ABSTRACT

PARTIAL DESTINATION RESOLUTION IN MULTICAST ELASTIC OPTICAL NETWORKS: A MIXED-INTEGER LINEAR PROGRAMMING APPROACH

by Andrew J. Rush

This paper explores the spectral assignment problem and proof of value of partial destination resolution (PDR) in multicast elastic optical networks using a supplied tree approach. The partial destination resolution method allows for subsets of destination nodes in multicast calls to be connected with lower than optimal bandwidth requirements. This method allows additional connections to be tailored around this flexibility, resulting in an increased amount of total throughput of the system at the expense of a subset of the nodes receiving lower bandwidth connections. The study is performed by adding PDR capabilities to existing elastic optical networking systems and performing a comparison study of the effects on percent demand throughput for a variety of network topologies and spectral slice and demand sizes. Limitations of the implementation of integer linear programming techniques with respect to runtime and memory usage are examined to determine the bounding of system sizes.
PARTIAL DESTINATION RESOLUTION IN MULTICAST ELASTIC OPTICAL NETWORKS: A MIXED-INTEGER LINEAR PROGRAMMING APPROACH

A Thesis

Submitted to the

Faculty of Miami University

in partial fulfillment of

the requirements for the degree of

Master of Science in

Computational Science and Engineering

by

Andrew J. Rush

Miami University

Oxford, Ohio

2016

Advisor________________________

Dr. Gokhan Sahin

Co-Advisor_______________________

Dr. Donald Ucci

Reader_________________________

Dr. Chi-Hao Cheng
Chapter 1 - Contents

List of Figures .................................................................................................................. v
Chapter 1 - Introduction: ............................................................................................... 1
Chapter 2 - Literature Review: ...................................................................................... 5
   Optical Networking Background: ............................................................................. 5
   State of the Art: ........................................................................................................... 7
      Elastic Optical Networking Background: .............................................................. 9
      OFDM in EONs: ...................................................................................................... 9
      Modulation Techniques in OFDM EONs: .............................................................. 10
      Modulation and Wavelength Conversion: .......................................................... 11
      Multicasting in Optical Networks: ...................................................................... 13
      Multicast Elastic Data Streams: .......................................................................... 15
Chapter 3 - Formulation Setup: ..................................................................................... 18
   Proposal Problem Statement: .................................................................................. 18
   Formulation Subsections: ......................................................................................... 18
Chapter 4 - Basic ILP/MILP Structure: ......................................................................... 20
   Center Frequency Slice and Radius vs Start Frequency Slice and Range: .................. 20
   Elastic Data Representation: .................................................................................... 21
   Downstream Continuity: ........................................................................................... 22
   Multi-Tier MILP Structure: ....................................................................................... 22
   Objective Function Setup: ......................................................................................... 23
   Core Constraint Setup: ............................................................................................. 24
   Computational Complexity: ...................................................................................... 24
Chapter 5 - Formulation: ............................................................................................... 25
   Initial Formulation: .................................................................................................... 25
      Problem Statement: ............................................................................................... 25
      Assumptions: .......................................................................................................... 25
      Constant Definitions: ............................................................................................. 26
      Variable Definitions: .............................................................................................. 26
      Objective Function: .............................................................................................. 27
      Connectivity Constraints: ..................................................................................... 27
      Continuity Constraints: ......................................................................................... 28
   Initial Formulation Discussion: ................................................................................... 29
      Initial Problems: .................................................................................................... 29
   Final Formulation: ...................................................................................................... 31
      Problem Statement: ............................................................................................... 31
      Assumptions (same as initial formulation): .......................................................... 31
      Constant Definitions: ............................................................................................ 32
      Variable Definitions: ............................................................................................. 32
      Range Definitions: .................................................................................................. 32
      Objective Function: .............................................................................................. 33
      Overlap Constraint: .............................................................................................. 33
Chapter 8 - Future Work ................................................................. 58
Chapter 9 - Conclusions: .............................................................. 60
Appendix: ....................................................................................... 62
References: ..................................................................................... 70
List of Figures

Figure 6-1: 4 edges, 5 demands, 5 slices Percent Demand Throughput vs Number of Frequency Slices Demanded ................................................................. 36

Figure 6-2: 4 edges, 5 demands, 5 slices Percent Demand Throughput vs Number of Frequency Slices Demanded ................................................................. 36

Figure 7-1: 6 edge, 6 demand Percent Demand Throughput vs Number of Available Frequency Slices ................................................................. 39

Figure 7-2: 6 Node Ring Topology Graph ................................................................. 41

Figure 7-3: 6 Node Ring Topology Percent Demand Throughput vs Available Frequency Slices ................................................................. 42

Figure 7-4: 9 Node Mesh Topology Graph ................................................................. 43

Figure 7-5: 9 Node Mesh Topology Percent Demand Throughput vs Number of Available Frequency Slices ................................................................. 44

Figure 7-6: 7 Node Binary Tree Topology Graph ................................................................. 45

Figure 7-7: 7 Node Binary Tree Topology Percent Demand Throughput vs Number of Available Frequency Slices ................................................................. 46

Figure 7-8: Reliability Test System 2 Percent Demand Throughput vs PDR .............................. 50

Figure 7-9: Reliability Test System 3 Percent Demand Throughput vs PDR .............................. 52

Figure 7-10: Reliability Test System 4 Percent Demand Throughput vs PDR .............................. 54

Figure 7-11: Reliability Test System 5 Percent Demand Throughput vs PDR .............................. 56
Chapter 1 - Introduction:

In recent years, the amount of network traffic consisting of demands from a source to many sources has increased radically. These type of connections exist whenever and wherever a source of information needs to be distributed to a large number of destinations simultaneously. This network demand can be accommodated most effectively with the added capabilities of handling multicast data connections, thereby building a tree of used edges in a network, and distributing the information to all leaf nodes in the system [1]. If we compound this with the recent growth of industries such as GOOGLE™ fiber, as well as the influx of research on Elastic Optical Networks (EONs) [1,3], then we see that this creates a problem space that is currently in great demand. While research in this area is currently being approached from a variety of areas, adapting the formulations with respect to allowing subsets of frequency bands to reach some destinations while retaining full connections to other destinations has not yet been published.

This report studies the implementation and proof of benefit of adding Partial Destination Resolution (PDR) to EONs using a candidate tree approach. The problem is currently defined as a static problem with all connections attempting to connect at the same time. This is solved as an all-at-once problem, requiring use of optimization software to find the optimal system state. This thesis makes use of Mixed Integer Linear Programming (MILP) formulations to find the optimal answers for each system configuration and set of demands. All MILP formulations are performed using IBM's ILOG CPLEX Optimization Studio and Optimization Programming Language (OPL) [2]. The result of this work is a study of proof of value of PDR and a comparison study of existing systems before and after PDR capabilities are added. This information will allow for conclusions to be drawn for some areas in EONs. PDR has a great impact and can contribute to networking demands.

We note that an EON is a specific type of optical networks that allows for allocation of spectral resources to be determined on a connection-by-connection basis. This elastic optical system differs from traditional optical networks by removing the predetermined "channels" through which
data is transferred. Instead, EONs allow for network resources, including upper and lower bounds of the frequency bands utilized, to be chosen at the time of connection and allocated across the entire network. This poses obvious problems as the computational complexity of establishing connections now has an added component. This complexity compounds upon itself with many different network protocols such as verification of connections and determination of "best fit" new connection routing. This causes the Routing and Spectral Assignment problem (RSA) to be significantly more complex and demanding of network resources when compared to WDM methods. When formulated as an MILP, the decision variables for frequency slices are directly related to this elasticity constraint, causing a portion of the runtime of the algorithm to be dependent upon this added complexity.

PDR is a concept used in a variety of multicast networking systems that allows for connections to be established to a subset of destination nodes at a lower than optimal bandwidth permitting sets of traffic demands to be better applied to network resources. This procedure provides flexibility to allocate resources in a suboptimal way for individual network demands, while allowing for the maximization of key performance metrics of the set of demands. As this problem is defined for multicast networks, different subsections of the destination nodes of the multicast tree may then be allowed to have different percentages of total bandwidth received for a connection.

This formulation uses a supplied tree approach, as the computational complexity of the RSA problem is a significant issue in itself. In some multicast tree determination algorithms for EONs, the problem is directly related to the vertex cover problem of computation theory [6]. This reduction is shown proving the problem to be NP-complete. This issue is avoided by providing a set tree for each connection for our formulation. For this thesis, this will be extended to be a candidate tree problem, allowing a specific tree to be chosen from a set of supplied trees.

The real-world added benefit of this procedure can be seen in a variety of forms. For example, distribution of live video feeds from Internet-based television can be visualized as a data stream from one source to many destinations. The core information of the broadcast would be stored near the center frequency of the transmission, while supplemental data, used for higher resolution of
sound and video would be stored towards the outer edge of the band range. By shrinking down the radius of the signal for PDR, destinations would still receive the majority of the information allowing for the broadcast to continue in an uninterrupted manner but at a lower quality. The rest of the multicast call could receive full video quality at no expense from the lowering of quality to the subset of destinations, which require throttling. This concept can further be adapted to areas such as online teleconferencing software or potentially network state updating software. This is of significance due to the explosive nature of demand growth with respect to the technology [11, 12].

In previous work, research has been conducted on many surrounding areas related to this topic. Extensive work has been done in traditional optical networks, ranging from addition of length adaptive capabilities to modulation and frequency conversion and light splitter placement.

Many works have been published in recent years in the area of EONs [1,3,5,22]. While the field itself is being readily explored, many different problems diversify the concentration of information. Many problems can be directly related to non-elastic or non-optical equivalent problems, yet pose very different difficulties when adapted to EONs. This can be attributed to a variety of effects of optical transmissions, including the speed of optical carrier waves and the physical limitations of processing speeds with respect to the transmission times. While this paper does not directly examine the physical constraints of the formulation, the work being provided could be extended to feasibility testing and network updating methods for future work.

Work has been done in the area of multicast EON formulation with respect to distance adaptive transmissions [1]. This foundation of modeling multicast traffic flow in EONs is a vital background piece upon which this thesis is built. The sited paper is focused around the distance adaptive approach to increase the value of the system by changing the modulation formats to fit different demands in an optimal way. The question itself is very different from the focus of this thesis; however, the background formulation upon which the elastic multicast data is modelled is directly related and used as a primary reference. Traffic data, the candidate tree approach, and some of the network constraints are closely related to this work.
The portion of the field that is not previously explored for this paper is adapting the concept of PDR to these multicast EONs. PDR is a concept derived from partial destination blocking and other adaptive throughput blocking methods implemented in other networking architectures. This has not been adapted to the elastic capabilities of EONs, making it a key contribution to the field. The elastic capability of EONs allows connections to be established so that less space is wasted between the frequency bands utilized across many different connections. This can be further extended by allowing capabilities to decrease supplied bandwidth to certain destinations within multicast calls and potentially allow additional partial connections to make use of the wasted space between the non-PDR established connections.

In this work, the background of the subject is explored leading to the current state of the art. Key research in modern networking and related fields is investigated with intention of setting background for which this research is based. The thesis includes key concepts of the current system, the entirety of the created PDR in multicast EONs algorithm used for the MILP, and the comparison study used to verify the benefits of the algorithm.
Chapter 2 - Literature Review:

Optical Networking Background:

Optical networking as a form of communication networks though hardline connections began in the 1980’s [8]. With the addition of Wavelength Division Multiplexing (WDM) and optical amplifiers, the basis of electrical communication equivalencies started to take shape. Optical networks offer very significant improvement upon other non-light based communication systems in areas including [8]:

- **Wider Bandwidth**: carrier frequency ranges are in the order of $10^{13}$ to $10^{15}$ Hz compared to the order of radio waves $10^6$ Hz and microwaves $10^{10}$ Hz. This increased range is directly related to the amount of frequency space allowed for channels to be allocated on optical networks. Increasing the available channel space allows for more connections to be established therefore increasing throughput of the system.

- **Small Size and Weight**: Optical cables are smaller and lighter when compared to similar copper based cables used for electrical networks.

- **Low Transmission Loss**: Optical signals are much lower in loss when compared to electric signals. Although light signals do experience a level of loss and degradation with respect to chromatic dispersion and reflection and refraction [27], this is at a much lower level than losses when compared to their electrical equivalent. With the advancement of physical layer knowledge of optical networking, signal degradation has been steadily decreasing with higher quality materials. This lower level of loss allows system designer to use less regenerators/repeaters in the signal channels.

Originally, optical networks were designed with single signal point-to-point operations in mind, allowing a source node to connect to exactly one destination node through a specific channel. This is the simplest form of optical communication using fiber. The signal would be sent from a source, transmitted in a method to reduce the losses and with lower noise, and then received at the destination node. The signal would then be interpreted and utilized.
Soon, adaptations were made upon the point-to-point operations using methods created for electric signals, such as modulation techniques and multiplexing. Physical constraints occurring due to physical devices were quickly overcome, allowing multiple signals to be combined and transmitted to a single destination. Over the last 20 years, this has grown and developed to be nearly equivalent in capabilities of other networking types including electrical communication systems.

One major difficulty with the transmission of network signals lies in the knowledge, or lack thereof, of global system state at any given time. As the system state relies upon communicating connections between different nodes, there is a lag associated with the known system state and the actual system state at a specific node. Therefore, network resources must be utilized to communicate and update each of the nodes to keep the global knowledge of the system as current as possible. In optical signals, this transmission time is very fast, yet the computational time to process the updates and retransmit is very slow, as it is an electrically based calculation compared to the actual transmission time that occurs at the speed of light.

Another major difficulty is establishing connections without having to modify wavelength at an intermediate point. In electrical signals, a penalty in certain networking algorithms allows a transmission to change wavelengths as a carrier in the middle of a transmission. This concept can be visualized as allowing an intermediate node to read a message and store it, then retransmit it using a different carrier wave. This concept allows connections to use different carrier waves at different points in a network to work around existing workloads. In optical networks, this is not nearly as feasible since the time that it would take to process the signal is a major limiting factor in the system. Optical workarounds would need to be implemented to “buffer” the signal and then change the carrier wave without having to process the signal itself through a computer.
State of the Art:

Wavelength Division Multiplexing-Based Optical Networks:

WDM on optical networks offers some major advantages. WDM allows for stacking many signals in the same channel, giving a high capacity for transmissions. WDM also has a lot of flexibility, allowing for reconfigurable wavelength switching [5]. Initially, this method of combining signals appeared to be the best option. Now, with data rates reaching 1 Tb/s, WDM is poses a major problem with Signal-to-Noise Ratio (SNR) when combined with medium-range to long-range traffic. Since the amount of noise with respect to the signal amplitudes is increasing, regeneration is needed more often to send over the same amount of space. In order to combat this, there needs to be more space in a given channel, which significantly reduces the number of available channels. Multiple methods have been considered to combat this effect, including changing modulation techniques to providing cheaper regeneration methods. Distance adaptive algorithms have become more relevant, as this would allow a signal to choose a different modulation scheme for different distances of transmissions, thereby allowing the algorithm to tailor calls to keep the SNR within a usable range and still use a reasonable channel size.

A key progressive feature of this research is the combative nature of the growing rate of transmissions speeds and the SNR of transmitted signals. In order to keep the SNR in a state that allows for low bit error rate at the receiving end, the signal needs to be increased in amplitude with respect to the noise. This eventually causes issues with the prevalence of physical impairments to grow with respect to transmission speed [5]. On a network layer, this can be combatted by relaxing the physical constraints placed upon the system such as removing the predetermined boundaries associated with the channels of WDM optical networks. This can be seen as one of the inspirational points which led to development of EONs, as the predetermined range of frequency bands is not set for EONs. Instead, the range can be tailored to fit each signal. Therefore, the system can create a channel of a specific value such that the SNR at the receiving end is delivered to fit the bit error rate required by the connection.
Another option that has been explored to combat this issue is to keep the rigid setup of non-EONs but change the modulation speed. When compared to EONs, this has an advantage in the computational and communication requirements of having preallocated the frequency slots for each channel. If a different modulation technique could be used in this networking form to meet higher demands, then the system would have a better capacity to utilize a wide range of calls while not increasing the computational requirements. This does come at a cost though, as increasing the SNR of higher speed signals will have the same problems as WDM eventually. This may be a short-term fix but, eventually, signal speeds would increase to the point that the same problem arises in each of our modern day modulation schemes.
Elastic Optical Networking Background:

Early work in EONs explored the value of the physical limitations removed by their “elastic” portion, thereby creating a new area of fundamental research for this new adaptation. Without the requirement to fit demands into preallocated spectral slots, connections could be established in a manner that allows for accommodation of large connections, which would fit few or none of the predefined slots, and small connections that would not utilize much of the frequency band range allocated to each of the slots. This demand range necessary for a specific signal is determined by many factors, including the duration of a call, the modulation scheme, and the physical characteristics of multiplexers, transmitters, and receivers. It can clearly be seen that this new adaption to elastic optical networks led to a vast variety of research areas filling all of these needs and more.

OFDM in EONs:

The next type of signal combination scheme examined for modern optical communications is Orthogonal Frequency Division Multiplexing (OFDM) [28]. OFDM allows for multiple data rate sub-wavelength or super-wavelength paths to be formed. This allows multiple signals to be combined in a way that they are distinguishable at the receiving end. This data-rate can be set by transponders that give just enough subcarriers to accommodate each signal demand in a method known as signal slicing [14]. The traffic is then aggregated to form super-channels. This creates the defining feature of EONs, which is the ability to set frequency upper and lower ranges dynamically, giving flexibility in granularity and data rate.

Zhang, et. al, published a survey of OFDM-Based Elastic Core Optical Networking examining the state-of-the-art up to 2013 for implementing the specific modulation scheme OFDM to fit into EONs [5]. This gave great insight into the state-of-the-field and the performance metrics being examined throughout. OFDM is a prime candidate for the multiplexing scheme of choice for many different uses of EONs. The immense flexibility and scalability in spectrum allocation and data rate accommodation allows for EONs to support a wide variety of traffic demands [5]. Unfortunately, as data rate demands grow, so does the required transmission speed, causing the
physical space and bottleneck limitations to become more severe [13]. This is a great driving force behind the implementation of methods such as PDR to EONs.

To examine the effects of OFDM in EONs, a comparison study would need to be done for the specific network system established. Research has been performed in numerous areas surrounding OFDM in EONs and the elasticity created, fueling many publications done up to current day including innovative research and some survey papers [5].

It is noteworthy to discuss the network knowledge and traffic grooming problems that arise with this OFDM in EON method. It has been determined that the precise spectrum utilization improvement depends significantly on the traffic patterns and network topology [15]. As the major change between EONs and fixed channel optical networks is the definition of start and end values for channels, this poses a major challenge with the global network awareness of the system state. In fixed channel optical networks, all nodes have global knowledge of the channel limits at all times, as they are fixed. In EONs these channel definitions change with respect to time, which means that a portion of the networks capabilities must be utilized as an updating mechanism to keep each of the nodes’ global knowledge of the system as current as possible. Even with this, there is a lag to the knowledge of the system with respect to distance from different portions of the network.

Modulation Techniques in OFDM EONs:

Further explored in this survey is the modulation format usage and adaptive modulation schemes being considered for OFDM in EONs. Many traditional electric based modulation schemes are adaptable to optical networks, as the mechanism of transfer of information is still wave based, using light instead of current. Similar to these electric based networks, phase shift keying, Quadrature Amplitude Modulation (QAM), and Amplitude-Phase Shift Keying (a-PSK) are explored in conjunction with OFDM [28]. Allowing for multi-level optical modulation formats further increases the ability of the network to handle increased traffic loads at the cost of increasing noise influence. As signal strength is limited, increasing the number of levels of optical modulation causes lower identification levels to be available to distinguish between signals,
thereby causing a set amount of noise to have a higher impact on higher-level modulation formats [5].

This brought about the development of distance adaptive modulation conversion in EONs. This technique allows different modulation techniques to be selected per signal in an EON based upon the destination distance from the source node. This concept is based upon the quality of modulation formats that allows different modulation techniques to have different amounts of spacing between signals and distinguishing characteristics, depending on the modulation format. For example, a 32-QAM or a 16-QAM modulation technique could be used to transmit the same message. The 32-QAM would have the capability of sending the same amount of data as 16-QAM but in less space. Another way to view this is that the use of 32-QAM signals can send more data in a short amount of time than 16-QAM. This comes at the cost of SNR, since the same number of bits of data are stored in a smaller amount of spectral space for the higher modulation technique [3]. This gives less spacing between thresholds to determine what information is being sent. As this travels through an optical channel, noise is added to the signal, causing the output signal to be harder to interpret for 32-QAM.

The concept of distance adaptive transmission for EONs uses this concept to determine where a certain modulation scheme is used on top of the original RSA problem. This is now called the Routing, Modulation, and Spectrum Assignment (RMSA) problem [26]. It has been proven that distance-adaptive modulation formats can improve network performance in many performance metrics including cost, energy usage, and spectrum usage [3]. Testing was done with both anycast and unicast connections.

Modulation and Wavelength Conversion:

This concept can be even further expanded by allowing modulation and wavelength conversion and regeneration to be added to the RSMA problem. This allows intermediate nodes to convert the wavelength or modulation scheme for a signal in the middle of a connection.
Initially this can be viewed as an extension of modulation usage in EONs, allowing a higher throughput modulation scheme to be used where it previously would not reach the destination with a usable SNR, instead allowing for regeneration of the signal at an intermediate point in the transmission. Therefore, the tradeoff can be made of costing the system time, processing speed, power, and therefore throughput to save in spectral resources. In conjunction with a full set of demands for a system, this can be viewed as a tradeoff between either data rate and spectral efficiency for transmission distance [3].

Another option comes with wavelength conversion, allowing for a subsection of a lightpath to use one frequency slice and the rest of the lightpath use another. Wavelength conversion is assumed to use a transponder that can modulate the signal to any arbitrary new wavelength and then convert the signal back to the optical domain, thereby doing an Optical to Electrical to Optical (OEO) conversion [3]. As noted before, this has a heavy cost, as any electrical computation is slow in comparison to transmission times for light-based signals. This defragmentation can lead to significant reduction in total spectrum required by a system [3]. In order to have the capability to do this conversion in a reasonable time period, high speed electrical regenerator/converters, which have very high monetary cost, are necessary. This means that modeling for systems with this capability would have limited nodes with the regeneration capacity. This poses another question on regeneration placement, which itself can be a NP-complete problem [3].
Multicasting in Optical Networks:

From the development of optical networks, it can be seen that many optical networking methods were developed to mimic their electrical equivalencies. This is slightly different with respect to multicasting as optical programming is still in its infancy. All computations done within the communication process must be performed by electric-based computer systems. As the order of magnitude of transfer rate of information for optical networks is larger than that of electrical signals, any processing severely hurts the throughput of optical networks. Therefore, it is of vital importance to reduce the amount of computational methods used in the communication process.

This concept relates to multicasting in a major way. In multicasting, a signal is sent from one source node to a set of destinations. In a graphical analogy, data is sent from a source node to some subset of intermediate nodes and then transmitted to another set of nodes. Finally, the data reaches all of its destinations. To do this in either an optical or electrical circuit switching system, information about the path must be established throughout all of the intermediate nodes, channels must be allocated, and then data must be routed through these edges. A major complication lies in the set of intermediate nodes that have multiple output paths leading from them. Light needs to be split and sent in all output directions. The device currently used for this method, namely light splitters, were not implemented until after the inception of optical network deployment.

Originally, the process of sending data in a multicast optical connection was accomplished by sending data to an intermediate switching node upon which the switching node would establish unicast connections with each of the destinations. This has severe impact upon the throughput of the system as processing time and connection time both take orders of magnitude longer than the propagation time of the signal. This was a key area that needed to change.

Upon the invention and implementation of light splitters for optical networks, the process of switching between nodes at the destination side was removed. This passive component, the light splitter, was able to complete this process without any computation needed. The light wave is
simply split between the outputs thereby preserving the signal, and transmitting in all output directions.

The scope of this thesis is not to examine or utilize the physical layer characteristics of optical networks. This physical layer understanding is important though, as it sets some limitations on the feasibility of actual implementations, so that the methods of simulation or algorithm have real-world meaning.

Multicast Optical Networks and Computational Complexity:

A key background work established for multicast optical networks was done by Wang, et al., in 2006 [6]. This study looked at the application of multicasting in all optical WDM networks with splitting constraints. The core purpose of the paper was to examine the formulation of this problem, determine the feasibility of the defined subproblems with respect to computational complexity, and to determine heuristic approaches for the NP-complete subproblems.

The formulation was done by approaching the routing and wavelength assignment problem in a two-part manner. First, the routing must be done by constructing a multicast tree from source to all destinations. Second, the wavelength assignment problem must find a frequency slice, or a set of frequency slices to assign to each incoming demand. This could be solved simultaneously; however, implementing this in practice is very difficult and runs into major computational complexity issues. Instead, an alternative approach of routing first, then assigning wavelengths was used [6]. Precedent was set for this method before this paper, e.g., [7, 16, 17, 19, 20].

An additional benefit of the two-step procedure includes a decreased computational time for fixed routing, as breaking the RSA problem down into different steps reduces the computational complexity of the system. This has a dramatic effect upon the system state, improving the stance against propagation delay and reducing impact on load balancing of the network [6].

Sahin and Azizoglu [17], suggest a two-step procedure that uses a minimum cost path heuristic as given by Takahashi and Matsuyama [18] to generate the multicast tree. The supplied tree is then
used in the algorithm to solve the frequency slice allocation problem. It is noted in this work that utilizing a single frequency slice throughout a whole network in a practical sense is unrealistic, so the demand could be modelled to use different wavelengths on different subsections of the tree and converted for certain subsets of the included edges.

Another major component of this work is the proof of computational complexity of the different formulations. A set of three different problems are attempted: first, a bi-criterion problem primarily maximizing the number of reached destinations; second, minimizing total cost, a single-criterion problem only maximizing the number of destinations reached; and third, problem consisting of the second problem adding an additional splitting capability constraint [6]. It is proven in the publication that the first problem is NP-complete due to its proof reduction from vertex cover. The second formulation removes the secondary criterion creating a polynomial solvable problem. The third adds a splitting constraint, which once again causes the problem to be NP-complete.

This two-step procedure is vital to this thesis formulation, as it sets the precedence for creating an algorithm with the candidate trees, or supplied tree, being already solved. This helps to alleviate the computational complexity of the system.

Multicast Elastic Data Streams:

Multicast tree construction is a highly studied problem. Each iteration of network topology or protocol creation requires a new multicast problem to be investigated. This is extended to work with elastic data sets shown by Zhu, et al., [2] in an adaptive multicast tree construction for elastic data streams. In this work, elastic data is modeled as datasets that can be downsampled. This concept, core to elasticity, allows for better utilization of the fragmented spectral resources.

This methodology uses a formulation defining multicast calls with a tuple containing the source node, a set of receivers, and a stream identifier. This can be reduced to only needed the source node and a stream to identify the multicast session [2].
The paper states that this elastic data stream concept is critical to the working methods of Voice-over-IP and video streaming services. The elasticity allows for downsampling of the data stream meaning that a connection can be established even if there are not enough network resources to accommodate the full demand at some portion of the tree, so long as it meets a minimum criterion. This is key to this thesis, as the concept of downsampling at different portions of a multicast tree gives the basis of PDR. It is also noteworthy that the study of Zhu et. Al., [2] was done with respect to general elastic data streams, not elastic optical networks.

Modelling of Multicast Elastic Optical Networks:

Modelling for multicast EONs can be approached in a variety of ways. In Walkowiak, et al., [1], a flow multicast model, a candidate tree model, and a heuristic approach are taken.

The flow model uses a classical approach to modelling multicast flow in computer networks shown by Dahl, et al.[21]. It is based on a unicast multi-commodity node-link formulation. This means that for each receiver in the multicast session, a unicast path originating at the root is created, and all unicast paths form a multicast tree. This work does not supply the number of required slices; instead, it is calculated within the procedure. The paper then extends the problem past this thesis’s scope of interest by adding modulation format ranges and bit rates and evaluates runtime and average number of frequency slices required with respect to each of the tested approaches [1].

The candidate tree model assumes that, for each multicast session, there is a set of candidate trees that start at the source and has all destinations as leaf nodes. A candidate tree is then selected based upon its calculated value, and determines a slice index according to its primary calculation.

The heuristic approach is also a candidate tree approach. The approach uses a version of Adaptive Frequency Assignment – Multicast from Walkowiak and Klinkowski [22]. The major goal for this approach is by adaptation, to choose the ordering of demands for the allocation of each single demand. It uses the systems bit-rate, the bit-rate multiplied by number of receivers, the bit-rate multiplied by the number of required frequency slices, and the last metric multiplied by the number
of receivers. This is a polynomial solution growing with the rate of demands, number of candidate trees, edges, and nodes.

The key result of this work is that the best solution found with respect to the performance metrics among the different solving methods varied with topology and number of demands/network topology. In general, the higher order candidate tree approaches were very successful in finding the better solutions, though this comes at the cost of very high computational time. It is an Integer Linear Programming (ILP) formulation, so the runtime will always grow rapidly. On the other hand, the heuristic approach gave reasonable values in multiple testing setups and has a vastly lower runtime. Therefore, the recommendation of the paper is that the candidate tree methods can be used to find the best trade-off between solution and runtime by selecting number of candidate trees.

This is key to this thesis’s formulation, as the method attempted is a candidate tree approach. The works of Walkowiak and Klinkowski [1] give a great foundation on which the formulation is built for then adding the PDR extension and developing and evaluating performance metric comparison study.
Chapter 3 - Formulation Setup:

Proposal Problem Statement:

The problem examined for the core of this thesis proposal work is modelling and examining the value of PDR in EONs using a supplied/candidate tree approach. Once modelled, the effect of PDR is quantified by comparison to other existing EON multicast systems. This initially is simply a comparison to a non-PDR multicast EON with candidate trees. This can be extended to encompass many different enhanced EONs with distance adaption, wavelength conversion, modulation conversion or other value-adding procedures.

Formulation Subsections:

In order to model this problem statement, many core features of multicast EONs must be implemented. The descriptions for the following subsections follow this section sequentially.

- Multicast Tree Construction Problem
- Basic ILP/MILP Structure
  - Center Frequency and Radius vs Start Frequency and Range
  - Elastic Data Representation
  - Downstream Continuity
- Multi-Tier MILP Structure
- Objective Function Setup
- Core Constraint Setup
- Computational Complexity
- Software Background

Multicast Tree Construction Problem:

The multicast tree construction problem is a defined as follows. Given a connected graph of \( N \) nodes and \( M \) edges, find the minimum cost tree with the source node as the root and all destinations as leaves. The definition of minimum cost can be defined according to the problem definition.
There are many algorithms that exist to solve problems like this. The solutions depend upon the definition of minimum cost. A simple approach would be to find the minimum spanning tree for the graph, weighting each of the edges by the remaining available frequency slots. Alternatively, an algorithm could be written to check all possible trees and select the “best” based upon some weighted criteria.

The difficulty with this tree construction lies within its interaction with the PDR in the EON formulation. As the formulation for PDR in EONs is a MILP problem and performed as an all-at-once operation, the definition of “best” tree can vary with respect to the determined highest value state of the objective function. In other words, each iteration of searching through the decision space for the MILP can change what is the optimal tree to use for a given connection. This means that the runtime of the algorithm is now in the order of the runtime of the PDR algorithm multiplied by the runtime of the best tree algorithm.

In order to combat this increased runtime to the system, a candidate tree approach is being taken. Instead of reevaluating the system state at every iteration, a group of candidate trees is being considered for each connection. This means that a static amount of time is being used for creating the trees. This does not increase the order of magnitude of the total system runtime. In the current formulation, due to license limitations, the number of candidate trees per connection is one (1). This has been tested with more than one tree but, in order to fit limitations and get a reasonable sample size, it has been reduced for this work.

Determination of which method to take with respect to tree construction is the first assumption made for this thesis. It was determined that the candidate tree approach offered the most benefit, as it allows a tradeoff of resolution of the answer with runtime. This was determined by examining the output ILP results of Walkowiak and Klinkowski [1].
Chapter 4 - Basic ILP/MILP Structure:

The initial formulation for PDR in EONs using a candidate tree approach is set up in a two-tiered system.

First, there is the allocation matrix, $X$, which is a three dimensional Boolean matrix of size $[n,d,w]$, where $n$ is the number of nodes in the graph, $d$ is the number of demands, and $w$ is the number of frequency slices. This allocation matrix, $X(i,j,k)$, contains the information of whether edge $i$ uses frequency slice $j$ on demand $k$.

Second, there is the connectivity matrix, $Z$, which is also a three dimensional Boolean matrix of size $[n,d,w]$, where $n$ is the number of nodes in the graph, $d$ is the number of demands, and $w$ is the number of frequency slices. This connectivity matrix, $Z(i,j,k)$, contains the information of whether node $i$ on demand $k$ is connected to the source node using frequency slice $j$. This can only be true if there exists a path from source to $i$ which all have $X(i,j,k) = true$ for some demand/frequency slice pair on all nodes in the path from the source to the current node.

This pair of Boolean matrices was the way the initial MILP was structured. All the information needed is stored within the allocation matrix to model the MILP. The connectivity matrix only adds a small amount of computational complexity while allowing for much clearer interpretation of data. All constraints could be written as a function of the allocation matrix, but the connectivity constraint would have to be added to every other constraint that needs to verify connectivity.

Center Frequency Slice and Radius vs Start Frequency Slice and Range:

The next area concern for the basic formulation is the creation of elastic frequency range for each connection, which is done in this work by creation of variable frequency slices from the supplied bandwidth. This is covered in the next section but a key point needs to be discussed before interpreting the elastic representation. In order to write constraint functions for the elasticity constraints, there needs to be a definition for a set of continuous frequency slices and a behavior for how to decrease bandwidth for PDR.
This means that there needs to be either a center frequency slice plus a radius of connection, or a starting index and length of connection. Both options have merit, as the center frequency slice and radius approach more similarly models the real-world equivalencies, but the start frequency slice and length allows for a fine granularity for increasing and decreasing bandwidths for connection.

Upon examination of current state of the art publications, the start band and range approach has been used after the preliminary formulation [1]. This offers a key advantage for the preliminary formulation, as it increases the resolution for small sample-size systems. As the software at this point is limiting the total number of constraints in the preliminary results, this added resolution is of vital importance.

Elastic Data Representation:

One of the core differentiating features of this thesis work versus existing work is performing this PDR function in the elastic optical domain. This is reflected in the formulation by not creating specific-sized large channels of differing bandwidths. Instead, the formulation is devised to create a large number of equally sized channels, which are implemented as the third dimension in the allocation matrix. This allows for the simulation of creating superchannels, sized at multiples of the bandwidth of each individual small channel.

A set of constraints are created that only allow a connection to be established in the connection matrix if all of the following connectivity constraints are satisfied:

- There exists the same range of frequency slices of at least $K$ number of contiguous frequency slices allocated to the signal for all required destinations and all parent edges of each destination on a single node/demand pairing ($K$ would represent the minimum number of required frequency bands to establish with PDR).
- Downstream continuity is conserved.
Downstream Continuity:

The concept of downstream continuity controls the availability of certain frequency slices depends upon the usage of the frequency slices on the parent edge of the current edge. In other words, a child edge can only use a specific frequency slice on a node/destination pair if its parent uses the same frequency slice on this same node/destination pair. As this would propagate through all levels of the multicast tree, this forces each leaf node to only have availability to use the frequency slices that are utilized by all other members of the tree which exist in the path from the source node to this leaf node. This alone enforces the continuity constraint for this MILP.

Multi-Tier MILP Structure:

After multiple iterations of the two-tiered system, a multi-tiered system was developed. This extended the two matrix decision variables (allocation and connectivity) to include a few extra decision variables that are designed to fully define the performance metric’s decision variable in the formulation. This change of formulation relies mostly upon the nature of linear programming and the concept of bounded and unbounded space. It allows for the definition of percent demand throughput to not only depend upon number of connected demands and their respective spectral demand sizes, but also the percent PDR applied to each of the destination nodes for each demand.

In order to write this program in a manner robust enough to withstand changes to the objective function in this way, a modular approach was taken. This separated the definitions of most decision variables from the majority of other decision variables. The goal was to make a layered modular system in which decision variables at a specific layer would only need to examine those in the layer directly above it. This is mostly true for each of the formulations now, except for a few instances where band start indices are verified by the connectivity matrix which is a 2-tier gap. An example of this type of tiered system can be found in Appendix 2.
Objective Function Setup:

The objective function is the driving force behind the optimization of the problem. Initially, I was using a dual-objective optimization definition with a primary and secondary criterion. For example, this could have been used to maximize number of connected destinations with a secondary criterion to minimize maximum utilized bandwidth in each edge. This has drastic ramifications upon computational complexity and has now been determined to be outside of the range of what is defined as the objective of this thesis. Now, the objective function is a single criterion function, e.g., to maximize percent demand throughput.
Core Constraint Setup:

Since this formulation uses a candidate tree/supplied tree approach, two key constraint types are needed: connectivity and continuity.

The connectivity constraint is multi-tiered, as the connectivity must first show that there exists a path/ frequency slice pair between the source and destination, and then determine if the set of frequency slices is of sufficient bandwidth, then determine if the entire tree exhibits connectivity. This is done through a series of constraints and is the driving force of such decision variables as bandStartIndex and bandwidthsInBandrange as shown in the upcoming “Current Formulation” section.

The continuity constraint is much simpler, as the formulation uses a downstream continuity approach. This ensures that the overarching continuity constraint is satisfied so long as the set of usable frequency slices on any child node is only that of the utilized frequency slices of its parent. Since our topology of the multicast call is a tree, this means that the child must only look at one node’s system state to determine this usable region.

Computational Complexity:

The computational complexity of this formulation is not the core purpose of this thesis but still has direct impacts upon the output. Since satisfiability (SAT), which is a known NP-complete problem, can be reduced to the solving of an ILP formulation, it is proven that ILP formulations are NP-complete [10]. This poses a major obstacle for our work since any NP-complete problem set is going to grow at an uncontrollable rate, eventually. This already high computational growth is compounded by the candidate tree approach, making the runtime for this formulation very problematic on a large scale. This limits the set of choices for variables and parameters for this thesis. The work endeavors to show value for the sets of data that can be used given the current means to handle the computational complexity in reasonable solution times.
Chapter 5 - Formulation:

Initial Formulation:

Problem Statement:

Show the value of PDR in EONs with a supplied tree approach by comparing percent destinations met for PDR system with non-PDR system.

Assumptions:

- Global knowledge of system state at all times
- Provided multicast tree is optimal (or provided trees are the required paths)
- Demands/Paths/Schedule/Graph are known beforehand
- Demands provide minimal/optimal required wavelengths to destinations
- No guard band required
- Unlimited range of signals (no regeneration)
- Constant modulation (i.e., no modulation conversion)
- No wavelength conversion
- No cost of signal splitting
- All nodes are capable of signal splitting
- All nodes can be destination nodes (excluding node designated as source)
- Same Source node for all demands
- Network Empty Initially
Constant Definitions:

\[ C_{ik} \]: (Integer) Cost of wavelength \( k \) on edge \( i \)

\[ D_{ij} \]: (Boolean) 1 iff node \( i \) is a destination node for connection \( j \)

\[ M_j \]: Center frequency for connection \( j \)

\( n \): (Integer) Number of nodes in the graph.

\[ P_i \]: Set of parent edges of node \( i \)

\[ T \]: (Integer) Number of Total Multicast Connections Demanded for Static Problem

(Also, equal to number of supplied multicast trees)

\[ U_{\text{min}} \]: Minimum required overall utility (range 0 to 1)

Used in Utility Definition in Constraint (7)

\[ \Lambda_i \]: Set of available wavelengths on edge \( i \)

Variable Definitions:

\[ R_i \]: (Integer) Radius of connection to node \( i \). (Number of bandwidth from \( M_i \) to edge of transmission range, or \( 2*R_i+1 \) = total bandwidths allocated)

\[ U_j \]: (Float: Range 0:1) Defined Utility of System

\[ X_{ijk} \]: (Boolean) 1 iff Edge \( i \) for connection \( j \) uses wavelength \( k \).

\[ X_{ik} \]: (Boolean) 1 iff wavelength \( k \) is used on edge \( i \)

\[ Z_{ij} \]: (Boolean) 1 iff node \( i \) is connected to source node for connection \( j \)

(wrt: min required bandwidth)

\[ Z_{ijk} \]: (Boolean) 1 iff node \( i \) is connected to the source node for connection \( j \) using wavelength \( k \). (Path exists from source to destination \( i \))
Objective Function (Maximize $z$):

$$z = (\sum_{i=1}^{n} \sum_{j=1}^{T} d_{ij} z_{ij}) + \varepsilon(\sum_{i=1}^{n} \sum_{k \in \Lambda_{i}} c_{ik} x_{ik})$$

(5.1.1)

Notes:

Primary Goal: maximize number of destinations reached for all connections

Secondary Goal: minimize cost of used edges

Connectivity Constraints:

$$Z_{jk} \leq \frac{\sum_{i \in P} X_{ijk}}{\sum P_i} \text{ for all } (M_j - R_j) \leq k \leq (M_j + R_j),$$

(5.1.2)

Notes:

Ensures that $Z_{ijk}$ is only connected if all parent edges are used (1 frequency slice).

$$Z_{ij} \leq \frac{\sum_{k=R_i}^{k+R_i} Z_{ijk}}{2R_i + 1}$$

(5.1.3)

Notes:

Ensures that $Z_{ij}$ is only connected if there exists a set of continuous set of frequency slices allocated to this connection on all parent edges.
\[
\sum_{j=1}^{n} X_{ijk} \leq 1 \quad (5.1.4)
\]

\[
X_{ijk} \leq X_{ik} \quad (5.1.5)
\]

Notes:

Ensures that no wavelengths are used more than once on multiple connections.

Continuity Constraints:

(6)

\[
R_i \leq R_j \text{ for all } j \in P_i \quad (5.1.6)
\]

Notes:

Downstream continuity constraint. Edges downstream from a parent edge cannot utilize wavelengths that are not utilized by the parent edge. In conjunction with the connectivity constraints, this should encompass the entirety of the EON multicast setup.

\[
R_i \leq 1 \quad (5.1.7)
\]
Initial Formulation Discussion:

The initial formulation creates a decision space for the ILP solver based around the formulation discussion from the previous sections. The objective of this first formulation was to show that value could be found by examining the PDR system’s optimal system state with respect to maximizing number of destinations reached and comparing it to the theoretical ideal state of the non-PDR system. As only small systems were being analyzed at this point, analytical answers were easy to verify. The problem supplied then could be modified to exhibit certain behaviors as to check corner cases, and specific boundary condition/dual values to ensure that the ILP was behaving correctly.

Initial Problems:

A few major problems were found with this preliminary formulation.

First, the connectivity constraints were written with respect to upper and lower bounds of the summation using the range decision variable. As this was devised before learning the limitations of the formulation format for CPLEX, this needed to be changed. This was modified by creating a series of linear constraints that bounded the decision space in the same manner as the initial constraint.

Second, the initial formulation still used the center frequency slice and radius approach instead of the current start frequency slice and range approach. This was changed because it agrees with the state of the art papers examined and increases resolution of possible “elastic” ranges.
Change Log:

The following was changed from the initial formulation through a series of new formulations resulting in the current state:

- Rewrite connectivity constraint with range bounds as set of constraints. The range variable becomes a constant max range, and a new constant for percent PDR allowed is added.
- Change center frequency slice and radius to start frequency slice and range.
- Rewrite constraint equations in terms of ranges instead of upper and lower bounds.
- Removed Z2 Variable – added a band start index variable instead.
- Removed redundancies for matrices and created new matrices instead (i.e., removed 2nd Z variable with different indexing and made it more readable with the band start index variables).
- Redefined objective function definition with respect to max number of contiguous frequency slices used for each path in multicast demands reaching the destination nodes.
Final Formulation:

Problem Statement:

Show the value of PDR in EONs with a supplied tree approach by comparing percent bandwidth utilization for PDR system with non-PDR system.

Assumptions: (same as initial formulation)

The full algorithm implemented for CPLEX optimization is located in Appendix 3.
Constant Definitions:

\[ D_{ij} \]: (Boolean) 1 iff edge \( i \) is a destination edge for connection \( j \)

\[ E_{ij} \]: (Boolean) 1 iff edge \( i \) is demanded for connection \( j \)

\[ K \]: (Float+) Minimum percentage bandwidth allowed to destination nodes

\[ P_i \]: (set of Int+) Set of parent edges of edge \( i \)

\[ R_j \]: (Int+) Number of frequency slices demanded for connection \( j \)

\[ n \]: (Int+) Total number of edges

\[ d \]: (Int+) Total number of demands

\[ f \]: (Int+) Total number of frequency slices per edge (assumed all same for each edge)

Variable Definitions:

\[ B \]: (Float+) Percent Demand Throughput

\[ Y_{ij} \]: (Int+) frequency slices in bandrange used for edge \( i \), demand \( j \)

\[ H_{ijk} \]: (Boolean) Allocation Matrix for frequency slice start \( k \) for edge \( i \), demand \( j \)

\[ H_{jk} \]: (Boolean) Allocation Matrix for frequency slice start without edges for edge \( i \), demand \( j \)

\[ X_{ijk} \]: (Boolean) 1 iff edge \( i \) for connection \( j \) uses frequency slice \( k \)

\[ Z_{ijk} \]: (Boolean) 1 iff edge \( i \) is connected to the source for connection \( j \) using wavelength \( k \). (Path exists from source to destination \( i \))

Range Definitions:

Edges = 0 .. (\( n-1 \))

Demands = 0 .. (\( d-1 \))

freqSlices = 0 .. (\( f-1 \))

frequencySlicesBuffered = 0 .. (\( f + R-1 \))
Objective Function (Maximize B):

\[
B \leq \frac{\sum_{i=0}^{n-1} \sum_{j=0}^{d-1} Y_{ij}}{\sum_{j=0}^{d-1} D_j R_j}
\]  

Overlap Constraint:

\[
\sum_{j=0}^{d-1} X_{ijk} \leq 1 \text{ for all } (i \in \text{Edges}, k \in \text{freqSlices})
\]

This constraint ensures that there are no two demands that utilize the same frequency slice/node pair.

Connectivity Constraints:

\[
Z_{ijk} \leq \frac{\sum_{m \in P} X_{mk}}{\sum_{P_i} P_i} \text{ for all } (i \in \text{Edges}, j \in \text{Demands}, k \in \text{freqSlices})
\]

Notes:

Ensures that \(Z_{ijk}\) is only connected if all parent edges are used.
\[ H_{ijk} \leq \sum_{r=0}^{\text{ceil}((R_j+1)*K-1)} \frac{Z_{i,j,(k+r)}}{\text{ceil}(R_j+2)*k} \] for all \( i \in \text{Edges}, j \in \text{Demands}, k \in \text{freqSlices} \) \hspace{1cm} (5.2.4)

Notes:

Ensures that a connection only has a start index if there exists a continuous range of frequency slices after the start point that is at least as large as the minimum percent PDR multiplied by the required max bandwidth.

\[ H_{jk} \leq \sum_{i \in \text{Edges}} \frac{H_{ijk}D_{ji}}{\sum_{m \in \text{Edges}} D_{jm}} \] for all \( j \in \text{Demands}, k \in \text{freqSlices} \) \hspace{1cm} (5.2.5)

Notes:

Ensures that all nodes in a demand matrix use the same starting index value.

\[ Y_g \leq \max(\sum_{r=0..R_j} Z_{i,j,(k+r)}D_{j,i}) \] for all \( i \in \text{Edges}, j \in \text{Demands}, k \in \text{freqSlices} \) \hspace{1cm} (5.2.6)

Notes:

Sums the total number of frequency slices in the bandrange for a node/demand pair.
Chapter 6 - Preliminary Results (From Proposal):

The preliminary formulation was built to optimize the percent bandwidth utilization given a list of supplied trees, demand matrix, destination matrix, ideal bandwidth array, and percent PDR limit.

The formulation is severely limited by the software edition currently owned by Miami University. The solution to this problem is obtaining a new enhanced version of the software, which is currently being processed by IT Services. The current version of the software limits the set of constraints to 300. This means that the number of constraints, which is dependent on number of nodes \( n \), number of demands \( d \), and number of frequency slices \( w \) in the current formulation cannot grow past a very small size. Since the formulation is built upon three-dimensional matrices, if the sum of \( n \times d \times w \) for all constraints bypasses the limit, the solver cannot execute. Therefore, I am restricted to using small sample sizes.

For this solution, the following is used:

- Graph from Appendix 1
- 5 frequency slices on all edges (4 on second plot)
- 5 destination nodes (6 total nodes including source)
- Common source node for all demands (node 1)
- A Variable set of Demands (indexed by \( x \)-axis). For \( x = 1 \), only 1 demand is supplied. For \( x = 2 \), 2 demands are supplied to each system. The demands are the same supplied to each system. Each demand has a specific supplied tree, parent matrix, demand required, and set of destinations.
- Minimum PDR Percent = 70%
Figure 6-1: 4 edges, 5 demands, 5 slices Percent Demand Throughput vs Number of Frequency Slices Demanded

Figure 6-2: 4 edges, 5 demands, 5 slices Percent Demand Throughput vs Number of Frequency Slices Demanded
Interpretation of Results:

Although severely limited in sample size, this solution already shows a noticeable difference in percent bandwidth utilization in the PDR case vs. the non-PDR case. This is very promising, as this indicates that there is value in PDR with respect to percent bandwidth utilization. It is not nearly a large enough sample to quantify value on a large system, nor does this implement the candidate tree choice yet; however, it is a very promising start to the implementation. As the goal for the proposal was to show preliminary proof of value, this has been achieved.

Runtime:

Currently runtime does not factor into the problem of this size. Compile time is only a few seconds and runtime is ranging in the millisecond durations. As this is only a small sample, that is to be expected. This will be a key determination for the larger scale problems though, as it will limit my total solution size for the defense.
Chapter 7 - Expansion of Results (Defense):

License Change:

For all data following the initial formulation and preliminary data, a new academic license was used. This license, issued to Miami University, allows for an unlimited number of constraints to be utilized for the formulation.

Runtime Analysis:

With the new license allowing for an unlimited number of constraints to be applied to the test system, runtime analysis becomes the primary focus. Due to the exponential growth rate of NP-complete test systems, the formulation needs to be carefully bounded and variable sizes must be chosen to optimize the scope of values examined without causing the runtime to grow to values which exceed viable testing times. For this thesis, testing times were limited to 1-8 hours per test, depending upon the size of the system and its complexity.

As a result of the runtime analysis, the candidate tree method proposed in the thesis proposal was removed from the formulation, as it was determined that the added benefit was outweighed by the computational costs that it introduced to the system.

Memory Limitations:

A secondary limitation found is the possibility that large systems exceed memory caps from the physical computer testing system. Larger test systems require very large amounts of memory space to store all information for the branch-and-bound searching algorithm used by CPLEX. As the solver progresses through found solutions, the amount of memory usage also grows to store the information of newly encountered solutions and system states. Therefore, not only do larger test systems require larger memory space, but also more computationally difficult problems utilize more memory, as solutions are found after larger periods of time. Due to issues with memory usage, many of the larger test systems were limited to 1-3 hours or until maximum memory usage was reached.
Percent Demand Throughput vs PDR:

Test System:
6 edges (Graph in Appendix 4)
6 demands
1 to 9 frequency slices
4 max frequency slices by demands
PDR: 0.25, 0.5, 0.75, 1.0

In this test, percent demand throughput is examined as a function of number of available frequency slices for a set of edges and demands with a given routing table. This data is compared for each of the PDR values from 0.25 to 1.0 and graphically displayed.

Result:

![Percent Demand Throughput vs Number of Available Frequency Slices](image)

*Figure 7-1: 6 edge, 6 demand Percent Demand Throughput vs Number of Available Frequency Slices*
Discussion:

The graph shows the effect of different PDR values on the system’s ability to effectively solve the spectral assignment problem. As the number of frequency slices increases, the 0.25 PDR version of the system is able to utilize a much larger percentage of the available frequency slices when compared to higher PDR value systems. For this example, the 0.25 PDR slope changes very little between each pair of adjacent points. This indicates that the spectral assignment algorithm is able to utilize a very large percentage of this newly available spectral space. Alternatively, the 1.0 PDR version of the system is only able to make use of every 4 frequency slices added, as no partial portions of the system can be utilized if there are less than 4 contiguous slices available.

Significance:

Utilization of the added adaptability of the low PDR versions of the system can have significant impact upon scheduled and dynamic allocation of demands to systems. In a real-world example of a large network topology and demand set, demands occur over time, causing increasing amounts of fragmentation of the spectral space as time passes. With a PDR version of this system, new demands can better utilize the spectral space that has been fragmented into less than optimal sets of frequency slices. This subsection of spectrum would previously be unusable to a non-PDR version of the system, but some lower PDR connections could make use of the space. Instead of waiting for either of the adjacent connections to end to utilize the fragmented space, lower than optimal spectral allocations can be utilized by the PDR versions of the system thereby increasing total throughput and spectral efficiency.
Percent Demand Throughput vs Network Topology.

In this section, PDR’s effect upon networks with different topologies is investigated. The three chosen topologies are 6 node ring, and 9 node directed mesh, and 7 node binary tree. All three tests were performed with the same set of 6 demands and investigated over a range of 1 to 6 frequency slices. The number of frequency slices is set at a maximum of 6 through investigation resulting in a loss of accuracy resulting after this value. This is due to the limitations of the testing system, thus only the first 6 indices are used.

6 Node Ring Test:
6 Node Ring Topology Graph:

![Figure 7-2: 6 Node Ring Topology Graph](image-url)
Ring Test System:
10 edges
6 demands
1 to 6 frequency slices
4 max frequency slices by demands
PDR: 0.25, 0.5, 0.75, 1.0

Results:

Discussion:

The effects of PDR in the ring topology are less pronounced than expected. The first 4 frequency slices behave as expected, allowing connections at low PDR where higher PDR connections cannot be created. The interesting section of this plot is at indices 5 and 6, which have better behavior at PDR 0.25 and 0.5, but no noticeable value from PDR 0.75. PDR 0.75 will never have added benefit over PDR 1.0 at index 5, as by definition, they are the same at this index. This is not the same at PDR 6 though, as the algorithm could not find a better solution
with the 0.75 PDR system over the 1.0. The ability to use 3 frequency slices instead of 4 on edges did not allow any increased total percent demand throughput for the system. This is likely due to the linear nature of the ring topology, where demands are higher for edges adjacent to the source and sparser further away from the source.

9 Node Mesh Test:

9 Node Mesh Graph:

![9 Node Mesh Topology Graph](image)

**Figure 7-4: 9 Node Mesh Topology Graph**

Mesh Test System:
12 edges
6 demands
1 to 6 frequency slices
4 max frequency slices by demands
PDR: 0.25, 0.5, 0.75, 1.0
Results:

![](figure7-5.png)

**Figure 7-5: 9 Node Mesh Topology Percent Demand Throughput vs Number of Available Frequency Slices**

Discussion:

The mesh topology is more spread out than the ring topology. The number of edges which are used enough to reach capacity of frequency slices is significantly higher than the ring topology. In the ring topology, the nodes adjacent to the source node have very high traffic, while the rest of the ring is lower. Although 2 paths exist to reach any destination, the utilization of edges not adjacent to the source node is highly dependent upon the demand set. For the mesh topology, the number of demands which utilize large portions of the quadrants of the mesh leading from the source node is very high, as there are only three edges in each direction from source. This allows the PDR method to prioritize the demands which best utilize the largest portions of each quadrant, while fulfilling partial demands for the rest of the multicast tree. Also, the 6 slice answers found for PDR 0.25 and 0.5 are lower than the PDR 0.75 answer. This indicates that the ideal system state was not found for PDR 0.25 and 0.5 in the 8 hour runtime for the data set.
7 Node Binary Tree Test:

9 Node Binary Tree Graph:

![7 Node Binary Tree Topology Graph](image)

Mesh Test System:
- 12 edges
- 6 demands
- 1 to 6 frequency slices
- 4 max frequency slices by demands
- PDR: 0.25, 0.5, 0.75, 1.0
Results:

![7 Node Binary Tree Topology Percent Demand Throughput vs Number of Available Frequency Slices](image)

Discussion:

The three topology tests all behave in a similar manner. Very low frequency slices, between 1 and 4, have very similar behavior in all three of the tests. Since lower PDR value systems are better able to adapt to the lowest frequency slice values on all tests. The area where the three differentiate greatly is on frequency slice 5 and 6. The linear trend of the PDR 0.25 continues through all of the tests but the ability for PDR 0.5 and 0.75 to react to the change of frequency slices is different. In the ring test system, only PDR 0.25 and 0.5 change at all between 4 and 6 frequency slices, while for the other 2 tests all three PDR values less than 1 are able to adapt.
Significance:

The topology change has some impact on the effectiveness of different PDR values. Specifically, the more compact topologies, such as ring, have less of an ability to adapt with the PDR function. This is likely due to the linear nature of the ring topology. Edges 1 and 6 of the ring topology must accommodate all traffic sourced from node A, which results in an optimization function which effectively attempts to find the best allocation on those two edges with respect to the number of destination nodes for the associated demand. This effect is also noticed in the binary tree topology, as edges 1 and 4 have a similar bottleneck capability. The effect is not as pronounced in this case as each direction from the source node in the binary tree can access 3 nodes, which allows for a much larger weight of different demands which can use one or both sides of the tree.

The mesh topology is the most spread, with demands that can use one to four of the directions from the source node. This added number of directions allows for partial demands to fit more easily, as demands are spread out more, leaving more space to fit subsets of the desired frequency slices.

Conclusions:

While topology does have some noticeable impact on these small systems, it is predominantly caused by the bottlenecks created in different areas of the graph. The more spread out topologies allow for more frequency slices to be used across all of the edges in the graph, resulting in more throughput and reactivity to PDR. Overall, PDR does have a positive effect on all topologies, but its effect is more pronounced on the topologies where a higher percentage of the edges are utilized to capacity.
Confidence Testing:

The primary goal of this set of tests is to gauge at what system size the results become inconsistent. For all ideals solutions, the trend of percent demand throughput should decrease with respect to an increasing PDR value. This is due to the feasible region of the larger PDR system being fully contained within the feasible region of the lower PDR system. This causes any solution found within a larger PDR system to be possible as a solution for smaller PDR systems.

Although it can be shown for a variety of very small systems similar to the preliminary results example that PDR allows for better solutions to be found with respect to percent demand throughput, once systems are increased in size much beyond the example test size, the optimal solution may no longer be able to be found within testing time and memory limitations. This limitation is the smaller of 8 hours of testing time or time to hit memory cap. Time to hit memory cap is dependent on the rate of growth of the system state, increasing with respect to number of edges, demands, frequency slices, max frequency slices by demands, or lower PDR values.

In this section, different test systems are used to determine if the ideal solution is found within the runtime and memory bounds. If the ideal solution is found for the PDR 1.0 system, then an increase in performance can be shown if a solution for any lower PDR values is found have a higher percent demand throughput. If the ideal solution for the PDR 1.0 system is not found, then results are inconclusive even if a better solution is found for the lower PDR solution.
Confidence Test System 1:

10 edges (graph in Appendix 5)
20 demands
20 frequency slices
5 max frequency slices by demands
PDR: 0.2, 0.4, 0.6, 0.8, 1.0

Result:

All PDR values run to memory cap resulted in no solution found.

Conclusion:

This system is significantly too large for testing with the current resources available.
Confidence Testing System 2:
6 edges (graph in Appendix 4)
6 demands
6 frequency slices
4 max frequency slices by demands
PDR: 0.25, 0.5, 0.75, 1.0

Result:

Optimal answers were not found for any PDR values, though a significant number of possible solutions were searched.

![Figure 7-8: Confidence Test System 2 Percent Demand Throughput vs PDR](image)

Conclusion:

This system exhibits the correct behavior for PDR but does search all available combinations of demands. The best answer found for each of the above data points is found within 5 seconds (for PDR 1.0) to 5 minutes (for PDR 0.25) but this cannot search the entire feasible region in 8 hours
of runtime. From this data, it can be determined that it is unlikely that larger system will be viable to solve for the optimal answer for all PDR values within testing boundaries, as this small system was already too large to search all possible answers. This does not invalidate the results though, as finding some solution which exists in the feasible region that is better than the ideal Non-PDR value still shows value of the PDR algorithm being added to the system. For real world implementation, this would not be a viable option, as systems are vastly larger than the given test systems, which already reach the runtime limitations.

It is also noteworthy that systems of such a small size can be analytically solved by hand to verify the above answers, indicating that the correct answer is found in a small subset of the runtime, through the upper bound solution is not reduced in a useful amount of time, resulting in the algorithm not being able to verify that the found answer is indeed optimal.
Confidence Test System 3:
6 nodes (graph in Appendix 4)
10 demands
6 frequency slices
4 max frequency slices by demands
PDR: 0.25, 0.5, 0.75, 1.0

Result:
Optimal answers were not found for any PDR, though a significant number of possible solutions were searched.

Conclusion:
This system exhibits the correct behavior for PDR, but does search all available combinations of demands. The best answer found for each of the above data points is found within 2 minutes (for PDR 1.0) to 30 minutes (for PDR 0.25) but this cannot search the entire feasible region in 8 hours of runtime. Higher percent demand throughput values are found for the lowest PDR value, but
given the amount of time required to find the solution, there is less confidence that enough of the feasible region is searched to find the ideal answer. Due to the small sample size of this problem, saturation occurs between 0.25 and 0.5 PDR. This is very common in M-ILP tests for PDR, as the lower PDR values can only be confidently found for very small systems, causing differences between larger PDR values to be only apparent if only a few combinations of $\frac{1}{2}$ or $\frac{3}{4}$ of the demand are used in the ideal system state.
Confidence Test System 4:
6 nodes (graph in Appendix 6)
10 demands
21 frequency slices
5 max frequency slices by demands
PDR: 0.2, 0.4, 0.6, 0.8, 1.0

Result:

Optimal answers were not found for any PDR. Only a small subset of the feasible region is searched to find the best returned answer.

Conclusion:

This system exhibits the correct behavior for PDR once again, but does not search a significant portion of the feasible region. The ideal solution for the Non-PDR solution may not have been found. The best answer found for each of the above data points is found within 5 minutes (for PDR 0.8) to 45 minutes (for PDR 0.2), but this cannot search the entire feasible region in 8 hours.
of runtime. Better answers are found for lower PDR values but, given the amount of time required to find the solution, this system size appears to be too large to search a significant portion of the feasible region. This is interesting because the correct trend is still appearing for the percent demand throughput with respect to PDR. Since only a small portion of the feasible region is searched, it is likely that there are better answers for each of the values found but the returned answer is the best found within testing limitations of 8 hours or memory cap.
Confidence Test System 5:
6 nodes (graph in Appendix 4)
10 demands
8 frequency slices
4 max frequency slices by demands
PDR: 0.25, 0.5, 0.75, 1.0

Result:

Optimal answers were not found for any PDR. Only a small subset of the feasible region is searched to find the best returned answer.

![Figure 7-11: Reliability Test System 5 Percent Demand Throughput vs PDR](image)

Conclusion:

This system does not exhibit the correct behavior for PDR. This is significant because this is not the largest system tested, nor the most complex. The best answer found for each of the above data points is found within 5 minutes (for PDR 0.8) to 45 minutes (for PDR 0.2) but this cannot search
the entire feasible region in 8 hours of runtime. This example does not find the optimal answers for PDR 0.2, 0.4, or 0.6 for certain, as the feasible region of the lower PDR values totally encompassed these of the higher PDR values. This means that any solution for PDR 0.8 would be a possible solution for PDR 0.6. This indicates that this system size is unreliable, as not enough of the feasible region is searched to find the close to optimal answers.
Chapter 8 - Future Work

To expand upon the work done in this thesis, the next logical step would be to implement a heuristic approach which mimics the logic of the M-ILP. This algorithm should be able to allow the flexibility inherent to PDR, which allows for subsections of the multicast tree’s edges to utilize smaller than optimal number of contiguous frequency slices. This could be accomplished in a variety of ways. One option would be to create a priority system for all demands in the demand set. Not only would these demands include all of the original 100% throughput demands, but also create new demands for each of the original demands at different PDR ratio values. Then, the algorithm would only allow for one of the possible PDR applications to be applied and systematically apply the highest “value” demands to the system. This application process would need its own logic, as the method of determining the best slices to apply each connection would be its own problem.

A second option to the heuristic approach would be to implement the same M-ILP logic to a branch-and-price algorithm as shown in Klinkowski, et. al [25]. This method has been shown to have in increase in performance over the branch-and-bound searching algorithm currently implemented in this thesis. Although this algorithm may be more efficient, it is still inherently a M-ILP; thus it has the same NP-complete nature of the current problem. Larger test systems may be able to be solved but eventually the exponential growth of the runtime will cause large system implementation to be infeasible.

All testing in this thesis was done with respect to the performance metric – percent demand throughput. While this is likely the most important performance metric to consider, as PDR is designed specifically to increase throughput of a system, there are other options for objective function definitions. The system could be redefined with a bi-criterion performance metric that could prioritize percent demand throughput, while also adding some weight to demands that are fully met. Alternatively, analysis could be performed with respect to percent bandwidth usage across all edges in the system. With size-restricted systems such as the ones tested in this thesis,
percent bandwidth utilization could be used as a secondary criterion to indicate how little fragmentation was left in the system.

If large test systems are able to be implemented, further exploration of the topology testing done in this thesis would improve significantly. Larger number of frequency slices would allow for more complex topologies to be tested. Without major runtime restrictions, testing can be done with many different source nodes in the graph, allowing for the bottleneck issues created in the ring and binary tree topologies to be lessened. This could create an interesting area to explore, as the subset of edges in the graph which receive close to maximum amounts of frequency slice allocation would be significantly higher, thus creating an environment which is ideal for the PDR algorithm.

Another expansion area of this thesis reliant upon increasing system size capabilities would be addition of candidate trees to the algorithm. Instead of supplying one routing table for the algorithm, multiple routing options can be given which allow for the algorithm to choose the routing option which best benefits the system with respect to the optimization function. Since candidate trees add a cost in runtime, they were dropped from this thesis. However, there is significant value to the routing options provided if the system size was not as restricted as this testing system.

All of the above future options could then be implemented in a static scheduling problem or a dynamic problem. This would allow demand durations to be incorporated that would result in different system states at various times. This would cause fragmentation at times after the first connection is established. The key purpose of PDR is to allow for added flexibility to be given to the spectral allocation problem. Fragmented scheduled or dynamic systems would be a very interesting area to apply PDR.
Chapter 9 - Conclusions:

Partial destination resolution allows for increased utilization of less than optimal subsections of spectral space to allow for network systems to utilize space that was previously unusable at specific times. This added flexibility allows for an increased percent demand throughput. All test systems that are solvable to optimal values show that PDR allows for an increase in performance with respect to percent demand throughput. As test systems grow beyond optimally solvable system sizes, increased performance trends continue to show until a point where increased complexity of the system outweighs the benefits of PDR. At this point, the exponential growth of mixed-integer linear programming methods become the major problem. To solve this, heuristic algorithms must be developed to allow for solving of the spectral assignment problem in polynomial time.

Small testing systems not solved to optimal still show trends which mirror the results of the optimally solved PDR systems. This indicates that solutions are able to be found in a small time period and a significant portion of the feasible region that is being searched. As testing systems grow in size and complexity, smaller subsections of the feasible region are searched, resulting in much less reliable outputs from the testing system.

The largest contributor to the increased runtime of the PDR systems appears to be the PDR ratio constant. The lower the value for PDR ratio is set, the more possible solutions exist within the feasible region. This greatly increases the complexity of the system and slows the search algorithm’s progress when exploring the entire feasible region. This growth is exponential, adding an additional significant growth factor to the runtime in addition to the NP-complete searching algorithm used for M-ILP solving.

For real world implementation of PDR, mixed integer linear problem solving techniques are not viable. As test systems ranging in size from 6 to 20 nodes, demand, and frequency slices are already causing runtime problems and inconsistency for results, real-world systems of thousands of nodes and demands would create a problem where no solution would be found in nearly all
problems, as the feasible region would be so large that the exponential runtime of this NP-complete system would grow so rapidly that it not feasible to be used.

The next step in adapting PDR to a real-world environment would be to develop a heuristic approach which incorporates the logic of the M-ILP formulation into a polynomial runtime solving algorithm. This spectral assignment problem could then be used in combination with a routing algorithm to be implemented to large scale test system.
Appendix:

Appendix 1: Node/Edge Numbering Scheme:
Appendix 2: Percent Bandwidth Utilization Organizational Scheme:
Appendix 3: OPL Program for Final Formulation:

().'/*********************************************
* OPL 12.6.3.0 Model
* Author: rushaj
* Creation Date: Mar 15, 2016 at 1:46:09 AM
*********************************************/

// System Parameters

float MinPercentBandwidthToConnect = 1;

// Declare Ranges

range Nodes = 0..11;
range Demands = 0..5;
range FrequencySlices = 0..5;
range FrequencySlicesBuffered = 0..8;  // Wavebands + Range for buffer

int DRMin[Demands] = [0,0,0,0,0];  // Must be Zero
int DRMax[Demands] = [3,3,3,3,3];

// Alloc/Init Constants (Values from .dat file)

int DestinationMatrix[Demands][Nodes] = ...;
int DemandMatrix[Demands][Nodes] = ...;
int ParentBoolMatrix[Nodes][Nodes] = ...;

// Decision Variables

dvar boolean DVAR_Utilization[Nodes][Demands][FrequencySlices];
//dvar boolean DVAR_DemandMet[Demands];
dvar boolean DVAR_Connectivity[Nodes][Demands][FrequencySlicesBuffered];
dvar boolean DVAR_bandStartIndex[Demands][FrequencySlices];
dvar boolean DVAR_bandStartIndexWithNodes[Nodes][Demands][FrequencySlices];
dvar int+ DVAR_Utility_BandWidthsInBandrange[Nodes][Demands][FrequencySlices];
dvar float+ DVAR_Utility_PercentDemandThroughput;
dvar int DVAR_maxBW[Nodes][Demands][FrequencySlices];

// Objective Function

maximize
      DVAR_Utility_PercentDemandThroughput;

// Constraints

subject to {

    // Overlap Constraint - Waveband on Node only assigned once.
    forall (n in Nodes, w in FrequencySlices)
    sum(d in Demands) DVAR_Utilization[n][d][w] <= 1;

}
// Connectivity Constraints

// A node can only be connected on d, w if all parent edges are connected on d, w.

forall (n in Nodes, d in Demands, w in FrequencySlices)
  DVAR_Connectivity[n][d][w] <= (sum(m in Nodes)
    DVAR_Utilization[m][d][w] * ParentBoolMatrix[n][m])
  / (sum(m in Nodes) ParentBoolMatrix[n][m]);

// Wavebands in Buffer Range = 0;
forall (n in Nodes, d in Demands, w in FrequencySlicesBuffered)
  if (w > max(w2 in FrequencySlices) w2) DVAR_Connectivity[n][d][w] <= 0;

// Determine bandStartIndex - Store starting index of bandranges used on n, d
forall (n in Nodes, d in Demands, w in FrequencySlices)
  DVAR_bandStartIndexWithNodes[n][d][w] <=
    sum(r in DRMin[d]..(ftoi(ceil(((DRMax[d]+1)/MinPercentBandwidthToConnect-1)))))
    DVAR_Connectivity[n][d][w+r] / (ftoi(ceil((DRMax[d]+1)*MinPercentBandwidthToConnect)DRMin[d]));

// A Bandrange for Demand d starts at w if all nodes in that demand start at w. Else not connected (= 0 ).
forall (d in Demands, w in FrequencySlices)
  DVAR_bandStartIndex[d][w] <= sum (n in Nodes)
    (DVAR_bandStartIndexWithNodes[n][d][w]*DemandMatrix[d][n])
    / (sum (m in Nodes) DemandMatrix[d][m]);

// Continuity/Contiguity Constraints:

forall (n in Nodes, d in Demands, w in FrequencySlices)
  DVAR_maxBW[n][d][w] <= (DRMax[d]+1)*DVAR_bandStartIndex[d][w];

forall (n in Nodes, d in Demands, w in FrequencySlices)
  DVAR_maxBW[n][d][w] <= (DRMax[d]+1)*DVAR_Utility_BandWidthsInBandrange[n][d];

forall (n in Nodes, d in Demands, w in FrequencySlices)
  DVAR_maxBW[n][d][w] <= max(size in ftoi(ceil(((DRMax[d]+1)*MinPercentBandwidthToConnect)-1)).DRMax[d])
    (sum(index in 0..size)(DVAR_Connectivity[n][d][w+index]-((DRMax[d]+1)/(DRMax[d]+2))))*(DRMax[d]+2);
// Utility Definition:

forall (n in Nodes, d in Demands)
    DVAR_Utility_BandwidthsInBandrange[n][d] <= max(w in FrequencySlices)((abs(DVAR_maxBW[n][d][w]) + DVAR_maxBW[n][d][w])/2)*DestinationMatrix[d][n];

forall (x in 0..0)
    DVAR_Utility_PercentDemandThroughput <= sum(n in Nodes,d in Demands) DVAR_Utility_BandWidthsInBandrange[n][d]
    / (sum(n in Nodes,d in Demands) DestinationMatrix[d][n]*(DRMax[d]+1));

}
Appendix 4: Expansion of Results Graph / Confidence Test 2, 3 and 5 Graph
Appendix 5: Confidence Test 1 Graph
Appendix 6: Confidence Test 4 Graph
References:


