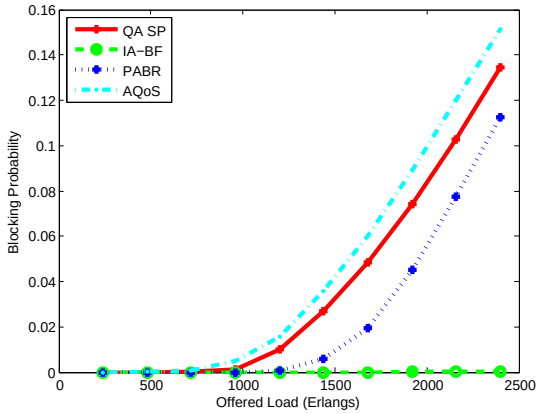
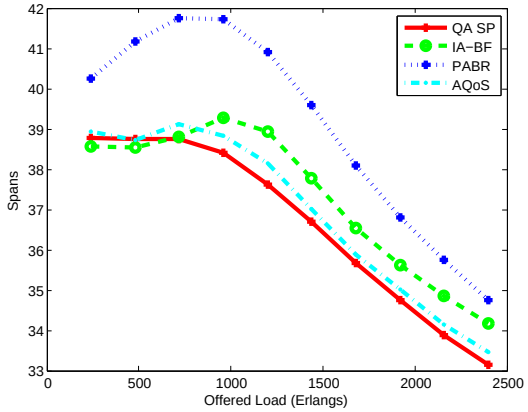


Figure 4.9: Connection Length - RAPTOR



The Quality Blocking probability is shown in Figure 4.7 and the Resource Blocking probability is shown in Figure 4.8. IA-BF is able to use its centralized knowledge to find a path and wavelength nearly all of the time, however, this leads to the dominance of Quality blocking. For PABR and AQoS, Quality blocking and Resource blocking are comparable. QA SP has a high amount of resource blocking due to its fixed path constraint.

The average accepted connection length, shown in Figure 4.9, initially increases for all of the algorithms and then decreases sharply. As the traffic load increases, for a limited range of offered load, all of the RWA algorithms are able to choose a slightly lower alternate path. However, as the load increases further the longer paths are blocked due to increasing ASE, FWM, and XPM. Connections requiring a shorter path have a higher probability of being accepted, and thus the average accepted connection length decreases.

The average connection quality, shown in Figure 4.10 has the inverse relationship. As the average path length increases, the average connection quality decreases. This is due largely to the dominance of ASE noise.

Figure 4.10: Connection Quality - RAPTOR

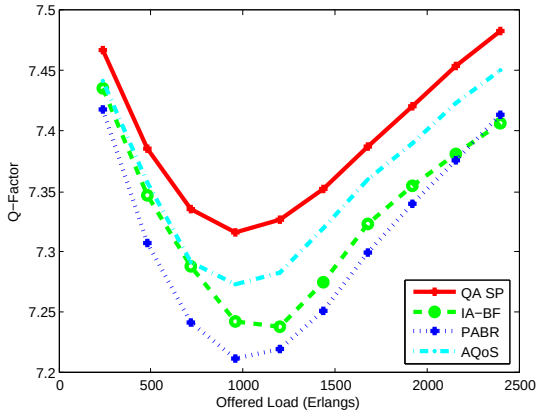


Figure 4.11: Connection Time Below Threshold - RAPTOR

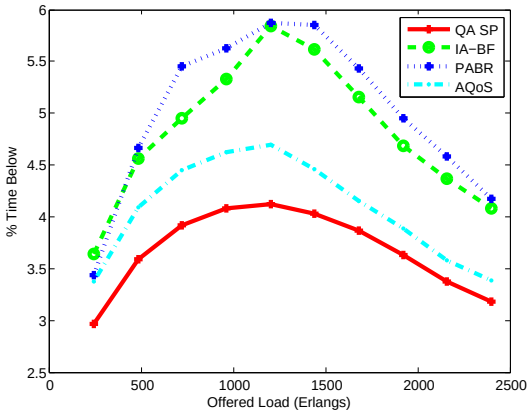
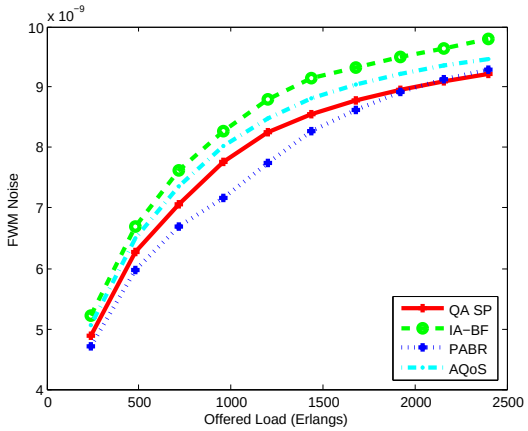


Figure 4.12: FWM Noise - RAPTOR



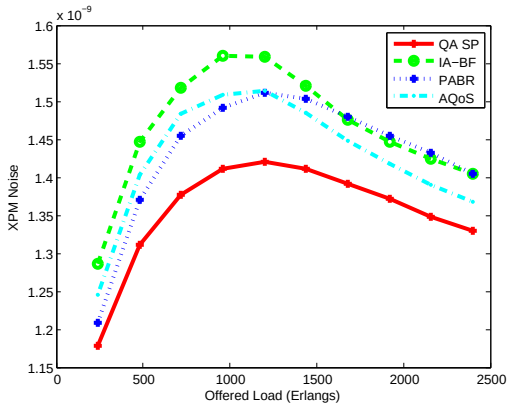
Physically aware algorithms (QA SP, IA-BF, PABR, and AQoS) all block connections with insufficient signal quality at the time that the connection request is initiated. A connection request with adequate quality will be affected by subsequent connections. Existing connection quality can decrease below the threshold when additional connections are added. The average percent time below threshold is shown in Figure 4.11.

QA SP has the lowest average time below threshold, a result of shorter connection lengths and a higher blocking probability. PABR has the longest path length and the largest time below the threshold. Longer paths have more noise due to ASE, FWM, and XPM, increasing the likelihood of the quality falling below the threshold.

Figures 4.12 and 4.13 show the amount of FWM and XPM for accepted connections. The FWM contribution is higher than the XPM on average, however, there are certainly individual cases where XPM dominates FWM. The dominant noise contributor is ASE.

RAPTOR has been shown to be a powerful tool capable of modeling large scale transparent optical networks. RAPTOR models three of the dominant physical impairments (ASE, FWM, and XPM). RAPTOR is fast and flexible. It includes new

Figure 4.13: XPM Noise - RAPTOR



performance metrics. While there are other network simulation tools available, only RAPTOR meets our needs.

Without this software package, it would be impossible to perform a detailed performance evaluation of existing RWA algorithms. The ability to develop and compare new and improved RWA algorithms would also be significantly more difficult. RAPTOR is an integral part of the research presented in this discussion.

CHAPTER 5

DYNAMIC PROGRAMMING - ROUTING AND WAVELENGTH ASSIGNMENT

Current RWA algorithms are unable to meet the needs of future transparent optical networks, with dynamic traffic demands and end-to-end optical transmission. Current algorithms yield a high level of blocking, do not scale well to large networks, and do not consider nonlinear impairments. We present a new algorithm based upon dynamic programming that addresses these shortcomings. We show through theory and simulation that our algorithm scales well to large networks. We show through simulation that our algorithm yields the lowest blocking probability in a variety of simulations. It is flexible and can adapt to situations where either linear or nonlinear impairments dominate.

Our findings have been submitted for publication. A conference version of the paper [40] is waiting to be accepted.

5.1 Dynamic Programming RWA

Our new RWA algorithm is named Dynamic Programming - RWA (DP-RWA or DP). The technique is based upon the dynamic programming paradigm. The light path is recursively computed in a bottom-up fashion. Note the optimal sub-structure property does not always hold for RWA, so this is not a true dynamic programming approach.

Each node n on the network stores a list of candidate paths, P_n. Each list contains a maximum of k paths. Each item on the list must store the path, path weight, optimal wavelength, and a list of available wavelengths.

INPUT: Connection request from s to t, $G = (V, E)$

Repeat until no candidate path changes

For all $e \in E$

DP-Relax(e)

Algorithm 5.8: Dynamic Programing - RWA (DP-RWA)

INPUT: Edge $e = (u, v)$

For all $p \in P_u$

1. $\rho = p \cup e$
2. Generate a list of wavelengths that are available on ρ. Let Λ be the set of available wavelengths.
3. Compute the path weight of all wavelengths in Λ and let λ be the wavelength with optimal path weight.
4. Remove all wavelengths from Λ with insufficient quality.
5. **If** Weight(ρ) > Weight($P_{v,k}$) for any k

   Insert ρ into P_v.

Algorithm 5.9: DP-Relax

DP-RWA operates by repeatedly calling DP-relax. Each iteration of calling DP-relax for all of the edges is called a phase. DP-RWA terminates after none of the candidate paths change in a phase.

DP-relax iterates over the candidate lightpaths in the head of the edge (P_u). Each path is extended by an edge and compared against the worst path in the tail of the edge (P_v). If the new path is better, it is inserted into the candidate light path.

The weight of path p is calculated using Equation 5.1. α is a parameter in the range of $0 \leq \alpha \leq 1$ used to bias the weight toward the best quality ($\alpha = 0$) or the shortest path with acceptable quality ($\alpha = 1$). Q_p is the quality of path p and l_p is the length in EDFAs of path p.

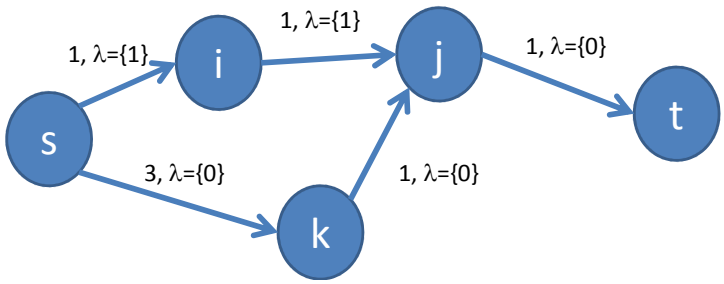
$$W_p = (1 - \alpha)\frac{Q_p}{\bar{Q_p}} + \alpha\frac{l_p}{\bar{l_p}} \tag{5.1}$$

$\bar{Q_p}$ and $\bar{l_p}$ represent estimates of the path quality and path length. $\bar{l_p}$ is typically the average of the shortest path and the diameter of the network. $\bar{Q_p}$ is usually the quality factor after $\bar{l_p}$ EDFAs, assuming zero nonlinear impairments.

A pruning technique can be used to limit the candidate path search space. There is a fixed level of ASE noise that any path will always add to a connection based upon the number of EDFAs. When considering any path, we can ignore any path where the noise on the path plus the ASE noise on the optimal shortest path to the actual destination is unacceptable.

A chief advantage of DP-RWA is the combination of route selection and wavelength selection in a single step. Each candidate path stores the wavelength with the greatest path weight. Thus, when the path computation is complete, the k shortest paths with the optimal wavelength are returned. As the degree of wavelength division multiplexing increases, this advantage becomes even more significant.

Figure 5.1: Example of Dijkstra Failure



Another advantage of DP-RWA is the incorporation of nonlinear impairments into path selection. Paths with an unacceptable level of noise are pruned from the search space, freeing the algorithm to find alternate routes. As nonlinear impairments become more significant in the future with higher degrees of WDM, this advantage will become even more significant.

Our approach is similar to the Bellman-Ford Algorithm, which relaxes the edges of the graph. We can not use an approach based upon Dijkstra's Algorithm, which relaxes the nodes of the graph. Dijkstra's Algorithm visits each node exactly one time. If DP-RWA used this approach, many paths would never be discovered, even with infinite path queues. An example is given in Figure 5.1.

Consider a connection request from s to t. Dijkstra's Algorithm would visit node i and then node j. At this point, node j would contain the candidate path (s, i, j). Node k would be visited next and contain the candidate path (s, k). Node t is visited last. Due to the wavelength continuity constraint, the path (s, i, j, t) is not valid. Thus, a Dijkstra based search would not discover a path.

A path does exist (s, k, j, t) with a free wavelength. This path is not discovered by Dijkstra because node k is visited after node j. A Bellman-Ford approach would discover this path after a maximum of 3 phases.

5.1.1 Computational Complexity

There are two ways to compare the computational requirements of DP-RWA. The theoretical analysis is presented here and the experimental results are presented later.

Lemma 1. *After the pth phase, all paths less than or equal to p in length will not change. In future phases, only paths greater than p will be created.*

Proof. The proof is by induction on the number of phases completed. Upon completion of phase 1, all edges of the form (s, u) will have been relaxed to create paths of length 1 from s. If an edge (s, u) is relaxed again in a subsequent phase, the path (s, u) will already belong to u's queue (or have been displaced by a path of higher quality). In either case, (s, u) will not create a new path. By the inductive hypothesis, after phase $p - 1$, all paths of length $p - 1$ will have been created (and will not be recreated in a subsequent phase). So, by the end of phase p all possible paths of length p will have been created. It follows that after phase p, any new path created must have a length greater than p. □

Theorem 2. *DP-RWA runs in $O(VEQ)$ time, where V is the number of verticies, E is the number of edges, and Q is the maximum queue length.*

Proof. At phase p, no paths of lengths less than p will be created (see Lemma 1). We are interested only in simple paths, so the maximum number of phases is V - 1. Each phase calls DP-relax E times, and each call to DP-relax takes $O(Q)$ time. Thus, the run time is $O(VEQ)$. □

In practice, DP-RWA offers good performance with a queue size of one. The optimal values of Q vary based on the network. In our simulations, the optimal value for the NSF network is 4 and the UC Davis Mesh network is 7.

It should also be noted that DP-RWA is called just once per connection request, independent of the number of wavelengths. If Λ is the number of wavelengths on the network, IA-BF takes Λ calls to Dijkstra's algorithm, or $O(\Lambda(V^2 + E))$ [41]. Thus, DP-RWA scales much better than IA-BF to large networks.

PABR and AQoS both require a k shortest path library, as using a single path leads to higher connection blocking. Yen's Algorithm runs in $O(KN(E + V \log V)$ [28], where K is the number of paths. Both algorithms also require time to update the edge weights when a connection is added or dropped.

5.2 Simulation Setup

RAPTOR (Routing Assignment Program for Transparent Optical Routes) was used to compare DP-RWA with existing RWA algorithms. RAPTOR is a custom built C++ discrete event simulator. RAPTOR is capable of modeling linear (ASE) and nonlinear impairments (FWM, XPM) in transparent optical networks.

To determine the performance of DP-RWA, three networks were used. The NSF network is a small 15 node network with limited connectivity. The UC Davis Mesh is a medium sized 24 node network with moderate connectivity. A standard 8x8 mesh network is a large scale network with a high degree of connectivity.

Workstations were attached to each of the nodes. The number of active workstations, and thus network load, was then varied. Each active workstation generates one Erlang of traffic.

Three scenarios were considered to determine the performance of DP-RWA across a variety of network conditions. In the first scenario, ASE is the dominant impairment. In the next scenario, the linear and nonlinear impairments are balanced and neither dominate. In the final scenario, the nonlinear effects of XPM and FWM are the

Table 5.1: Varied Simulation Parameters

Parameter	ASE Dominant	Balanced	Nonlinear Dominant
Channel Power (mW)	1.0	1.5	2.0
NZDSF γ ($\frac{1}{km*W}$)	2.6	2.75	2.9
EDFA Noise Figure (dB)	3.5	3.0	2.5
Minimum Q-factor	5.489	5.526	5.059

Table 5.2: Constant Simulation Parameters

Parameter	Value
Arrival Interval (Mean)	250 seconds
Connection Duration (Mean)	250 seconds
Nonlinear Half Window	20 channels
Frequency Center	193.1 THz
Frequency Spacing	50 GHz
EDFA Spacing	80 km
NZDSF α	0.25 dB/km
NZDSF D	4 ps/(km*nm)
NZDSF S	0.08 ps/(km*nm^2)

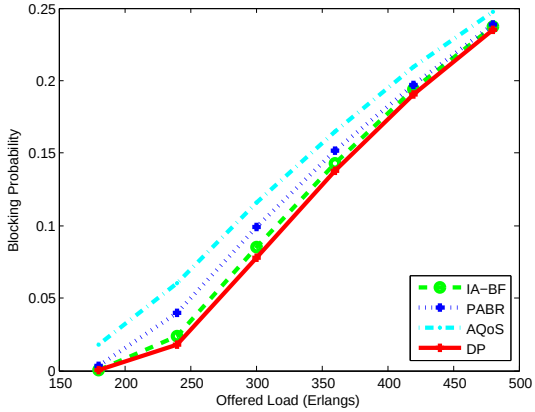
dominant impairments. The parameters listed in Table 5.1 each denote one of the scenarios.

In all cases, the α value for DP-RWA was set to 1.0. There is a trade-off, lower values of alpha lead to higher signal quality but also a higher blocking probability.

The parameters listed in Table 5.2 are held constant for all of the scenarios.

The PABR, AQoS, DP-RWA, and IA-BF algorithms are compared. These algorithms were used as they represent the state of the art physically aware RWA algorithms. PABR, and AQoS use parallel multi-probing with a maximum of 4 probes per request. IA-BF and DP-RWA are centralized and thus do not use probes.

Figure 5.2: Overall Blocking - DP-RWA on Mesh Network



5.3 Results

This section presents the results for the DP-RWA algorithm. The UC Davis Mesh network results are presented first, followed by the NSF network. The 8x8 Mesh network results are presented last.

5.3.1 UC Davis Mesh Network

The results from the balanced scenario are presented in Figures 5.2 - 5.5. Figure 5.2 shows the overall blocking probability. DP-RWA blocks 3.7% fewer connections than IA-BF. DP-RWA outperforms PABR by 10.7% and AQoS by 23.8%.

Figure 5.3 displays the average time of path computation for a connection request. IA-BF has the highest time, growing linearly with the wavelengths. AQoS, PABR, and DP-RWA grow much more slowly. DP-RWA grows due to the increased numbers of wavelengths for which it has to track availability and quality. AQoS and PABR grow due to the increased time spent updating the edge costs. At 161 wavelengths, DP-RWA requires just 5.6% of the run time IA-BF. DP-RWA has the lowest run time.

Figure 5.3: Run Time - DP-RWA on Mesh Network

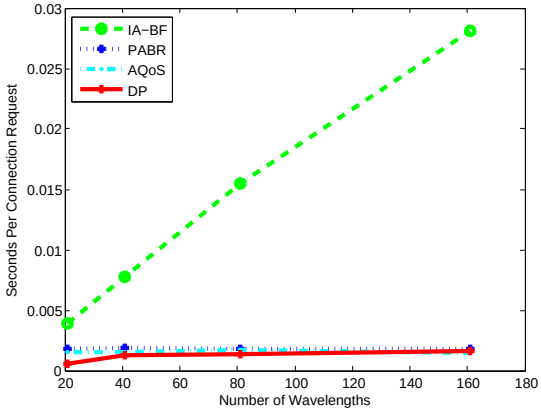
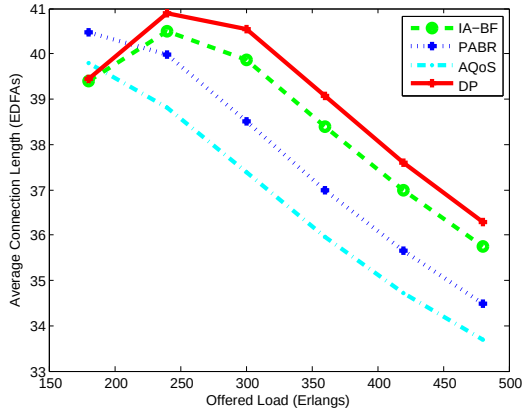


Figure 5.4: Connection Length - DP-RWA on Mesh Network



DP-RWA's search process prunes paths with an unacceptable signal quality, resulting in paths that are slightly longer than the other algorithms. Figure 5.4 shows the average connection length of accepted connections. The use of these longer paths enabled DP-RWA to achieve the best overall blocking probabilities.

A side effect of longer paths and more active connections is lower signal quality, displayed in Figure 5.5. IA-BF has an average Q-factor that is 0.3% greater than DP-RWA. PABR and AQoS outperform DP-RWA by 1.7% and 1.9% respectively. It should be noted that average connection quality is not a primary concern of RWA, the

Figure 5.5: Connection Quality (threshold ≈ 5.526) - DP-RWA on Mesh Network

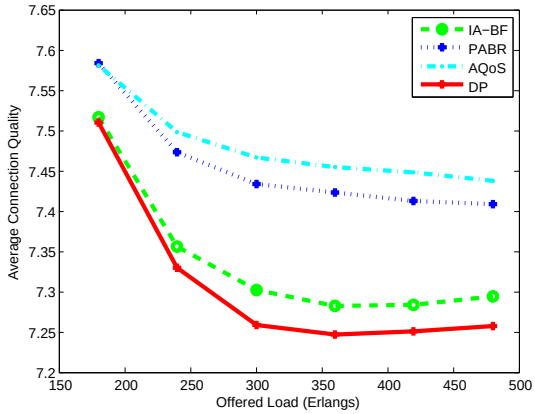
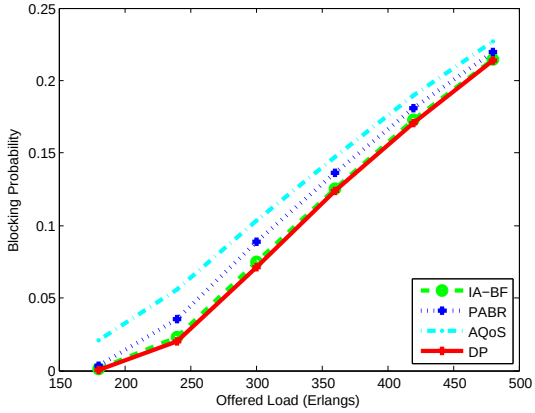


Figure 5.6: Overall Blocking - ASE Dominant Scenario on Mesh

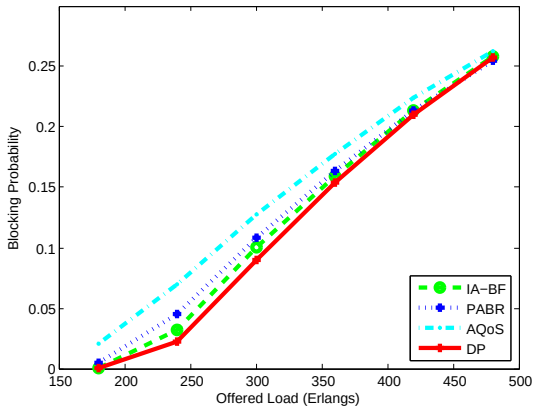


objective is typically to maximize the number of active connections with an acceptable level of quality. The thresholds for the minimum Q-factor are listed in Table 5.1.

The results from the ASE dominant simulation are presented in Figure 5.6. In this scenario, the effects of ASE are roughly four times greater than the sum of the FWM and XPM effects, on average. DP-RWA outperforms IA-BF, PABR, and AQoS by 2.0%, 2.4%, and 10.7%, respectively.

The results from the nonlinear dominant simulation are presented in Figure 5.7. In this scenario, the effects of FWM and XPM are roughly four times greater than

Figure 5.7: Overall Blocking - Nonlinear Dominant Scenario on Mesh



ASE noise, on average. DP-RWA outperforms IA-BF, PABR, and AQoS by 4.0%, 7.7%, and 20.3%, respectively.

In all 3 scenarios, DP-RWA has the lowest blocking probability. As the nonlinear effects become more significant, the gap between DP-RWA and IA-BF (and the other algorithms as well) grows larger. This is due to DP-RWA pruning paths where the noise (both linear and nonlinear) is too high. IA-BF does not consider the nonlinear effects in its path computation. As future networks will utilize an increasing number of wavelengths, DP-RWA likely will perform even better when compared to IA-BF.

5.3.2 NSF Network

Simulations also were run on the NSF network topology. The diameter of the NSF network is just 71 spans, smaller than the UC Davis Mesh network's diameter of 85 spans. For this reason, the Q factor thresholds are a little higher. The other parameters were left unchanged from Table 5.2.

In terms of blocking probability, DP-RWA outperformed IA-BF by 1.9%. The results are presented in Figure 5.8. DP-RWA outperformed AQoS by 7.3% and

Figure 5.8: Overall Blocking - DP-RWA on NSF Network

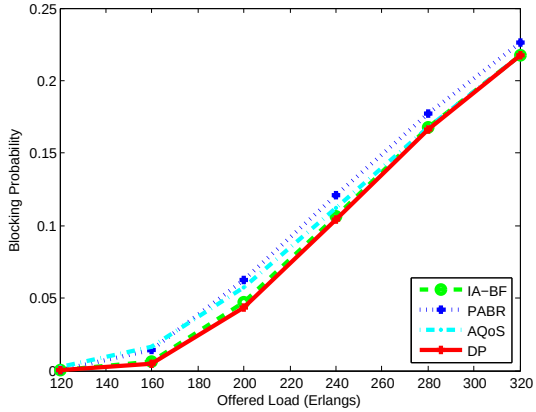
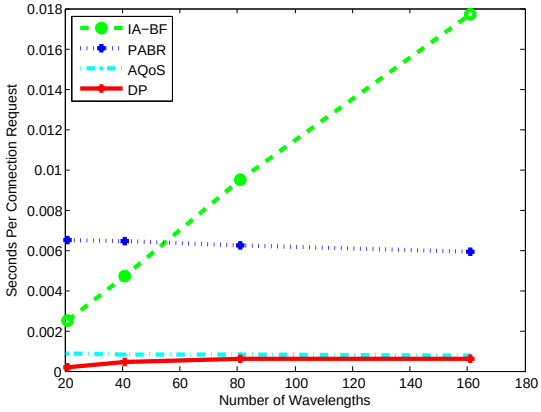


Figure 5.9: Run Time - DP-RWA on NSF Network



PABR by 12.2%. DP-RWA's ability to choose alternate paths was muted by the NSF network's lack of connectivity.

Figure 5.9 shows the run time per connection request. Again, DP-RWA yields the lowest run time, requiring 96.7% less time per connection request than at 161 wavelengths. The time for IA-BF grows linearly with the wavelengths, while DP-RWA grows very slightly.

The average connection length is shown in Figure 5.10. DP-RWA is able to setup connections where the other algorithms fail, however, these connections have a longer

Figure 5.10: Connection Length - DP-RWA on NSF Network

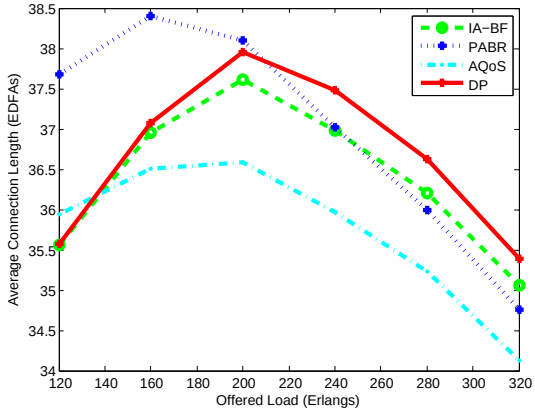
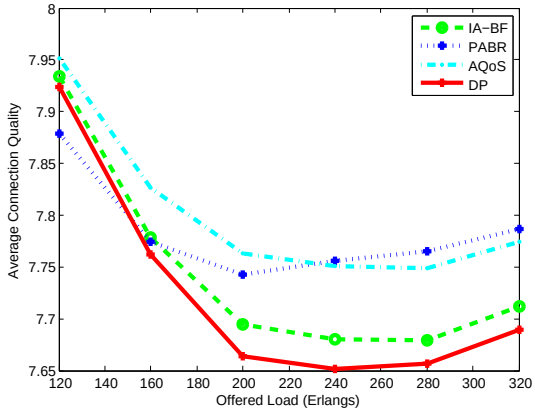


Figure 5.11: Connection Quality (threshold ≈ 5.839) - DP-RWA on NSF Network



length. Due to the increased path length and higher acceptance rate, DP-RWA has the lowest average signal quality, as shown in Figure 5.11. The results on the NSF network are similar to the Mesh results presented earlier.

DP-RWA also performs the best in both the linear dominant and nonlinear dominant scenarios. In the linear dominant scenario (shown in Figure 5.12), DP-RWA blocks 1.2%, 5.4%, and 3.8% fewer connections than IA-BF, AQoS, and PABR.

In the nonlinear dominant scenario, DP-RWA blocks 2.0%, 4.9%, and 9.5% fewer connections than IA-BF, AQoS, and PABR. The results are displayed in Figure 5.13.

Figure 5.12: Overall Blocking - ASE Dominant on NSF

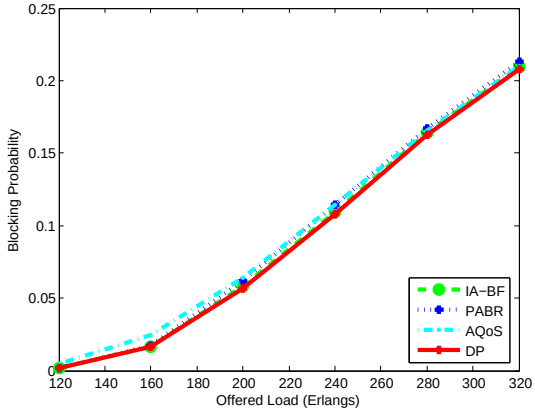
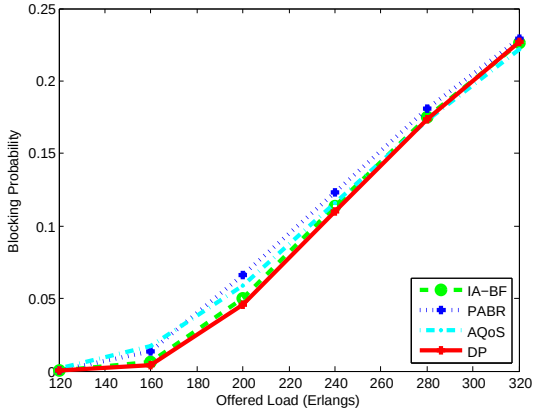


Figure 5.13: Overall Blocking - Nonlinear Dominant on NSF

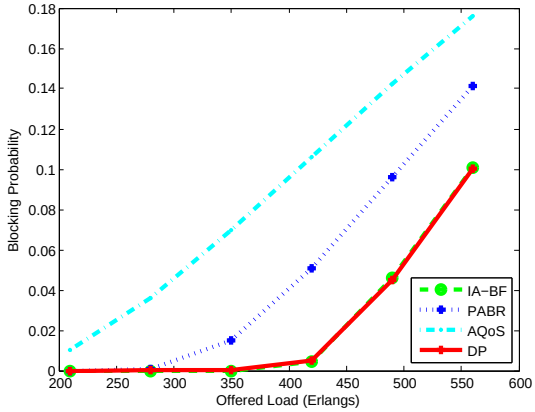


As with the Mesh network results, the gap in performance increases with the nonlinear impairments.

5.3.3 8x8 Mesh Network

Simulations also were run on a standard 8x8 Mesh network topology. The diameter of the 8x8 Mesh network is 112 spans (8 EDFAs / span * 14), larger than the UC Davis Mesh network's diameter of 85 spans. For this reason, the Q factor thresholds are a little smaller. The other parameters were left unchanged from Table 5.2.

Figure 5.14: Overall Blocking - DP-RWA on 8x8 Mesh



The 8x8 Mesh network results are different. In the balanced scenario, IA-BF outperforms DP-RWA by 0.1%. However, DP-RWA greatly outperforms AQoS and PABR, blocking 71.8% and 50.5% fewer connections. This gap in performance is tremendous and much greater than in the NSF or UC Davis topologies. With the high level of connectivity in the 8x8 Mesh network, the centralized approaches of DP-RWA and IA-BF are much better at managing the resources of the network than AQoS and PABR, which are distributed algorithms.

The running time of the 8x8 Mesh is roughly 10 times longer than the NSF network and 5 times longer than the UC Davis Mesh network. At 161 wavelengths, connection requests require approximately 0.104 seconds per connection request for IA-BF. In scenarios where delay is a primary concern, this is likely not acceptable. DP-RWA requires just 4.1% of the run time of IA-BF

The average connection length is presented in Figure 5.4. As the traffic load increases, the path length increases slightly for DP-RWA and IA-BF, while decreasing for PABR and AQoS. In the NSF and UC Davis Mesh topologies, the path length decreased as load increased for all four algorithms. With the high degree of connectivity in the 8x8 Mesh network, DP-RWA and IA-BF are able to find longer alternate

Figure 5.15: Run Time - DP-RWA on 8x8 Mesh

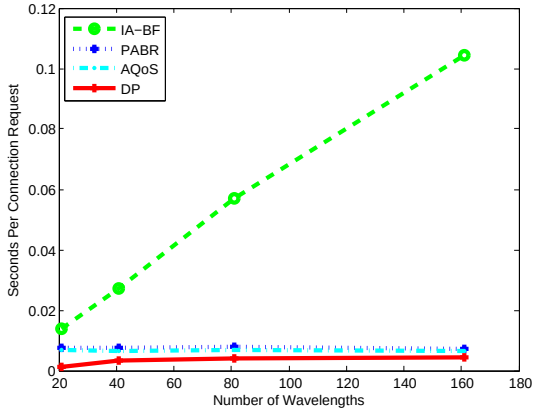
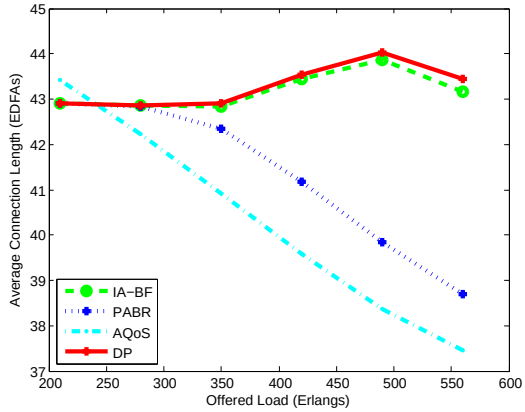


Figure 5.16: Connection Length - DP-RWA on 8x8 Mesh



paths where PABR and AQoS are not. This provides insight into the extremely high blocking rates of PABR and AQoS.

Figure 5.5 shows the average connection quality for accepted connections. While DP-RWA and IA-BF accept roughly the same amount of connections, DP-RWA has a 1.1% higher Q-factor on average. PABR and AQoS have the highest quality, a result of their high blocking probability.

Figure 5.17: Connection Quality (threshold ≈ 5.048) - DP-RWA on 8x8 Mesh

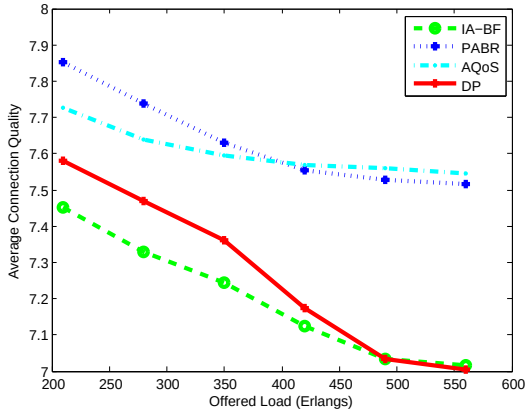
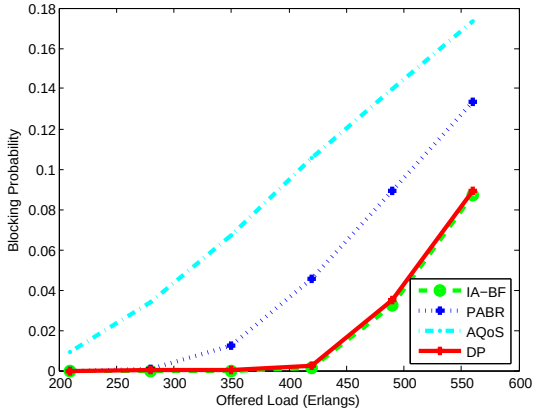


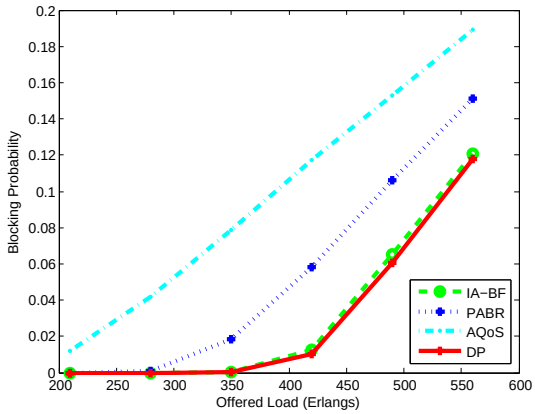
Figure 5.18: Overall Blocking - ASE Dominant on 8x8 Mesh



The results are similar in the ASE dominant scenario (shown in Figure 5.18). IA-BF blocks 0.1% fewer connections than DP-RWA, while DP-RWA is able to soundly outperform AQoS and PABR by 76.5% and 54.9%.

In the nonlinear dominant scenario, DP-RWA outperforms all of the algorithms. It blocks 4.9%, 69.8%, and 43.6% fewer connections than IA-BF, AQoS, and PABR. The results are displayed in Figure 5.19.

Figure 5.19: Overall Blocking - Nonlinear Dominant on 8x8 Mesh



5.3.4 Statistical Analysis

To determine the significance of the differences between DP-RWA and the other algorithms, we performed a statistical analysis. We used 50 random seeds, resulting in millions of connection requests for each data point. The analysis was performed on the UC Davis Mesh network, focusing only on the blocking probability.

The results for the arithmetic mean of the blocking probability are shown in Table 5.3. DP-RWA has the lowest blocking probability in all cases but one, followed by IA-BF, PABR, and AQoS. The lone exception is that PABR has a lower blocking probability at 600 Erlangs.

Table 5.4 presents the standard error of the mean of the blocking probability. The standard error is greatest at the moderate traffic levels. In most cases, the difference between DP-RWA and IA-BF is greater than several times the standard error.

The most convincing argument for DP-RWA's performance is in the p-values presented in Table 5.5. We used a one-sided heteroscedastic T-test to calculate the p-values. In all cases but one, the hypothesis that DP-RWA has the lowest blocking probability is supported by a p-value of less than 0.01. In most cases, the p-values

Table 5.3: Arithmetic Mean of the Blocking Probability

Traffic Load	DP $\hat{\mu}$	IA-BF $\hat{\mu}$	PABR $\hat{\mu}$	AQoS $\hat{\mu}$
60 Erlangs	0.00000000	0.00000000	0.00000000	0.00001622
120 Erlangs	0.00000000	0.00000000	0.00001492	0.00295098
180 Erlangs	0.00030082	0.00081496	0.00452866	0.02079638
240 Erlangs	0.01882528	0.02507148	0.04094620	0.06251236
300 Erlangs	0.07733730	0.08477522	0.09826898	0.11479810
360 Erlangs	0.13907384	0.14418284	0.15242564	0.16507742
420 Erlangs	0.19092860	0.19449840	0.19837020	0.20933274
480 Erlangs	0.23524412	0.23761338	0.23866574	0.24780436
540 Erlangs	0.27323652	0.27490198	0.27441602	0.28163524
600 Erlangs	0.30678242	0.30814374	0.30612852	0.31243502

Table 5.4: Standard Error of the Mean of the Blocking Probability

Traffic Load	DP SE	IA-BF SE	PABR SE	AQoS SE
60 Erlangs	0.0000000	0.0000000	0.0000000	0.0000065
120 Erlangs	0.0000000	0.0000000	0.0000077	0.0002148
180 Erlangs	0.0000335	0.0000084	0.0002205	0.0004708
240 Erlangs	0.0003966	0.0004362	0.0005629	0.0006529
300 Erlangs	0.0005443	0.0005788	0.0005931	0.0005596
360 Erlangs	0.0004272	0.0004244	0.0004363	0.0004451
420 Erlangs	0.0004007	0.0003753	0.0004018	0.0003845
480 Erlangs	0.0003634	0.0003361	0.0003125	0.0003194
540 Erlangs	0.0003219	0.0003913	0.0003583	0.0003699
600 Erlangs	0.0002787	0.0002946	0.0002900	0.0002548

Table 5.5: P-value Results

Hypothesis	DP $\mu <$ PABR μ	DP $\mu <$ IA-BF μ	DP $\mu <$ AQoS μ
180 Erlangs	6.442E-25	2.579E-07	3.699E-41
240 Erlangs	2.422E-50	5.878E-18	4.176E-67
300 Erlangs	1.637E-45	2.442E-15	9.941E-70
360 Erlangs	2.076E-39	1.779E-13	1.980E-64
420 Erlangs	2.580E-23	2.294E-09	6.835E-55
480 Erlangs	1.282E-10	3.685E-06	3.202E-45
540 Erlangs	8.614E-03	7.919E-04	6.038E-31
600 Erlangs	5.541E-02	6.276E-04	5.630E-27

are much smaller than 0.01. This is a product of simulating millions of connection requests per data point.

The only exception is the 600 Erlang case when comparing DP-RWA and PABR. As PABR has a lower blocking probability than DP-RWA, the hypothesis has to be rejected for this case. However, in all other cases, there is very strong statistical evidence that DP-RWA has the lowest blocking probability.

5.4 Conclusion

Our results have shown that DP-RWA is better than IA-BF, AQoS, and PABR with regard to blocking probability for the NSF and UC Davis Mesh networks. IA-BF and DP-RWA perform similarly in the 8x8 Mesh network, although DP-RWA does outperform IA-BF in the nonlinear dominant scenario. In all cases, the performance of DP-RWA improved as the nonlinear impairments became more dominant.

A statistical analysis has shown that the differences in blocking probabilities between DP-RWA and the other algorithms is very significant.

DP-RWA accomplishes this performance despite the fact that DP-RWA uses significantly fewer computational resources than IA-BF. DP-RWA scales well to networks

with a large number of wavelengths. DP-RWA is a very flexible algorithm that can adapt to networks where either linear or nonlinear effects dominate. When blocking probability and running time (or network delay) are jointly considered, DP-RWA has been shown to be the best algorithm.

CHAPTER 6

ANT COLONY OPTIMIZATION BASED RWA APPROACHES

Ant Colony Optimization (ACO) was proposed in [42]. ACO is a metaheurstic from the field of Artificial Intelligence (AI). Metaheuristics, such as ACO, Tabu Search, and Genetic Algorithms, can be applied to a range of Computer Science optimization problems.

This chapter presents our research into ACO approaches for solving RWA. The first section discusses ACO and presents our own implementation. The next section presents our research in applying $\mathcal{MAXMIN}$ ACO ($\mathcal{MM}$-ACO). We then compare the two approaches. The final section compares the performance of ACO-RWA and $\mathcal{MM}$-ACO-RWA to the other RWA algorithms.

6.1 Ant Colony Optimization

Ant colonies are intelligent distributed systems. Even though individual ants have severe limitations (some are even blind), through their social organization they are able to accomplish complex tasks that greatly exceed the capabilities of an individual ant. A detailed presentation on ACO can be found [43].

Ants communicate with each other through chemicals called pheromones. Trail pheromones are used by ants to mark paths on the ground, for example, paths from the nest to a food source. Foraging ants are able to sense these chemicals to find paths to food sources. This mechanism is known as *stigmergy*, where a trace left on the environment by current actions influences the performance of future actions.

Nature's process of stigmergy can be simulated by a computer, providing a foundation for the swarm intelligence based approaches in artificial intelligence. ACO

is a common swarm intelligence approach, but others also exist (Particle Swarm Optimization, Stochastic Diffusion Search).

In a typical ACO implementation, artificial ants are given a task (example: find a shortest path to food). Each individual ant detects the pheromone levels, calculates a set of probabilities of choosing each edge, and then randomly selects an edge. When an ant reaches its goal node, it returns to the colony laying pheromone levels proportional to the path's quality. The idea is that over time the pheromone levels on the bad path choices will evaporate, while the pheromone levels on better paths increase. Usually, the ants will converge on one solution if they are given sufficient time.

Artificial ants are given capabilities that real world ants do not have, for optimization purposes. For example, individual ants are likely unable to adjust their pheromone drop rate based upon a calculation of the path quality. These optimizations generally decrease the time to convergence and increase the final path's quality.

ACO is used to solve problems that can be reduced to finding paths in a graph. Typically these problems are optimization problems known to be NP Complete. The most studied application of ACO is the Traveling Salesman Problem (TSP). ACO has been used for the general network routing program. [43]

Others have attempted to solve RWA using ACO. Two approaches are [44, 45]. In both papers, the authors show that their ACO approach is able to outperform a simple Fixed Path routing scheme, however the authors state that the computational requirements to do so prevent the implementation from scaling to real world applications.

It is difficult to replicate these approaches directly, as many of the important details were omitted from the papers. Presented here is our unique application of ACO to solve RWA, named ACO-RWA or ACO.

The paths are ranked using the same formula as DP-RWA. The weight of path p is calculated using Equation 6.1. α is a parameter in the range of $0 \leq \alpha \leq 1$, used to bias the weight toward best quality ($\alpha = 0$) or shortest path with acceptable quality ($\alpha = 1$). Q_p is the quality of path p and l_p is the length in EDFAs of path p.

$$W_p = (1 - \alpha)\frac{Q_p}{\bar{Q_p}} + \alpha\frac{l_p}{\bar{l_p}} \tag{6.1}$$

$\bar{Q_p}$ and $\bar{l_p}$ represent estimates of the path quality and path length. $\bar{l_p}$ is typically the average of the shortest path and the diameter of the network. $\bar{Q_p}$ is usually the quality factor after $\bar{l_p}$ EDFAs, assuming zero nonlinear impairments. To achieve the lowest blocking probability, a value of $\alpha = 1.0$ is used.

The pseudo-code for the ACO algorithm is shown in Algorithm 6.10. The number of iterations, I and number of ants per iteration, A, are configurable parameters.

INPUT: Connection request from s to t, $G = (V, E)$

Initialize the pheromone levels

For $i = 1$ to I

For $a = 1$ to A

Calculate a path from s to t

Evaporate Pheromone levels

For $a = 1$ to A

Deposit Pheromone levels

Return the best path(s) found

Algorithm 6.10: Ant Colony Optimization - RWA (ACO-RWA)

The initial pheromone levels will be set based upon an approximation of the level of pheromones each iteration will add. Each edge will be initialized with a pheromone

Table 6.1: ACO-RWA Parameters

Parameter	Value	Description
Number of Ants	20	Number of Ants Per Iteration
α	1	Pheromone Power Index
β	5	Heuristic Information Power Index
ρ	0.1	Pheromone Evaporation Rate

level of $\frac{n}{l}$, where n is the number of ants per iteration and l is the length of the shortest path from source to destination.

All ants will deposit pheromones on the edges based upon the length of the path found. Specifically, each ant will add $\frac{1}{l}$ of pheromone, where l is the length of the path found.

Only simple paths with an acceptable signal quality will be considered. If a cycle is detected or if there are no wavelengths available with acceptable quality, the ant's search will terminate. Another ant will start searching in its place, beginning at the source node.

As the ant is searching for a path, it chooses the next destination node probabilistically based upon the pheromone levels and a heuristic function. The probability that ant k, currently at node i will select node j is based upon the formula below.

$$p_{i,j}^k = \frac{[\tau_{i,j}]^\alpha [\eta_{i,j}]^\beta}{\sum_{l \in N_i^k} [\tau_{i,l}]^\alpha [\eta_{i,l}]^\beta}, j \in N_i^k \tag{6.2}$$

In the formula above, $\tau_{i,j}$ represents the pheromone level on edge (i,j). $\eta_{i,j}$ is a heuristic function, which in our simulation is set to $\frac{1}{d_{j,v}}$, where v is the goal node. The values of α and β are set by the user, in our simulations we use the values in Table 6.1. The notation N_i^k represents the set of nodes that are a neighbor of ant k at node i.

After each iteration, some of the pheromone levels will evaporate. This value is controlled by the input parameter ρ.

The values presented here are believed to be appropriate selections based upon suggestions in previous research and extensive tuning. Initially, parameters were set to values described in [43]. Each of the parameters were then varied to determine which combination yielded the best results.

6.1.1 Convergence Analysis

The stochastic nature of ACO-RWA makes it difficult to analyze the time to convergence on the optimal solution. The technique presented here named Associated Discrete Deterministic Process (ADDP) is taken from [46].

The main idea of ADDP is that ACO is a stochastic process, as each iteration is based upon the random paths of the ants. Since the paths selected by the ants are random, so are the pheromone vectors in each iteration. As the number of ants per iteration, S, increases, the impact of their behavior collectively become less stochastic due to statistical laws (such as the Law of Large Numbers). This allows ADDP to approximate the pheromone values. ACO-RWA fits in the framework of ADDP.

The approximation performed by ADDP can be extended to provide an asymptotic convergence result. In [46], Theorem 2.1 shows that after a period containing a fixed number of iterations, the pheromone vectors of ADDP deviate from ACO by less than some δ with a probability of at least 1 - ϵ. This requires that S is sufficiently high based upon ϵ.

6.2 MAX MIN Any Colony Optimization

$\mathcal{MAX-MIN}$ Ant Colony Optimization ($\mathcal{MM}$-ACO) is an extension of ACO proposed in [47]. To our knowledge, no one outside of Montana State University has attempted to use a $\mathcal{MM}$-ACO approach for solving RWA.

The $\mathcal{MM}$-ACO approach presented here, named $\mathcal{MM}$-ACO-RWA or $\mathcal{MM}$-ACO, is a fair approach for Dynamic Lightpath Establishment (DLE), like the other RWA algorithms we are considering.

$\mathcal{MM}$-ACO adds four modifications to ACO. First, in ACO each ant adds pheromone levels. In $\mathcal{MM}$-ACO, only the ant which found the best path adds pheromone levels. This strategy may lead to a stagnation early in the search process, so pheromone levels are bounded by the interval $[\tau_{min}, \tau_{max}]$. Third, pheromone levels are initialized to the upper limit, making the initial search iterations very exploratory. The final modification is related to stagnation. If after several iterations, the best path(s) found do not change, the algorithm will start over.

The pseudo-code for the $\mathcal{MM}$-ACO-RWA algorithm is shown in Algorithm 6.11. As with ACO-RWA, the number of iterations, I and number of ants per iteration, A, are configurable parameters. bs is used to store the best-so-far path(s). $\mathcal{MM}$-ACO-RWA adds another parameter S, which is the number of iterations to detect stagnation. If after S iterations a better bs is not found, the pheromone levels will reset.

The parameters used in our $\mathcal{MM}$-ACO-RWA simulations are given in Table 6.2. As with ACO-RWA, the values presented here are believed to be appropriate selections based upon suggestions in previous research and extensive tuning. Initially, parameters were set to values described in [43]. Each of the parameters were then varied to determine which combination yielded the best results.

INPUT: Connection request from s to t, $G = (V, E)$

For $i = 1$ to I

Initialize the pheromone levels

For $s = 1$ to S

For $a = 1$ to A

Calculate a path from s to t

If (Weight(a_{path}) > Weight(bs_{path}))

$bs = a$

$s = 1$

Evaporate Pheromone levels

Deposit Pheromone levels for bs only

Return the best path(s) found

Algorithm 6.11: MAX MIN Ant Colony Optimization - RWA (MM-ACO-RWA)

Table 6.2: MM-ACO-RWA Parameters

Parameter	Value	Description
Min Pheromone Value	0.1	Minimum value for a pheromone level
Max Pheromone Value	1.0	Maximum value for a pheromone level
Stagnation Value	20	Number of iterations to detect stagnation
Number of Iterations	2	Number of iterations

6.2.1 Convergence Proof

It can be shown that $\mathcal{MM}$-ACO-RWA is guaranteed to find an optimal solution with a probability arbitrarily close to 1 if $\mathcal{MM}$-ACO-RWA is given sufficient time. In practice, the time required may be astronomically large. The proofs presented here are based upon work in [43].

Theorem 3. *Let $P^*(\theta)$ be the probability that $\mathcal{MM}$-ACO-RWA finds an optimal solution at least one time within the first θ iterations. For an arbitrarily small $\epsilon > 0$ and for a sufficiently large θ it holds that*

$$P^*(\theta) \geq 1 - \epsilon, \tag{6.3}$$

$$\lim_{\theta \to \infty} P^*(\theta) = 1. \tag{6.4}$$

Proof. Due to the pheromone trail limits τ_{min} and τ_{max}, any feasible solution (including the optimal one) can be found with a probability $p_{min} > 0$. A trivial lower bound for p_{min} can be given by:

$$p_{min} \geq \hat{p}_{min} = \frac{\tau_{min}^{\alpha}}{(N_C - 1)\tau_{max}^{\alpha} + \tau_{min}^{\alpha}}, \tag{6.5}$$

where N_C is the cardinality of the set of paths. This derivation is based upon the worst-case situation, where the pheromone trail associated with the optimal solution is τ_{min} while the other choices ($N_C - 1$ of them) have pheromone levels of τ_{max}.

Any generic solution $s\prime$, including the optimal solution s^*, can be generated with the probability $\hat{p} \geq \hat{p}_{min}^{n} > 0$, where $n < +\infty$ is the maximum length of a path ($n = |V|$ in our case). Only one ant needs to find an optimal solution, so a lower bound for $P^*(\delta)$ is given by:

$$P^*(\theta) = 1 - (1 - \hat{p})^\theta. \tag{6.6}$$

If we choose a sufficiently large θ, the probability can be made larger than any value $1 - \epsilon$. Thus, it follows that $\lim_{\theta \to \infty} P^*(\theta) = 1$. □

The proofs can be extended to cases where local search and heuristic information are added to $\mathcal{MM}$-ACO-RWA.

Using the technique of *level-reaching*, the process of convergence to the optimal solution can be investigated. The proof presented here is based upon a proof shown in [48].

With $\alpha = 1$, the fitness of lightpaths in $\mathcal{MM}$-ACO-RWA can be grouped into classes based upon their length in spans. Let f_1, ..., f_d, be the different lengths on the finite set S, where the lengths are sorted to that $f_1 > ... \ f_d$. The jth *level set* is defined as $A_j = \{x \in S | f(x) = f_j\}$.

Consider our $\mathcal{MM}$-ACO-RWA algorithm. If we denote the best-so-far solution as $\hat{x} \in A_j$, then $\hat{x}$ will not change until an ant finds a better solution than $\hat{x}$. A lower bound on the probability of finding a better solution than $\hat{x}$ is given by δ_j. The expected running time at level j is thus bounded by $1/\delta_j$.

Theorem 4. *An upper bound on the overall runtime of $\mathcal{MM}$-ACO-RWA before convergence on the optimum path is $1/\delta_1 + ... + 1/\delta_{d-1}$.*

Proof. By the definition of $\hat{x}$, once $\hat{x}$ has left level set j it can never return to A_j. This follows from the best-so-far approach of $\mathcal{MM}$-ACO-RWA. Thus, each level is set is visited at most one time (A_1 to A_{d-1}). In the worst case, $\hat{x}$ visits each level set exactly once before reaching the optimal level set (A_d). □

The proof presented here depends on the best-so-far approach of $\mathcal{MM}$-ACO-RWA. Approaches that use an iteration best ant or all ants to deposit pheromones (such as ACO-RWA) can not be analyzed using this proof.

It should also be noted that the ADDP presented with ACO-RWA can be applied to $\mathcal{MM}$-ACO-RWA.

6.3 ACO-RWA vs. MM-ACO-RWA

To compare our ACO-RWA and $\mathcal{MM}$-ACO-RWA approaches, we ran simulations varying the number of ants. The other parameters listed in Tables 6.1 and 6.2 were left unchanged. The simulations were conducted on the UC Davis Mesh network with a high traffic load and 21 wavelengths per fiber. The physical impairments were balanced, so that neither the linear or nonlinear effects dominated.

The results presented in this section are the average of five simulations, with each seed simulating ten hours of traffic. A statistical analysis is presented at the end of this chapter to show the significance of our results.

Figure 6.1 shows the overall blocking probability. As the number of ants increases, the blocking probability decreases for ACO-RWA. With more than 15 ants per connection request, ACO-RWA has a lower blocking probability than $\mathcal{MM}$-ACO-RWA. Note that this simulation is at a high traffic load and the performance may differ at other traffic levels.

$\mathcal{MM}$-ACO-RWA is not as sensitive to the number of ants; it is able to find good routes with fewer numbers of ants. This is likely caused by two distinctions. First, $\mathcal{MM}$-ACO-RWA adds pheromones to the best-so-far path only. This leads to stagnation, meaning many of the ants find the same path. Thus, adding additional ants are less beneficial. The second reason is that $\mathcal{MM}$-ACO-RWA restarts after

Figure 6.1: Overall Blocking - Ants Per Request

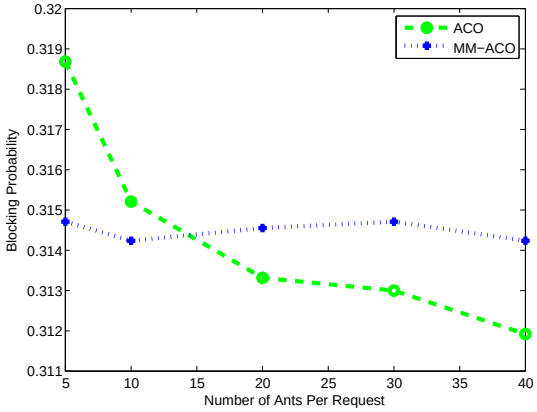
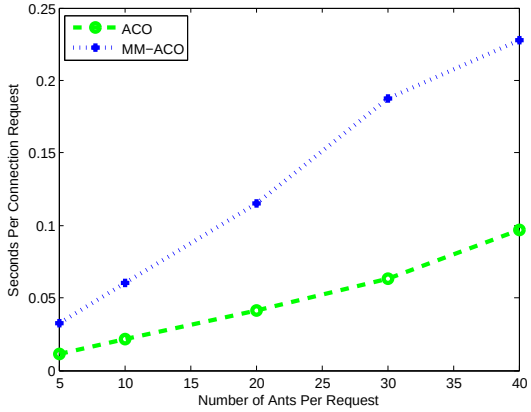


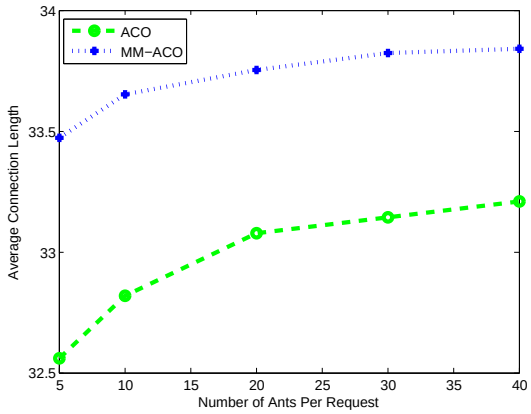
Figure 6.2: Run Time - Ants Per Request



stagnation is detected. This leads to additional iterations, meaning each ant generates more paths. With each ant attempting additional paths, there is less gain from adding additional ants.

The run time per connection request is shown in Figure 6.2. $\mathcal{MM}$-ACO-RWA has the highest run time, a result of the restarting after stagnation. With just 20 ants per connection request, both algorithms have a very high run time (0.04123 seconds for ACO-RWA and 0.11513 seconds for $\mathcal{MM}$-ACO-RWA). At that rate, only 8.69 paths are computed per second for $\mathcal{MM}$-ACO-RWA (24.22 for ACO-RWA).

Figure 6.3: Connection Length - Ants Per Request



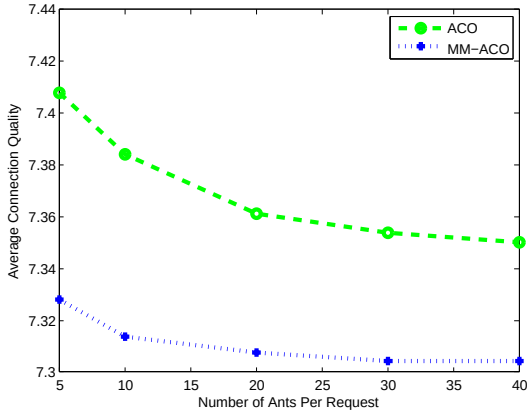
The run times compare very unfavorably to the other RWA algorithms. The average for IA-BF is 3.8985e-3 and 5.3056e-4 for DP-RWA. The high run times likely mean that ACO based RWA algorithms are not a practical solution for some network applications.

It should be noted that our ACO-RWA approaches are centralized. A significant advantage of ACO is the ability to distribute the computation, which would achieve considerable speedup. ACO-RWA uses its global knowledge to evaluate paths, so changes would be required to distribute the ACO-RWA route calculation.

As the number of ants increase, the connection length of accepted connections increases, as shown in Figure 6.3. This may be counter-intuitive to some. With a greater number of ants, the algorithms are able to search more exhaustively. They accept more connection requests, but the average path length increases.

ACO-RWA's average path length is 32.9615 spans, while the average for $\mathcal{MM}$-ACO-RWA is 33.7076 spans. ACO-RWA's ability to find shorter paths that use fewer resources is a chief reason why ACO-RWA has a lower blocking probability. This is likely a result of the differences in the way ACO-RWA and $\mathcal{MM}$-ACO-RWA search for

Figure 6.4: Connection Quality (threshold ≈ 5.526) - Ants Per Request



paths. While ACO-RWA's approach is very exploratory, $\mathcal{MM}$-ACO-RWA's rapidly converges to a solution.

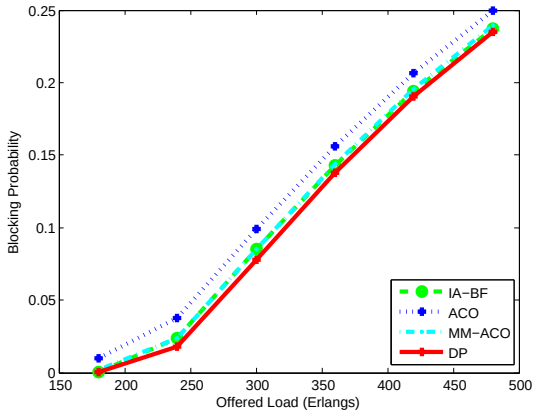
Figure 6.4 displays the average connection quality. ACO-RWA has a 0.82% higher average signal quality on average. This is a large accomplishment considering ACO-RWA has a higher connection acceptance rate. ACO-RWA accomplishes this by finding shorter paths. A p-value of 0.001636 from a one-sided heteroscedastic T-test shows this small difference is significant.

The results presented above are for a simulation with a very high traffic load. The results may be different at lower traffic levels and with different network topologies.

6.4 Results

This section presents a more detailed analysis comparing ACO-RWA and $\mathcal{MM}$-ACO-RWA against the other RWA algorithms. Results for the NSF, Mesh, and 8x8 Mesh network are presented. We also discuss scenarios where linear and nonlinear impairments dominate.

Figure 6.5: Overall Blocking - ACO-RWA on Mesh Network



ACO-RWA and $\mathcal{MM}$-ACO-RWA are compared against IA-BF and DP-RWA. These algorithms were selected as they represent the state-of-the-art in RWA algorithms. The remaining algorithms are not displayed in the graphs to maximize the graph's readability, but will also be considered in this section.

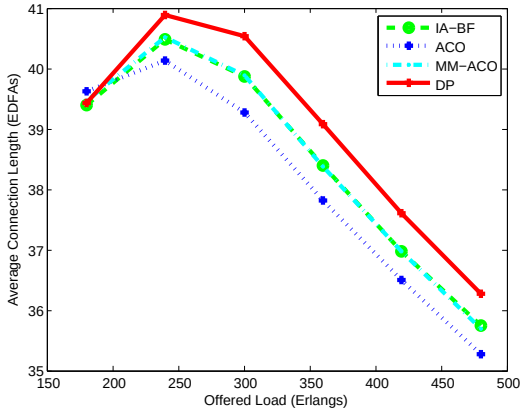
6.4.1 UC Davis Mesh Network

The overall blocking probabilities for the UC Davis Mesh network are shown in Figure 6.5. DP-RWA and IA-BF both have lower blocking probabilities than ACO-RWA and $\mathcal{MM}$-ACO-RWA. ACO-RWA and $\mathcal{MM}$-ACO-RWA block 13.1% and 4.3% more connections, respectively, than DP-RWA. IA-BF has a slightly lower blocking probability than $\mathcal{MM}$-ACO-RWA.

Between the two AI algorithms, $\mathcal{MM}$-ACO-RWA offers the lowest blocking probability. The average $\mathcal{MM}$-ACO-RWA blocking probability is 0.114786, compared to 0.126400 for ACO-RWA.

Both ACO-RWA and $\mathcal{MM}$-ACO-RWA have a lower blocking probability than PABR, LORA, AQoS, QM, and IA-FF. These algorithms are not displayed in the graph to avoid cluttering the results.

Figure 6.6: Connection Length - ACO-RWA on Mesh Network



A statistical analysis of the significance of the blocking probability differences is presented in Section 6.4.4.

ACO-RWA has the shortest average accepted connection length of 38.105, followed by IA-BF (38.478), $\mathcal{MM}$-ACO-RWA (38.485), and DP-RWA (38.969). The results are shown in Figure 6.6. ACO-RWA's inability to find longer routes is likely a main reason it has a higher blocking probability.

Figure 6.7 presents the average connection quality for accepted connections. ACO-RWA has the highest quality, a product of its lower connection acceptance rate and connection length. ACO-RWA is followed by IA-BF, $\mathcal{MM}$-ACO-RWA, and DP-RWA. The main objective of RWA is to maximize the number of accepted connection requests with sufficient quality, not the average connection quality.

The results presented earlier in this section are from a balanced scenario, where the linear and nonlinear impairments are roughly equivalent. To evaluate as many scenarios as possible, simulations were run where the linear and nonlinear impairments dominate.

The overall blocking probabilities from the ASE dominant scenario are shown in Figure 6.8. DP-RWA and IA-BF again have the lowest blocking, followed by

Figure 6.7: Connection Quality (threshold ≈ 5.526) - ACO-RWA on Mesh Network

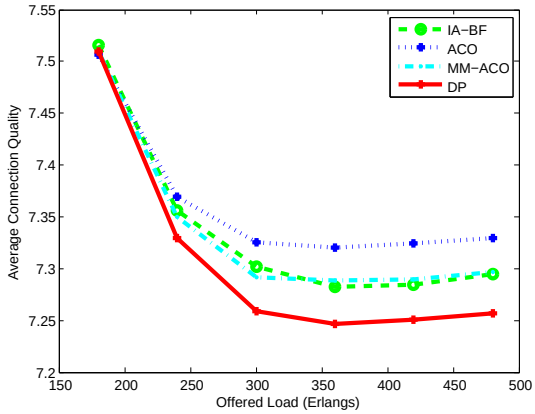
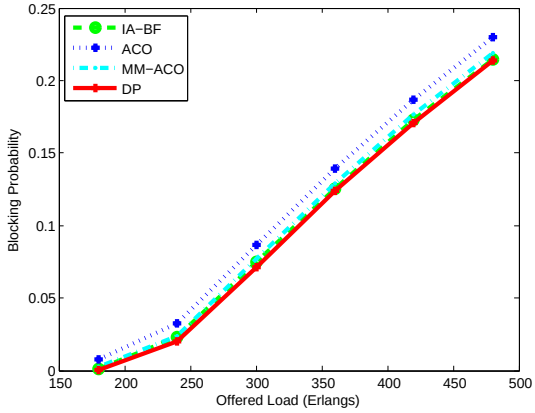


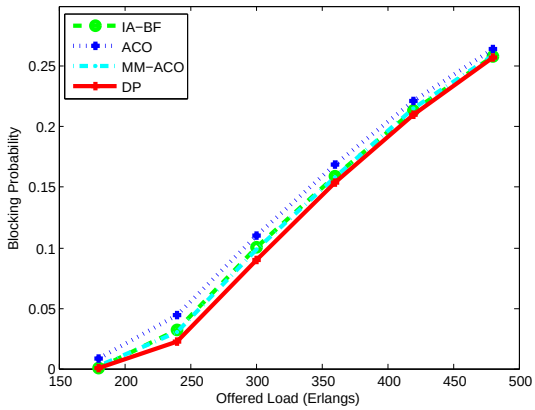
Figure 6.8: Overall Blocking - ACO-RWA on Mesh Network with ASE Dominant



$\mathcal{MM}$-ACO-RWA, IA-FF, ACO-RWA, PABR, AQoS, and QM. The performance of $\mathcal{MM}$-ACO-RWA is close to IA-BF, as $\mathcal{MM}$-ACO-RWA's bias toward shorter paths works well in scenarios where the linear impairments (ASE) dominate.

Figure 6.9 shows the overall blocking probability for the nonlinear dominant scenario. ACO-RWA has the highest blocking probability, then $\mathcal{MM}$-ACO-RWA, IA-BF, and DP-RWA. ACO-RWA and $\mathcal{MM}$-ACO-RWA attempt shorter paths and are unable to route connection requests away from congested areas in the network. Both

Figure 6.9: Overall Blocking - ACO-RWA on Mesh Network with Nonlinear Dominant



ACO-RWA and $\mathcal{MM}$-ACO-RWA have a lower blocking probability than IA-FF and QM.

6.4.2 NSF Network

Figure 6.10 shows the overall blocking rate in the NSF network. As in the Mesh network, ACO-RWA and $\mathcal{MM}$-ACO-RWA have a higher blocking probability than DP-RWA and IA-BF. $\mathcal{MM}$-ACO-RWA is the best performing AI algorithm, blocking 11.06% fewer connections than ACO-RWA. DP-RWA is the top performing overall algorithm, followed by IA-BF.

$\mathcal{MM}$-ACO-RWA and ACO-RWA outperform PABR, LORA, and IA-FF in terms of blocking probability, but they are unable to outperform AQoS and QM. In the loosely connected NSF network, AQoS and QM perform best when compared against the other RWA algorithms.

The average connection length of accepted connections is shown in Figure 6.11. There are large differences between the algorithms. ACO-RWA has the shortest length of 35.62 spans. DP-RWA is able to setup connections with a longer length, having

Figure 6.10: Overall Blocking - ACO-RWA on NSF Network

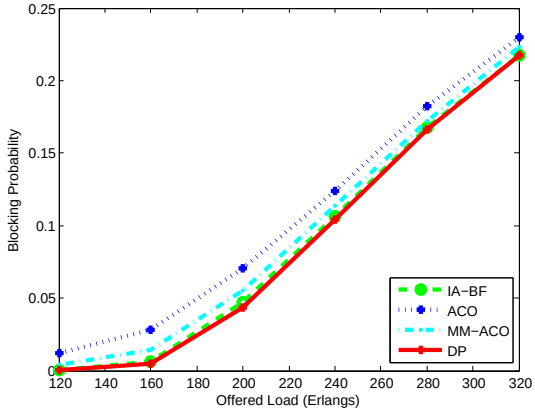
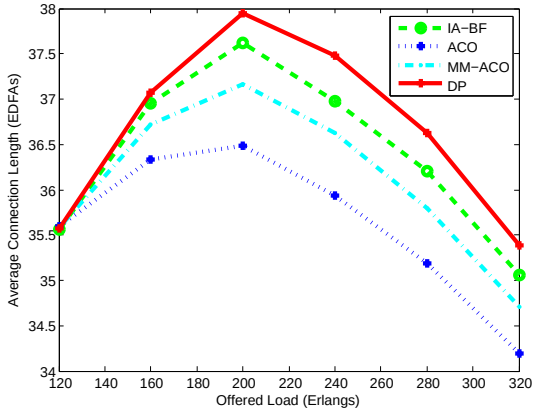


Figure 6.11: Connection Length - ACO-RWA on NSF Network



the highest average of 36.40 spans. IA-BF has the second highest connection length (36.39), followed by $\mathcal{MM}$-ACO-RWA (36.09).

Figure 6.12 shows the average connection quality. ACO-RWA has the highest quality, followed by $\mathcal{MM}$-ACO-RWA, IA-BF and DP-RWA. The ordering is a reversal of the overall blocking ranking, showing the trade-off. Algorithms which accept more connections will have a lower average signal quality.

Figure 6.12: Connection Quality (threshold ≈ 5.839) - ACO-RWA on NSF Network

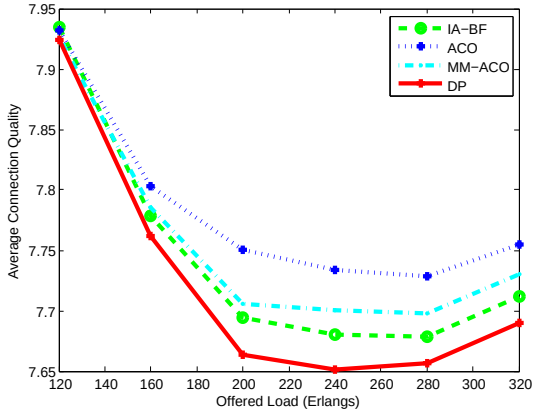
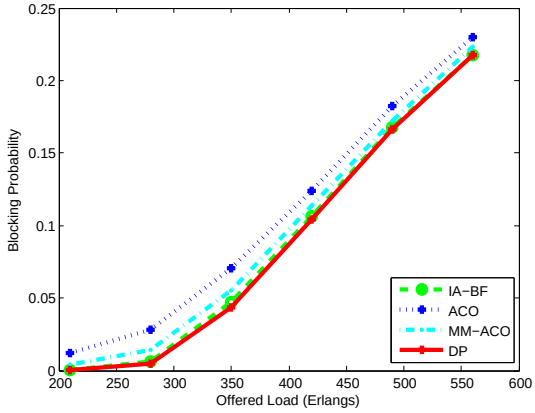


Figure 6.13: Overall Blocking - ACO-RWA on 8x8 Mesh Network

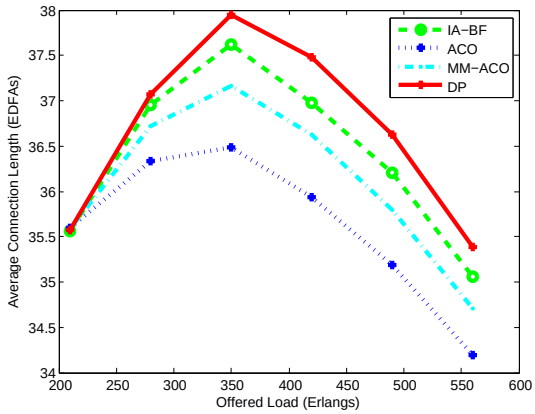


6.4.3 8x8 Mesh Network

Figure 6.13 shows the overall blocking results for the 8x8 Mesh network. DP-RWA again has the lowest blocking probability, narrowly outperforming IA-BF. Among the AI algorithms, $\mathcal{MM}$-ACO-RWA has an average blocking probability of 0.028627, lower than ACO-RWA's 0.036463.

Both ACO-RWA and $\mathcal{MM}$-ACO-RWA have a lower blocking probability than PABR, LORA, AQoS, QM, and IA-FF. While ACO-RWA and $\mathcal{MM}$-ACO-RWA can

Figure 6.14: Connection Length - ACO-RWA on 8x8 Mesh Network



not match the blocking probability of DP-RWA and IA-BF, they are able to outperform the other RWA algorithms in most scenarios.

The average connection length is shown in Figure 6.14. ACO-RWA has the lowest average connection length, followed by $\mathcal{MM}$-ACO-RWA. The AI algorithms deposit pheromones based on the length of the path, biasing the results toward shorter paths. DP-RWA has the longest path length, followed by IA-BF.

Figure 6.15 displays the average connection quality. ACO-RWA has the highest average quality, followed by $\mathcal{MM}$-ACO-RWA. They have the highest quality due to having a lower connection acceptance rate. DP-RWA has the lowest average connection quality, while IA-BF has the second lowest.

6.4.4 Statistical Analysis

To determine the significance of the differences between ACO-RWA, $\mathcal{MM}$-ACO-RWA, and the other RWA algorithms, we performed a statistical analysis. We used 50 random seeds, resulting in millions of connection requests for each data point. The analysis was performed on the UC Davis Mesh network under the balanced scenario, focusing only on the blocking probability.

Figure 6.15: Connection Quality (threshold ≈ 5.048) - ACO-RWA on 8x8 Mesh Network

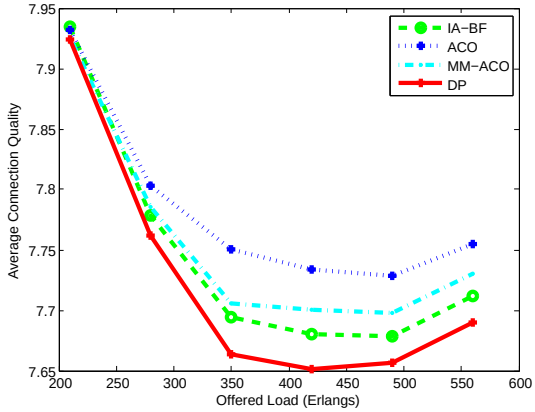


Table 6.3: Arithmetic Mean of the Blocking Probability - ACO-RWA

Load	ACO $\hat{\mu}$	MM-ACO $\hat{\mu}$	DP $\hat{\mu}$	IA-BF $\hat{\mu}$	PABR $\hat{\mu}$	AQoS $\hat{\mu}$
60	0.0102754	0.0017852	0.0000000	0.0000000	0.0000000	0.0000162
120	0.0097678	0.0016581	0.0000000	0.0000000	0.0000149	0.0029509
180	0.0116944	0.0023159	0.0003008	0.0008149	0.0045286	0.0207963
240	0.0401891	0.0255446	0.0188252	0.0250714	0.0409462	0.0625123
300	0.0990690	0.0848917	0.0773373	0.0847752	0.0982689	0.1147981
360	0.1579320	0.1454255	0.1390738	0.1441828	0.1524256	0.1650774
420	0.2078869	0.1967897	0.1909286	0.1944984	0.1983702	0.2093327
480	0.2504209	0.2400469	0.2352441	0.2376133	0.2386657	0.2478043
540	0.2873219	0.2776639	0.2732365	0.2749019	0.2744160	0.2816352
600	0.3199957	0.3109823	0.3067824	0.3081437	0.3061285	0.3124350

The results for the arithmetic mean of the blocking probability are shown in Table 6.3. $\mathcal{MM}$-ACO-RWA has a lower blocking probability than ACO-RWA, PABR, and AQoS, on average. Both DP-RWA and IA-BF both outperform $\mathcal{MM}$-ACO-RWA, although the differences between $\mathcal{MM}$-ACO-RWA and IA-BF are small.

Table 6.4 presents the standard error of the mean of the blocking probability. The standard error is very small for all algorithms, peaking at the moderate traffic levels. The differences between the algorithms is greater than many times the standard error,

Table 6.4: Standard Error of the Mean of the Blocking Probability - ACO-RWA

Load	ACO SE	MM-ACO SE	DP SE	IA-BF SE	PABR SE	AQoS SE
60	0.0001865	0.0003120	0.0000000	0.0000000	0.0000000	0.0000065
120	0.0001300	0.0002246	0.0000000	0.0000000	0.0000077	0.0002148
180	0.0002389	0.0002031	0.0000335	0.0000084	0.0002205	0.0004708
240	0.0004664	0.0005354	0.0003966	0.0004362	0.0005629	0.0006529
300	0.0005963	0.0006212	0.0005443	0.0005788	0.0005931	0.0005596
360	0.0004233	0.0004538	0.0004272	0.0004244	0.0004363	0.0004451
420	0.0003522	0.0004765	0.0004007	0.0003753	0.0004018	0.0003845
480	0.0003359	0.0003849	0.0003634	0.0003361	0.0003125	0.0003194
540	0.0004020	0.0004218	0.0003219	0.0003913	0.0003583	0.0003699
600	0.0002886	0.0003573	0.0002787	0.0002946	0.0002900	0.0002548

in most situations. This provides compelling evidence that our conclusions regarding blocking probability are significant.

In the majority of cases, the standard error for ACO-RWA and $\mathcal{MM}$-ACO-RWA is somewhat higher than the other RWA algorithms. This is expected as the ACO algorithms are stochastic and thus more sensitive to random number generator fluctuations. $\mathcal{MM}$-ACO-RWA has a greater standard error than ACO-RWA.

The most convincing argument for $\mathcal{MM}$-ACO-RWA's performance is in the p-values presented in Tables 6.5 and 6.6. We used a one-sided heteroscedastic T-test to calculate the p-values. The first table compares $\mathcal{MM}$-ACO-RWA against ACO-RWA, PABR, and AQoS. The p-values are significantly below 0.01, strongly supporting our hypothesis that $\mathcal{MM}$-ACO-RWA has a lower overall blocking probability than ACO-RWA, PABR, and AQOS.

Table 6.6 compares $\mathcal{MM}$-ACO-RWA against IA-BF and DP-RWA. In all scenarios, our hypothesis that DP-RWA has a lower blocking probability than $\mathcal{MM}$-ACO-RWA is supported. For traffic loads of 240 and 300 Erlangs, we have to reject our

Table 6.5: P-value Results - ACO-RWA #1

Load	MM-ACO $\mu <$ ACO μ	MM-ACO $\mu <$ PABR μ	MM-ACO $\mu <$ AQoS μ
180	3.2155E-50	3.8325E-11	1.9105E-45
240	4.8714E-37	5.5771E-36	2.0972E-64
300	6.1863E-30	3.1619E-28	1.4732E-57
360	1.6995E-36	4.0854E-19	3.0576E-52
420	7.9671E-33	6.8805E-03	2.3354E-36
480	1.6133E-36	3.4998E-03	9.5343E-28
540	3.8988E-30	4.1238E-08	1.6557E-10
600	6.0299E-35	1.0728E-17	7.5168E-04

Table 6.6: P-value Results - ACO-RWA #2

Load	MM-ACO $\mu >$ IA-BF μ	MM-ACO $\mu >$ DP μ
180	2.3076E-09	1.6373E-13
240	2.4964E-01	1.4952E-16
300	4.4612E-01	7.8041E-15
360	2.5271E-02	4.0399E-17
420	1.5494E-04	2.3242E-15
480	4.0960E-06	1.0113E-14
540	3.4792E-06	5.3259E-13
600	1.3370E-08	5.9772E-15

hypothesis that $\mathcal{MM}$-ACO-RWA $\mu >$ IA-BF μ. At higher traffic levels (360 Erlangs and higher), the hypothesis is supported.

Both ACO-RWA and $\mathcal{MM}$-ACO-RWA are interesting RWA algorithms. We have shown that $\mathcal{MM}$-ACO-RWA outperforms AQoS, PABR, and ACO-RWA by a statistically significant margin. While DP-RWA has a lower blocking probability, $\mathcal{MM}$-ACO-RWA has the advantage of extending to distributed applications.

CHAPTER 7

PERFORMANCE EVALUATION AND COMPARISON OF RWA STRATEGIES

This section includes conclusions drawn from a wide array of scenarios, including different network topologies and modifying some of the parameters listed in Table 5.2. The goal of this chapter is to expand the current understanding of RWA performance in transparent optical networks. RAPTOR gives us the ability to compare algorithms using various scenarios and new performance metrics. We explore domains where both linear and nonlinear impairments dominate.

Our findings have been submitted for publication. A conference version of the paper [49] and a journal version of the paper [50] are waiting to be accepted.

7.1 Signal Quality

Physically aware algorithms have a tremendous benefit to signal quality. Consider two simple routing algorithms: a standard shortest path algorithm (SP) and a quality aware shortest path algorithm (QA SP) which blocks paths with unacceptable signal quality level. QA SP has a 7.6% higher average signal quality (shown in Figure 7.1). QA SP connections spend only 9.61% of their time below the quality threshold, for SP the number is 25.63%. (shown in Figure 7.2).

The benefit to quality comes with a minimal cost. The top performing physically aware algorithms (DP-RWA and IA-BF) have a blocking probability that is comparable to SP, despite the fact they block connections with insufficient quality. Figure 7.3 shows the blocking probability of DP-RWA and IA-BF compared to SP.

At low traffic levels, DP-RWA and IA-BF have a lower blocking probability while at high traffic levels SP has a lower blocking probability. This is due to DP-RWA

Figure 7.1: Connection Quality (threshold ≈ 5.526) - SP vs. QA SP

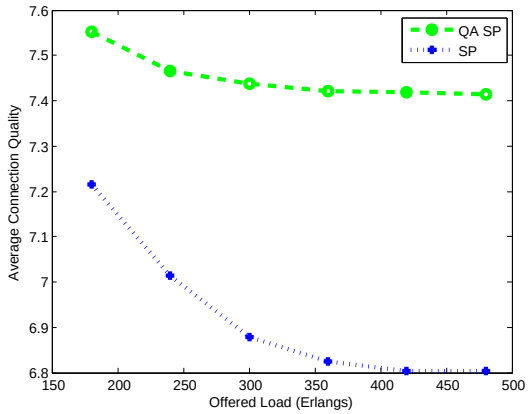


Figure 7.2: Percent Time Below Threshold - SP vs. QA SP

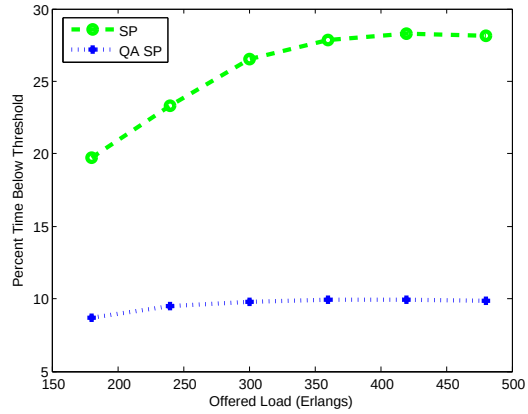
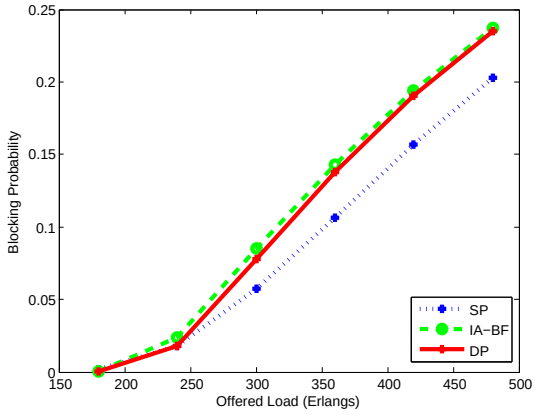


Figure 7.3: Overall Blocking - DP-RWA and IA-BF vs SP



and IA-BF's ability to find alternate, but longer, paths. This is advantageous at low to moderate traffic levels. Network operators are unlikely to tolerate blocking probabilities above 10%, so the low to moderate traffic levels (where DP-RWA and IA-BF have the lowest blocking) are likely to be predominant.

When considering physically aware options, adaptive routing algorithms offer a significant improvement over both fixed path routing or fixed alternate routing. QA SP, a fixed alternate routing algorithm has the worst performance in every simulation we attempted. For the UC Davis Mesh network, DP-RWA blocks 29.3% fewer connections when compared to QA SP.

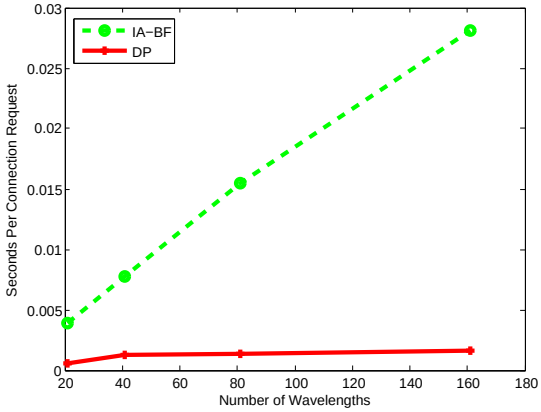
7.2 DP-RWA versus IA-BF

DP-RWA and IA-BF choose paths in a similar manner and thus have a similar blocking performance. IA-BF finds the shortest path on each wavelength, ranks them according to distance, and then tests their quality. DP-RWA simply attempts to find the shortest paths with acceptable quality, pruning intermediate paths with an unacceptable signal quality. The subtle difference of eliminating paths with unacceptable signal quality of DP-RWA allows the algorithm to slightly lower the blocking probability, in general. The additional connections from the higher acceptance rate have a side effect of lowering the signal quality by a small margin.

A significant benefit of DP-RWA over IA-BF is a dramatic reduction in route computation time. Figure 7.4 compares the difference in run time between DP-RWA and IA-BF. DP-RWA requires just 5.6% of the run time of IA-BF.

As optical networks move toward connections with shorter durations (and eventually to Optical Burst Switching (OBS) and Optical Packet Switching (OPS)) this

Figure 7.4: Run Time - DP-RWA vs IA-BF



distinction becomes very important. This is also significant in applications where network delay is a primary concern.

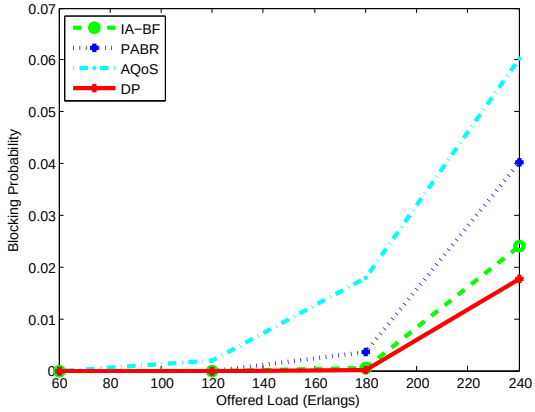
The run time of IA-BF grows linearly with the number of wavelengths, while DP-RWA does not. As the number of wavelengths per fiber continues to increase, IA-BF's run time will become an even greater disadvantage.

PABR and AQoS offer run times similar to DP-RWA. However, PABR and AQoS have an additional overhead of recomputing the edge weights each time a connection is added or deleted. For AQoS, the edge weight computation time grows linearly with the number of wavelengths.

7.3 Traffic Levels

The differences among the algorithm's blocking probabilities is greatest under a moderate traffic load. At lower traffic levels, the algorithms have a very similar level of minimal blocking. This gap increases in the moderate traffic ranges, only to shrink again as the load becomes high.

Figure 7.5: Overall Blocking - Low Traffic Levels



The overall blocking probability at lower traffic levels is shown in Figure 7.5. It is important to consider lower traffic levels as optical networks will ideally operate in these ranges.

DP-RWA and IA-BF accept all connection requests in the 120 or fewer Erlang scenarios, and nearly all requests at 180 Erlangs. Both algorithms easily outperform PABR and AQoS in term of blocking probability.

7.4 Traffic Models

Most of the research in RWA algorithms has focused on the same traffic model: a Poisson process generating connection requests. The destination of the request is chosen randomly with each node having an equal probability of being selected. This results in a uniform spreading of the traffic.

RAPTOR has three different options for choosing the destination. The first model is the standard uniform model, as described above. Another option is a distance weighted model, where the probability of selecting a destination node is proportional to the node's distance from the source. In other words, there is a bias toward destina-

tions with a greater distance. The final option is an inverse distance weighted model, where the probability of selecting a destination node is proportional to the inverse of the node's distance from the source. This creates a bias toward destinations with a small distance.

7.4.1 Distance Weighted Model

In the distance weighted traffic model, destination nodes are selected randomly with a bias toward destinations with a greater distance. The probability of selecting node j as a destination from node i, $p_{i,j}$ is proportional to the distance between i and j, $d_{i,j}$. The formula is given in Equation 7.1.

$$p_{i,j} = \frac{d_{i,j}}{\sum_{i \neq k} d_{i,k}} \tag{7.1}$$

Networks using a distance weighted traffic model can accept fewer connection requests, as each connection request takes more resources due to its longer length. For the UC Davis Mesh network, the distance weighted traffic model accepts between 15% and 20% fewer connections.

The overall blocking probability using a distance weighted traffic model is displayed in Figure 7.6. DP-RWA has the lowest blocking probability, followed by IA-BF, PABR, and AQoS. DP-RWA blocks 6.0%, 14.7%, and 39.2% fewer connections than IA-BF, PABR, and AQoS. The differences are larger than in the uniform distribution, showing that DP-RWA has a great ability to adapt to various network conditions.

The average connection length for accepted connections is shown in Figure 7.7. For DP-RWA, the average accepted connection has a length of 47.9 EDFAs. This is a 26% increase over the random uniform traffic model for DP-RWA (which had an average of 38.0). The other RWA algorithms have similar increases.

Figure 7.6: Overall Blocking - Distance Weighted Traffic Model

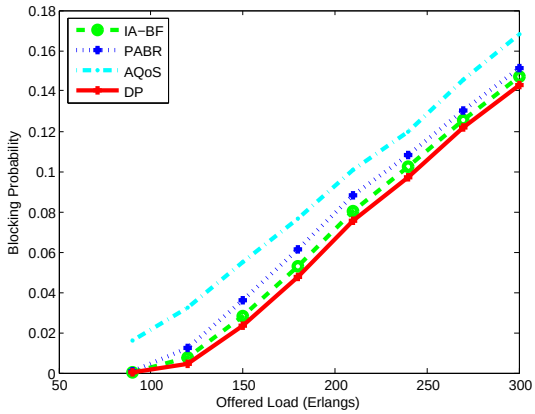


Figure 7.7: Connection Length - Distance Weighted Traffic Model

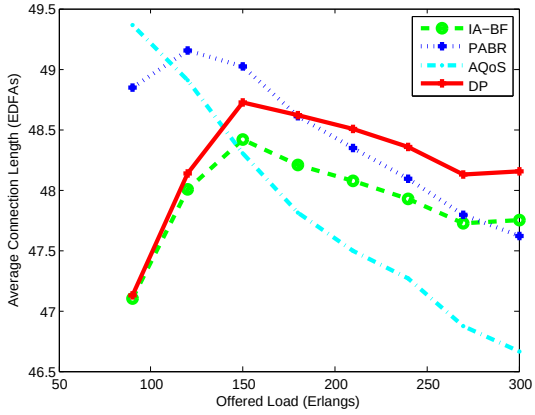


Figure 7.8: Connection Quality (threshold ≈ 5.526) - Distance Weighted Traffic Model

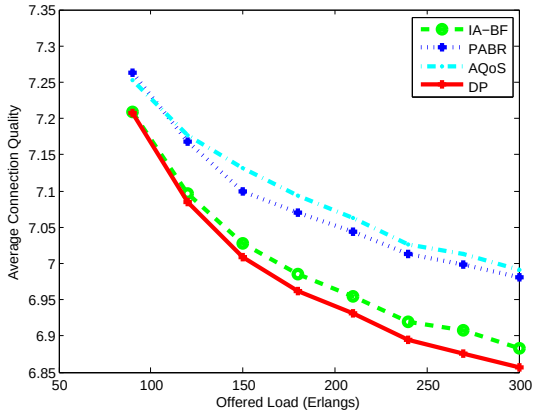


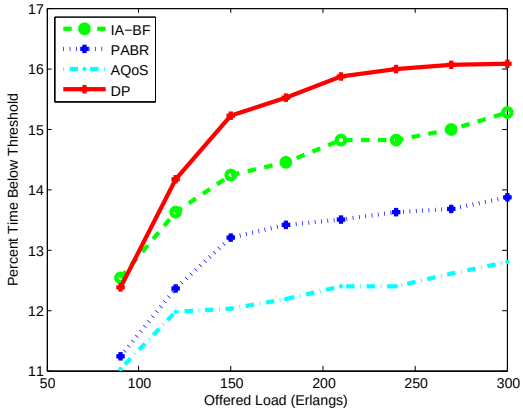
Figure 7.8 shows the average connection quality using the distance weighted traffic model. In terms of signal quality, DP-RWA has the worst performance. This is a result of DP-RWA having a higher acceptance rate, adding additional noise into the network.

Figure 7.9 shows the percent time the average connection spends below the quality threshold. Under this metric, the performance of the algorithms (from best to worst) is AQoS, PABR, IA-BF, and DP-RWA. This list is identical to a ranking of the algorithms by their acceptance rate (from lowest to highest). There is a trade-off. Algorithms that accept more connections have a lower signal quality and spend more time below the quality threshold, on average.

7.4.2 Inverse Distance Weighted Model

In the inverse distance weighted traffic model, destination nodes are selected randomly with a bias toward destinations with a smaller distance. The probability of selecting node j as a destination from node i, $p_{i,j}$ is proportional to the inverse of the distance between i and j, $d_{i,j}^{-1}$. The formula is given in Equation 7.2.

Figure 7.9: Percent Time Below Threshold - Distance Weighted Traffic Model



$$p_{i,j} = \frac{d_{i,j}^{-1}}{\sum_{i \neq k} d_{i,k}^{-1}} \tag{7.2}$$

When using the inverse distance weighted traffic model, the network can handle additional connection requests, as each connection request uses fewer resources due to its reduced connection length. For the UC Davis Mesh network, the inverse distance weighted traffic model can support twice as many connections as the uniform model.

The overall blocking probability is displayed in Figure 7.10. There is only a minimal difference between the algorithms. The RWA choice is simpler for shorter connection requests, reducing the advantages of the more sophisticated algorithms. DP-RWA has the lowest blocking probability, followed by PABR, IA-BF, and AQoS. This is one of the few scenarios where PABR outperforms IA-BF.

Figure 7.11 shows the average connection length of accepted connections. As expected, the average length of 27.1 is significantly lower than the random uniform (38.0) and distance weighted (47.9) models.

The average connection quality for accepted connections is shown in Figure 7.12. Using this metric, the ranking of the algorithms (from best to worst) is AQoS, PABR,

Figure 7.10: Overall Blocking - Inverse Distance Weighted Traffic Model

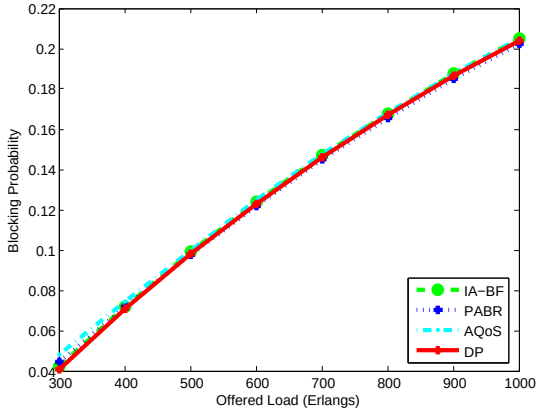


Figure 7.11: Connection Length - Inverse Distance Weighted Traffic Model

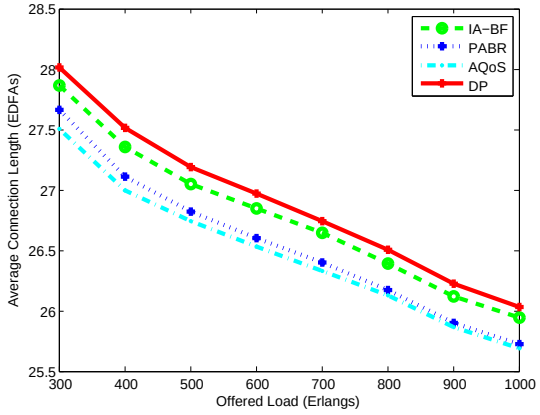


Figure 7.12: Connection Quality (threshold ≈ 5.526) - Inverse Distance Weighted Traffic Model

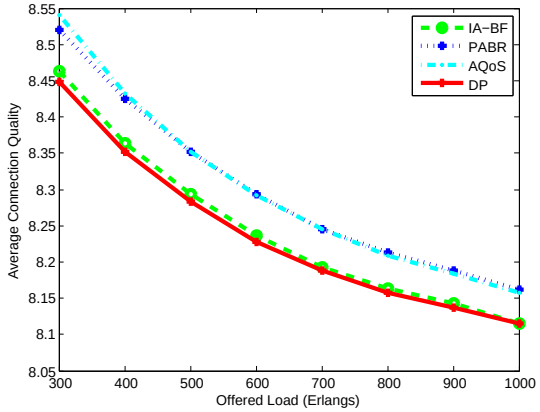
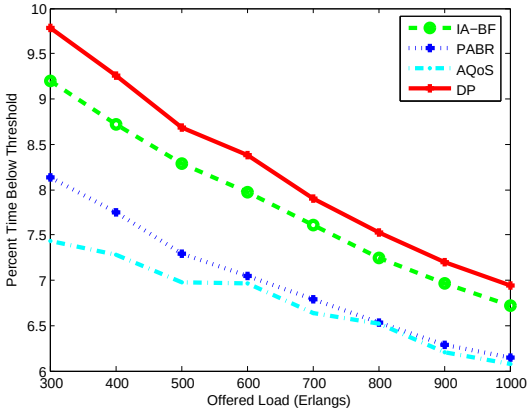


Figure 7.13: Percent Time Below Threshold - Inverse Distance Weighted Traffic Model

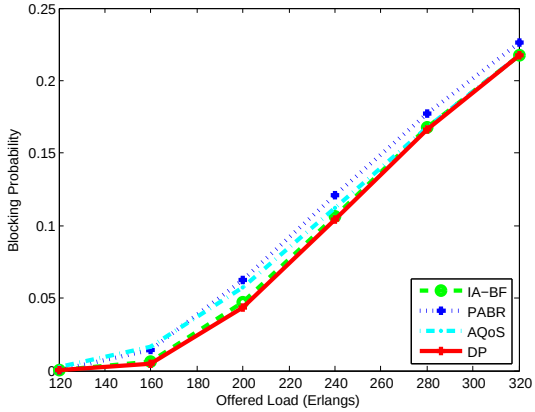


IA-BF, and DP-RWA. This is the same ranking as the random uniform and distance weighted scenarios.

Figure 7.13 shows the percent time the average connection spends below the quality threshold. DP-RWA has the highest percent time below threshold, due to DP-RWA having a higher acceptance rate which introduces additional noise into the network.

RAPTOR currently supports only a Poisson process to generate the connection requests. A Poisson process effectively simulates circuit switched networks such as

Figure 7.14: Overall Blocking - NSF Network



the current generation of optical networks. It would be beneficial to add a Pareto distribution. Pareto distributions are used to simulate packet switched networks, such as optical networks using Optical Burst Switching (OBS) [51] and Optical Packet Switching (OPS) [52].

7.5 Network Topologies

It is important to consider other network topologies. Our previous research has focused on the UC Davis Mesh network. The NSF network (16 nodes, 25 edges) is a smaller network which suffers from a lack of connectivity between the eastern and western regions. A standard 8x8 Mesh network contains 64 nodes and 112 edges. The 8x8 Mesh network has a high degree of connectivity.

The results of the NSF network simulations are displayed in Figure 7.14. The difference in blocking between the algorithms is minimal. The advantages of DP-RWA and IA-BF are muted by the lack of connectivity. In less connected networks, the RWA algorithm choice is less important. DP-RWA has the lowest blocking probability, followed by IA-BF, AQoS, and PABR.

Figure 7.15: Overall Blocking - 8x8 Mesh Network

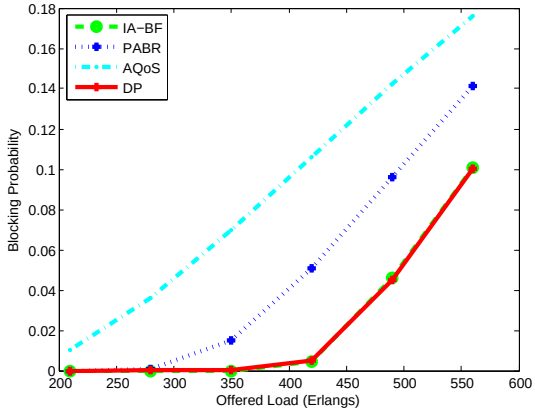
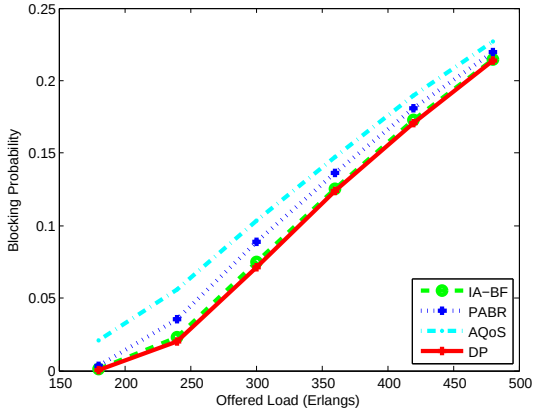


Figure 7.15 shows the results of the 8x8 Mesh simulations. Once again, DP-RWA and IA-BF are the top performing algorithms. For this network topology, IA-BF has the lowest blocking probability followed closely by DP-RWA. There is a very large difference between PABR and AQoS. DP-RWA blocks 50.5% fewer connections than PABR and 72.2% fewer connections than AQoS. In the highly connected 8x8 Mesh network, the RWA choice is very significant.

The performance of QM and AQoS are very dependent upon the network topology. It was expected that AQoS would offer significantly lower blocking probability than QM. This was the case for the highly connected 8x8 Mesh network, but QM has the lower blocking probability for both the NSF and UC Davis Mesh networks.

LORA and PABR are also dependent upon the network topology. These algorithms depend on a parameter β used to control the bias toward choosing longer, but less utilized, paths. The optimal values of β are not obvious, and network operators will need to tune the variable for optimal performance considering the network topology and current traffic load.

Figure 7.16: Overall Blocking - ASE Dominant



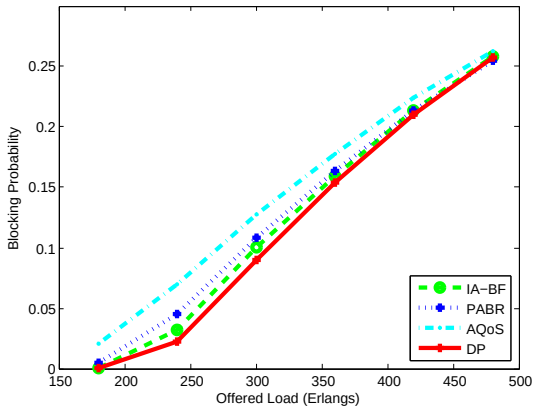
The dependence of QM, AQoS, LORA, and PABR on the network topology is a concern. IA-BF and DP-RWA are adaptable and perform well across a wide variety of simulations.

7.6 Physical Impairments

It is also important to consider scenarios where both linear and nonlinear impairments dominate. In the previous simulations, the impairments were balanced so that neither dominated. The results of a simulation where ASE noise is a dominant factor are presented in Figure 7.16. In this scenario, ASE is roughly 3 to 4 times greater than the sum of FWM and XPM, with some variance among the different algorithms.

DP-RWA is the best performing algorithm. DP-RWA outperforms IA-BF, PABR, and AQoS by 2.0%, 2.4%, and 10.7% respectively with regard to blocking probability. The differences are smaller than in the balanced scenario. As the linear impairments become more significant, the RWA problem becomes more like the traditional routing problem and the RWA choice becomes less important.

Figure 7.17: Overall Blocking - Nonlinear Dominant



The nonlinear dominant results are presented in Figure 7.17. In this scenario, the total nonlinear effects (XPM and FWM) are roughly 2.5 to 3.5 times greater than ASE, with some variance between the algorithms. Using realistic parameters, it becomes difficult to increase of the nonlinear effects beyond the levels discussed here.

DP-RWA is again the top performing algorithm. DP-RWA outperforms IA-BF, PABR, and AQoS by 4.0%, 7.7%, and 20.3% respectively. The differences between the algorithms are larger than in the ASE dominant and balanced scenarios. This supports our conclusion that as nonlinear impairments increase, the RWA selection is more important. In fact, the difference between DP-RWA and IA-BF has doubled.

7.7 Wavelength Assignment Algorithms

Most of the research in RWA has focused on the Routing Algorithm (RA), relying on standard Wavelength Assignment (WA) algorithms such as First Fit (FF) and Random Pick (RP). A general consensus is that FF offers the tightest wavelength packing and thus lowest blocking probability, while Random Pick offers the highest signal quality.

Figure 7.18: Overall Blocking - FF vs RP

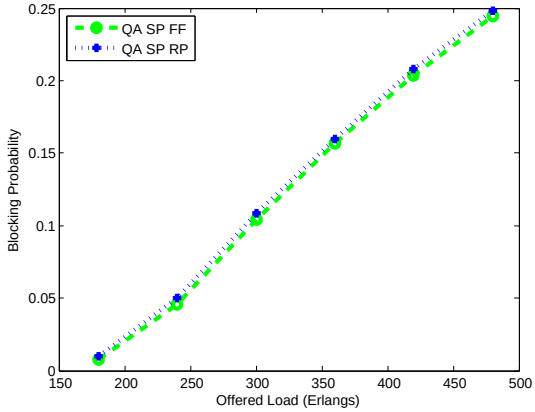
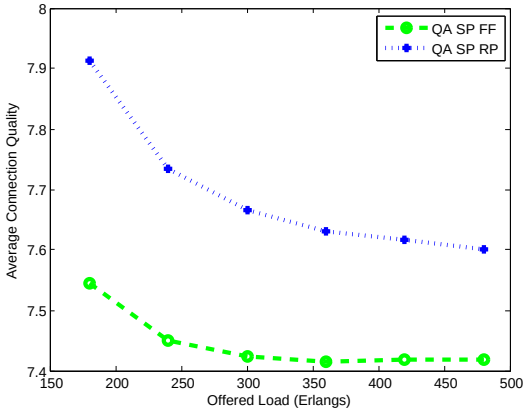


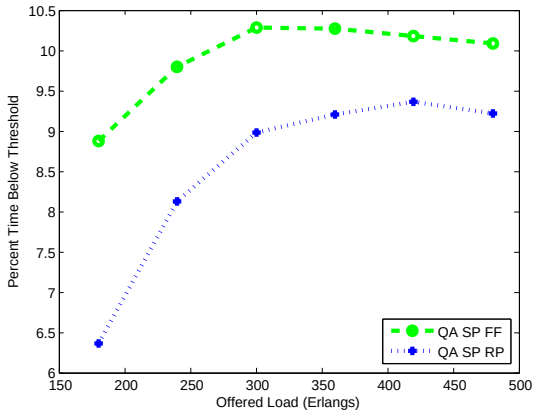
Figure 7.19: Average Connection Quality - FF vs RP



To evaluate the performance of FF and RP, we ran simulations using the two WAs while using QA SP as the RA. The blocking probability is shown in Figure 7.18. FF offers the lowest overall blocking, as expected. However, the margin is small as FF blocks 2.9% fewer connections than RP.

RP offers better signal quality than FF. Figure 7.19 shows that RP has a 3.2% higher average signal quality. RP connections also spend less time below the quality threshold, as shown in Figure 7.20. This is a consequence of FF accepting more connections.

Figure 7.20: Percent Time Below Threshold - FF vs RP



Simulations from RAPTOR support the previous research that FF offers the lowest blocking probability and RP offers the highest signal quality. The differences between the two algorithms is minimal, the RA selection is much more important than the WA selection.

7.8 Wavelength Division Multiplexing

Future optical networks will increase the number of wavelengths per fiber. DP-RWA, LORA, and PABR's complexity does not increase with the number of wavelengths, but QM, AQoS, IA-BF, IA-FF all have a complexity that grows with the number of wavelengths.

Simulations were run where the number of wavelengths per fiber and traffic load were varied. The wavelength count and traffic load were always increased by the same rate, so the traffic load per wavelength was unchanged.

The overall blocking probability of QA SP is shown in Figure 7.21. Interestingly, the blocking probability declines even though the traffic load per wavelength is held

Figure 7.21: Overall Blocking - Various Number of Wavelengths

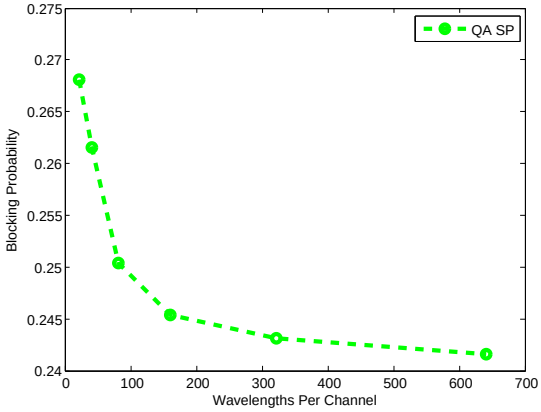
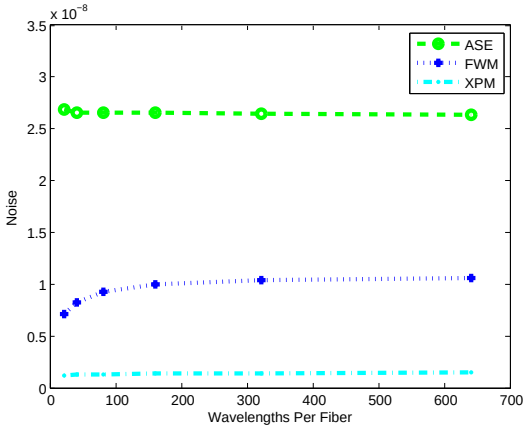


Figure 7.22: Noise - Various Number of Wavelengths



constant. This is due to the WA algorithm having more options to select from, allowing an increasing rate of connection acceptance.

The impairments from this simulation are displayed in Figure 7.22. The nonlinear impairments grow slightly as the wavelengths per fiber increase. This may be an unexpected result to some. Our investigation has shown when considering the noise on a specific channel, only the nearest neighbors of the channel contribute meaningful noise. Thus adding new channels does not directly increase the noise.

The reason for the small increase in noise is that the channels with the lowest and highest frequencies do not have as many neighboring channels contributing noise.

Consider, for example, the channel with the lowest frequency. There are no channels with a lower frequency than it, so only channels with a higher frequency can contribute to the noise. This contrasts to a frequency in the middle of the range, where both lower and higher frequencies are active. As the number of channels per wavelength increase, the percentage of channels with a full window (and thus more nonlinear impairments) increases.

7.9 Hot Spots

RAPTOR's GUI has proven a valuable tool in detecting hot spots in the network. Hot spots typically occur in the interior links of the network. These links have to carry local traffic, plus the traffic from one region of the graph to another. They serve as a bottleneck in the network, limiting the number of connections that can be setup.

RAPTOR currently supports static configurations of a network. In other words, once the topology is set, the resources can not change. It could be extended to support network faults, where a resource is temporarily unavailable. This might occur for a planned maintenance upgrade, a power outage, or an accidental cutting of a cable.

Hot spots could also be created due to unbalanced traffic demands. Currently, each node adds the same level of traffic. Traffic demands will naturally variate for many reasons, such as the time of day, natural disasters, and local events. The addition of unbalanced traffic loads would be valuable addition to RAPTOR.

It would be interesting to determine which algorithms are able to adapt to dynamic network topologies and unbalanced traffic demands.

7.10 Conclusion

RAPTOR is a useful tool for evaluating transparent optical networks and freely-available to the academic community for non-commercial use. Various network topologies and parameters, larger number of wavelengths, and alternative traffic patterns can be easily simulated with RAPTOR. This chapter provided valuable insight into the performance of a variety of RWA algorithms for all-optical networks.

Physically aware RWA algorithms offer major improvements to signal quality with a minor impact to blocking probability. In most scenarios, DP-RWA is the best performing RWA algorithm. As next generation optical networks move toward the goal of transparency and higher degrees of Wavelength Division Multiplexing, nonlinear impairments will increase. These changes will make the development of new and more sophisticated RWA algorithms very important.

CHAPTER 8
CONCLUSION

Optical networks form the foundation of today's information infrastructure. Current generation optical networks consist largely of point-to-point electronically transmitted links which switch between nodes and repeaters. There is a trend in optical networking to move from the current generation opaque networks toward transparent networks. Transparent networks use only optical devices, eliminating the costly need for OEO conversions. Unfortunately, transparent networks present a unique challenge in maintaining acceptable signal quality levels. The goal of this research is to expand the current understanding of RWA performance in transparent optical networks.

Our research achieved this goal in three ways. First, we developed RAPTOR, a custom built optical network simulator. Using RAPTOR, we performed an extensive analysis of existing RWA algorithms. Last, we presented our new dynamic programming based RWA algorithm with promising results.

RAPTOR enables us to perform an in-depth study of transparent optical networks in a unique and realistic manner. None of the existing network simulation tools come close to modeling transparent optical networks like RAPTOR. RAPTOR contains physically aware modules, allowing us to accurately estimate signal quality. RAPTOR is fast and multi-threaded, allowing us to study large optical networks. RAPTOR is an integral part of the research presented in this dissertation.

Using RAPTOR, we conducted an extensive performance analysis of existing RWA algorithms. We explored many different traffic models, traffic loads, signal quality, and network topologies in a comprehensive fashion. We showed the importance of using physically aware RWA algorithms. We directly compared the leading RWA algorithms in a manner has not been done before.

We presented Dynamic Programming - RWA, or DP-RWA, a new physically aware RWA algorithm. DP-RWA has the best overall performance in most scenarios. It is flexible and adapts well to all network conditions we studied. It scales very well to large optical networks. It shows good promise for future optical networks.

We discussed Ant Colony Optimization and its applicability to the RWA problem. We created ACO-RWA and $\mathcal{MM}$-ACO-RWA, two Ant Colony Optimization based RWA algorithms. The results were mixed, although ACO-RWA and $\mathcal{MM}$-ACO-RWA might be useful in some applications.

8.1 Future Work

While RAPTOR has proved to be a very useful tool, there are a couple areas in which it could be improved. RAPTOR studies three of the dominant physical impairments, it would be helpful to consider other nonlinear impairments, such as Self Phase Modulation. The calculation of ASE noise could be expanded to a cascading formula, allowing for a heterogeneous mix of amplifiers with different noise figures and gains.

RAPTOR simulates only Poisson process based traffic models, which work well for circuit switched networks. It would be beneficial to consider Pareto distributions as well, especially since research has shown they are effective in modeling packet switched networks. While today's optical networks are circuit switched, there is much research into Optical Burst Switching (OBS) and Optical Packet Switching (OPS). A Pareto distribution might model OBS and OPS more accurately.

RAPTOR's GUI has shown a tremendous capability to visualize optical networks, helping us to perform an extensive analysis. It would be beneficial to add additional

functionality to the GUI. The user interface is a little awkward and unstable at times, so it would benefit from a little fine tuning.

Finally, RAPTOR would benefit from the addition of hot spots. Hot spots could be added in two ways. First, there could be a network resource that goes offline temporarily. Another method of creating hot spots would be to use a non uniform traffic generation pattern. It would be interesting to determine how the algorithms adapt to hot spots.

REFERENCES CITED

[1] Wikipedia, the free ecyclopedia, 2009.

[2] B. Mukherjee. *Optical WDM Networks*. Springer Science and Business Media, 2006.

[3] Single forward pumping EDFA. Technical Report, Samsung Electronics, 1997.

[4] W. Linn. *Physically Aware Agile Optical Networks*. Doctoral Dissertation, Montana State University, July 2008.

[5] F. B. Shepherd, A. Vetta. Lighting Fibers in a Dark Network. *IEEE Journal on Selected Areas in Communications*, 22(9):1583–1588, November 2004.

[6] F. Lederer M. Golles I.M. Uzunov. *Physics Letter A*, 231(195), 1997.

[7] M.E. Fermann A. Galvanauska M.L. Stock K.K. Wong D. Harter L. Goldberg. *Optical Letters*, 24(1428), 1999.

[8] T. Hori N. Nishizawa H. Nagai M. Yoshida T. Goto. *IEEE Photon Technologies Letters*, 13(13), 2001.

[9] J. Berthold A. Saleh L. Balir, Jane Simmons. Optical Networking: Past, Present, and Future. *Journal of Lightwave Technology*, 26(9):1104–1117, May 2008.

[10] P. S. Henry. Lightwave Primer. *IEEE Journal of Quantum Electronics*, QE-21:1862–1879, December 1985.

[11] M. Azizoglu. *Phase Noise in Coherent Optical Communications*. Doctoral Dissertation, Massachusetts Institute of Technology, 1991.

[12] T. Jackson M. Lee T. Hahn W. Lin R. Wolff B. Mumey K. Repasky. EDFA Transient Reduction Using Power Shaping. *IASTED WOC 2008*, pages 208 – 212, 2008.

[13] M. Yano F. Yamagishi T. Tsuda. Optical MEMS for Photonic Switching - Compact and Stable Optical Crossconnect Switches for Simple, Fast, and Flexible Wavelength Applcations in Recent Photonic Networks. *IEEE Journal of Selected Topics in Quantum Electronics*, 11(2):383–394, March/April 2005.

[14] A. R. Chraplyvy. Limits on Lightwave Communications Imposed By Optical-Fiber Nonlinearities. *IEEE/OSA Journal of Lightwave Technology*, 8:1548–1557, October 1990.

[15] A. R. Chraplyvy D. Marcuse F. Forghieri R. W. Tkach. Reduction of four-wave mixing crosstalk in WDM systems using unequally spaced channels. *IEEE Photonics Technology Letters*, 6(6):754–756, 1994.

[16] G. P. Agrawal. *Nonlinear Fiber Optics*. Academic Press, 2007.

[17] G. P. Agrawal. *Lightwave Technology: Telecommunication Systems*. Wiley, 2005.

[18] A. R. Chraplyvy. Optical Power Limits in Multi-Channel Wavelength-Division-Multiplexed Systems Due to Stimulated Raman Scattering. *Electronics Letters*, 20(2):58–59, 1984.

[19] G. P. Agrawal. *Nonlinear Fiber Optics*. Academic Press, 2001.

[20] T. Deng, S. Subramaniam. Adaptive QoS Routing in Dynamic Wavelength Routed Optical Networks. *Broadband Networks*, pages 184–193, 2005.

[21] G. P. Agrawal. *Fiber Optic Communication Systems*. Wiley-Interscience, 2002.

[22] J. He M. Brandt-Pearce S. Subramaniam. QoT-Aware Routing in Impairment-Constrained Optical Networks. *Proceedings from IEEE Globecom*, pages 2269–2274, 2007.

[23] H. Zang J. Jue, B. Mukerjee. A Review of Routing and Wavelength Assignment Approaches for Wavelength Routed Optical WDM Networks. *Optical Networks Magaizne*, January 2000.

[24] C. Meyerb B. Jaumarda, B. Thiongane. Comparison of ILP formulations for the RWA problem. *Optical Switching and Networking*, 10:157 – 172, November 2007.

[25] S. Evan A. Itai, A. Shamir. On the Complexity of Timetable and Multicommodity Flow Problems. *SIAM Journal of Computing*, 5:691–703, 1976.

[26] R. M. Karp. *Complexity of Computer Computations*. Plenum Press, 1972.

[27] I. Chlamtac A. Ganz, G. Karmi. Lightpath Communctions: an approach to high bandwidth optical WAN's. *IEEE Transactions on Communications*, 40(7):1171–1182, July 1992.

[28] E. Martins, M. Pascoal. A New Implementation of Yen's Ranking Loopless Paths Algorithm. *Quarterly Journal of the Begian, French, and Italian Operations Research Societies*, 2009.

[29] S. Ramamurthy. *Optical Design of WDM Network Architectures*. Doctoral Dissertation, University of California - Davis, 1998.

[30] W. Lin, R. Wolff. A Lexicographically Optimized Routing Algorithm for All-Optical Networks. *Optical Communication System and Networks*, 2007.

[31] W. Lin R. Wolff, T. Hahn. A Distributed Impairment Aware Framework for All-Optical Networks. *Optical Switching and Networking*, 2009.

[32] Y. Huang J. Heritage, B. Mukherjee. Connection Provisioning With Transmission Impairment Consideration in Optical WDM Networks with High-Speed Channels. *Journal of Lightwave Technology*, 23(3), March 2005.

[33] S. Ramamurthy, B. Mukherjee. Survivable WDM mesh networks, part I - protection. *Proceedings of IEEE INFOCOM*, pages 744–751, March 1999.

[34] R. Barry, S. Subramaniam. The MAX-SUM Wavelength Assignment for WDM Ring Networks. *Proceedings of Optical Fiber Conference*, pages 121 – 122, February 1997.

[35] X. Zhang, C. Qiao. Wavelength Assignment for Dynamic Traffic in Multi-Fiber WDM Networks. *Proceeding of International Conference on Communications*, Volume 1, pages 406–410, June 1997.

[36] A. McGregor, B. Shepherd. Island hopping and path colouring with applications to WDM network design. *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms*, pages 864–873, 2007.

[37] H. Hua R. Shankar M. Florjanczyk T. Hall A. Vukovic. Multi-degree ROADM based on wavelength selective switches: Architectures and scalability. *Optical Communications*, 279(1):94–100, 2007.

[38] H. Feng J.P. Heritage B. Ramamurthy, D. Datta, B. Mukherjee. SIMON: A Simulator for Optical Networks. *Conference on All-Optical Networking*, pages 130 – 134, 1999.

[39] S. Pachnicke J. Reichert S. Spaelter E. Voges. Fast analytical assessment of the signal quality in transparent optical networks. *Journal of Lightwave Technology*, 24:815 – 824, 2006.

[40] T. Hahn. An Efficient, Dynamic Algorithm for Routing In Transparent Physically Aware Optical Networks. *IEEE Globecom*, 2010.

[41] R. Rivest C. Stein T. Cormen, C. Leiserson. *Introduction to Algorithms*. MIT Press, 2009.

[42] M. Dorigo. *Learning and Natural Algorithms*. Doctoral Dissertation, Politecnico di Milano, Italie, 1992.

[43] M. Dorigo, T. Stuetzle. *Ant Colony Optimization*. The MIT Press, 2004.

[44] S. Varela, M. Sinclair. Ant Colony Optimization for Virtual Wavelength Path Routing and Wavelength Allocation. *Proceedings of the Congress on Evolutionary Computation*, 1999.

[45] B. Garlick, R. Barr. Dynamic Wavelength Routing in WDM Networks via Ant Colony Optimization. *Lecture Notes in Computer Science*, pages 250 – 255, 2002.

[46] W. Gutjahr. On the finite-time- dynamics of ant colony optimization. *Methodology and Computing in Applied Probability*, 8:105 – 133, 2006.

[47] T. Stuetzle et H.H. Hoos. MAX MIN Ant System. *Future Generation Computer Systems*, 16:889–914, 2000.

[48] W. Gutjahr. Mathematical Runtime Analysis of ACO Algorithms: Survey of an Emerging Issue. *Swarm Intelligence*, pages 59 – 79, 2007.

[49] T. Hahn B. Mumey R. Wolff. Performance Evaluation of Physically Aware RWA Strategies for Transparent Optical Networks. *IEEE Globecom*, 2010.

[50] T. Hahn B. Mumey R. Wolff. Performance Evaluation of Physically Aware RWA Strategies for Transparent Optical Networks. *ACM Transactions on Networking*, 2010.

[51] C. Qiao, M. Yoo. Optical burst switching (OBS) - a new paradigm for an Optical Internet. *Journal of High Speed Networks*, 8:69 – 84, 1999.

[52] G. Rouskas, L. Xu. *Emerging Optical Network Technologies*. Springer US, 2004.