Architectural Study of High-Speed Networks with Optical Bypassing

by

Poompat Saengudomlert

B.S.E., Electrical Engineering, Princeton University (1996)
M.S., Electrical Engineering and Computer Science,
Massachusetts Institute of Technology (1998)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2002

Author ....
Department of Electrical Engineering and Computer Science
August 31, 2002

Certified by ....
Eytan H. Modiano
Assistant Professor, Department of Aeronautics and Astronautics
Thesis Supervisor

Certified by ....
Robert G. Gallager
Professor, Department of Electrical Engineering and Computer Science
Thesis Supervisor

Accepted by ....
Arthur C. Smith
Chairman, Department Committee on Graduate Students

Architectural Study of High-Speed Networks

with Optical Bypassing

by

Poompat Saengudomlert

Submitted to the Department of Electrical Engineering and Computer Science
on August 31, 2002, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

We study the routing and wavelength assignment (RWA) problem in wavelength division multiplexing (WDM) networks with no wavelength conversion. In a high-speed core network, the traffic can be separated into two components. The first is the aggregated traffic from a large number of small-rate users. Each individual session is not necessarily static but the combined traffic streams between each pair of access nodes are approximately static. We support this traffic by static provisioning of routes and wavelengths. In particular, we develop several off-line RWA algorithms which use the minimum number of wavelengths to provide l dedicated wavelength paths between each pair of access nodes for basic all-to-all connectivity. The topologies we consider are arbitrary tree, bidirectional ring, two-dimensional torus, and binary hypercube topologies. We observe that wavelength converters do not decrease the wavelength requirement to support this uniform all-to-all traffic.

The second traffic component contains traffic sessions from a small number of large-rate users and cannot be well approximated as static due to insufficient aggregation. To support this traffic component, we perform dynamic provisioning of routes and wavelengths. Adopting a nonblocking formulation, we assume that the basic traffic unit is a wavelength, and the traffic matrix changes from time to time but always belongs to a given traffic set. More specifically, let N be the number of access nodes, and $\mathbf{k}$ denote an integer vector $[k_1, k_2, ..., k_N]$. We define the set of $\mathbf{k}$-allowable traffic matrices to be such that, in each traffic matrix, node i, $1 \le i \le N$, can transmit at most k_i wavelengths and receive at most k_i wavelengths. We develop several on-line RWA algorithms which can support all the $\mathbf{k}$-allowable traffic matrices in a rearrangeably nonblocking fashion while using close to the minimum number of wavelengths and incurring few rearrangements of existing lightpaths, if any, for each new session request. The topologies we consider are the same as for static provisioning. We observe that the number of lightpath rearrangements per new session request is proportional to the maximum number of lightpaths supported on a single wavelength. In addition,

we observe that the number of lightpath rearrangements depends on the topological properties, e.g. network size, but not on the traffic volume represented by $\mathbf{k}$ as we increase $\mathbf{k}$ by some integer factor.

Finally, we begin exploring an RWA problem in which traffic is switched in bands of wavelengths rather than individual wavelengths. We present some preliminary results based on the star topology.

Thesis Supervisor: Eytan H. Modiano
Title: Assistant Professor, Department of Aeronautics and Astronautics

Thesis Supervisor: Robert G. Gallager
Title: Professor, Department of Electrical Engineering and Computer Science

Acknowledgments

First and most importantly, I would like to thank my two thesis advisors, Professor Eytan Modiano and Professor Robert Gallager, for making my learning experience at MIT truly memorable. I feel extremely fortunate to have had a chance to work with them. Without their guidance and patience, this thesis would not be possible.

Professor Vincent Chan has been very kind to me by spending a lot of time on various discussions. I am grateful for his support. I also want to thank my past advisors, Professor Muriel Médard and Professor Amos Lapidoth, for teaching me how to do research in the early stages of my graduate study.

Several fellow graduate students created the friendly and highly intellectual atmosphere around MIT for me. I would like to recognize several of them: Chalee Asavathiratham, Randall Berry, Serena Chan, Li-Wei Chen, Aaron Cohen, Todd Coleman, Alvin Fu, Angelia Geary, Chi (Kyle) Guan, Shane Haas, Tracey Ho, Ramesh Johari, Thierry Klein, Emre Koksal, Julius Kusuma, Nicholas Laneman, Chunmei Liu, Thit Minn, Michael Neely, Asuman Ozdaglar, Natanael Peranginangin, Brett Schein, Karin Sigurd, Jun Sun, Poonsaeng Visudhiphan, Frank Wang, Hung-Jen Wang, Guy Weichenberg, Edmund Yeh, and Won Yoon.

My stay in Edgerton House was a lot of fun thanks to my roommates Yu-Han Chang and Lawrence (Kai) Shih who made our apartment feel like home to me. In Edgerton House, I also enjoyed the company of Stephane Bratu, Pei-Lin Hsiung, Eden Miller, Minghao Qi, Sarah Rhee, and Chatchai Unahabhokha.

My three high-school classmates from Thailand, Ariya Akthakul, Chalee Asavathiratham, and Poonsaeng Visudhiphan, joined me at MIT and gave me a lot of moral support for my research. Besides them, I am grateful to know several other Thai students at MIT: Pongpun Amornvivat, Sutapa Amornvivat, Siwaphong Boonsalee, Yot Boontongkong, Siriwut Buranapin, Virat Chatdarong, Nuwong Chollacoop, Acharawan Chutarat, Thunyachate Ekvetchavit, Kachaphol Harinsuit, Singh Intrachooto, Jen Jootar, Thanisara Kiatbaramee, Watjana Lilaonitkul, Pimpa Limthongkul, Manoj Lohatepanont, Narintr Narisaranukul, Mukaya Panich, Joshua Pas, Pitiporn Phanaphat, Piyada Phanaphat, Pradya Prempraneerach, Taweechai Pureetip, Thirapun Sanpakit, Thapanee Sirivadhanabhakdi, Paisarn Sonthikorn, Atiwong Suchato, Suchatvee Suwansawat, Isra Taulananda, Ratchatee Techapiesancharoenkij, Yunyong Thaicharoen, Chayakorn Thanomsat, Ariyasak Thepchatri, and Warit Wichakool.

Throughout my study, I received constant love and supports from my beloved family and my future wife Chaliga Vanduangden. I would like to dedicate this thesis to them.

Contents

Chapter 1

Introduction

1.1 Optical Bypassing in All-Optical Networks

Optical fiber is a communication medium with a potential transmission bandwidth up to 25 THz [Gre93]. Practical networks employ wavelength division multiplexing (WDM) in which the fiber bandwidth is divided into multiple frequency bands often called wavelengths. In current practical WDM systems, only a portion of the fiber bandwidth is utilized. In addition, the highest transmission rate over a single wavelength is 40 Gbps, whereas the total transmission rate over multiple wavelengths in a single fiber is currently beyond 1 Tbps [RS01].

While processing WDM traffic electronically at every network node may be technologically feasible, it yields a very expensive network architecture. Electronic processing at every node was adopted in the early days of communication networks, e.g. the ARPANET, when the cost of transmission dominated the cost of processing at all the network nodes. However, for current high-speed networks with optical transmission technology, we expect the cost of electronic information processing to dominate the cost of optical information transmission. Therefore, it is desirable to eliminate unnecessary electronic processing in the network. For example, consider the scenario in figure 1-1. There are two sources, each sending one wavelength worth of traffic to the destination. The wavelength from source 1, denoted by λ_1, can be combined with the wavelength from source 2, denoted by λ_2, using an optical multiplexer without any electronic processing. In this case, we say that the traffic session from source 1 *optically bypasses* electronic processing at node 2.

In the given example, the use of optical bypassing requires no more wavelengths than that

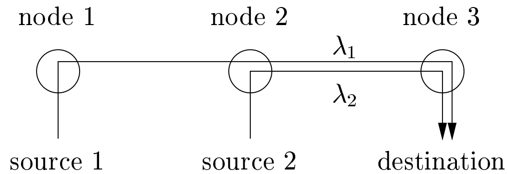


Figure 1-1: Example network to illustrate optical bypassing.

required by electronic processing. However, this is not always the case. Suppose for example that each source in figure 1-1 transmits only half a wavelength worth of traffic to the destination. It follows that, with optical bypassing of traffic from source 1 at node 2, i.e. no multiplexing of traffic at the subwavelength level, we need to utilize two wavelengths on the link from node 2 to node 3. If we use an electronic switch at node 2 to multiplex the two traffic streams, then only a single wavelength is required. Thus, optical bypassing may require more wavelengths when the bypassed traffic sessions have smaller rates than the rate of a single wavelength. Despite additional required wavelengths, the cost savings from the elimination of electronic processing could still be attractive enough to justify optical bypassing.

In an *all-optical network* architecture, each traffic session optically bypasses electronic processing at all intermediate nodes, i.e. nodes that are neither the source nor the destination of that session. In other words, there is no electronic reception and retransmission of data packets by any intermediate node. We shall concentrate on all-optical network architectures in this thesis.

Optical wavelength changers allow us to change the wavelength of a traffic session at intermediate nodes without electronic processing. Since optical wavelength changers are very expensive, we shall assume no optical wavelength conversion except when explicitly indicated. With this assumption, each optically bypassed traffic session is subjected to the *wavelength continuity constraint*, which dictates that the session must travel on the same wavelength on all links from the source node to the destination node. For a given traffic session, define its *lightpath* to be the route and the wavelength used to support that session. There are usually multiple ways to assign a lightpath for a given session. The problem of assigning lightpaths for all traffic sessions in the network is called the *routing and wavelength assignment (RWA) problem*, which is the main topic of this thesis.

We have seen an example in which optical bypassing increases the required number of wavelengths in a fiber when the rates of bypassed traffic sessions are smaller than one wavelength unit. The following example shows that, even when the rate of each session is equal to one wavelength, optical bypassing may require additional wavelengths in a fiber due to the wavelength continuity constraint. For example, consider the scenario in figure 1-2. The rate of each session is one wavelength. Without optical bypassing, the required number of wavelengths in each fiber is equal to the maximum link load which is two wavelengths in this example. On the other hand, with optical bypassing, we need three wavelengths because each lightpath necessarily shares a transmission link with each of the other two lightpaths and thus needs a distinct wavelength. Notice that two wavelengths suffice in this example if wavelength changers are employed.

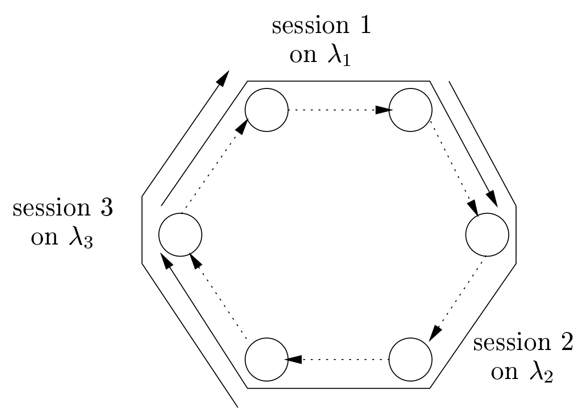


Figure 1-2: Increase in the number of wavelengths due to the wavelength continuity constraint.

In short, optical bypassing serves as an approach to reduce the cost of electronic processing of information at the network nodes, but possibly at the cost of additional wavelengths. In fact, optical bypassing can be viewed as a special case of the general trade-off between switching and transmission costs in communication networks. What motivates us in this special case is the potential of a significant reduction in switching cost with only a slight increase in transmission cost.

1.2 Reconfigurable Switching Node Model

Our generic model of a reconfigurable switching node is illustrated in figure 1-3. Traffic sessions on each input fiber are separated by an optical demultiplexer (DMUX). The wavelengths (and hence the traffic sessions) on the same wavelength from different input fibers go through a reconfigurable optical switch dedicated to that wavelength. Such a switch is called a wavelength selective switch.

Each wavelength selective switch is subjected to the *crossbar constraint*, which dictates that no more than one input (output) can be connected to a single output (input). Traffic sessions on different wavelengths switched to the same output fiber are combined by an optical multiplexer (MUX). Some input sessions are terminated or dropped to the end users or the subnetwork connected to this network node. Similarly, some output sessions are transmitted or added from the end users or the subnetwork.

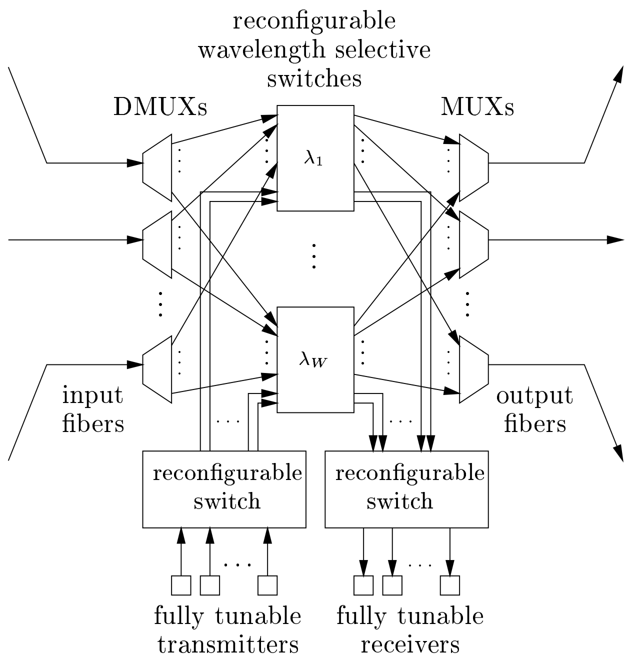


Figure 1-3: Reconfigurable switching node model.

We shall assume that optical transmitters and receivers are *fully tunable*, i.e. a single transmitter (receiver) can be used to transmit (receive) on any wavelength in the fiber. When possible, we shall discuss how this assumption can be relaxed. The use of tunable transmitters and receivers requires additional optical switches in order to guide the transmitted and received wavelengths to appropriate optical switches, as illustrated in figure 1-3.

Certain wavelengths may be used to provide dedicated static connections. For these wavelengths, the transmitters and receivers need not be tunable. In addition, we can replace reconfigurable wavelength selective switches with fixed wavelength selective switches. Figure 1-4 shows a reconfigurable switching node model in which a subset of wavelengths, denoted by λ_{V+1}, ..., λ_W, are used for dedicated static connections. While this node model is less flexible than the one in figure 1-3, it can provide cost savings from the smaller number of reconfigurable components.

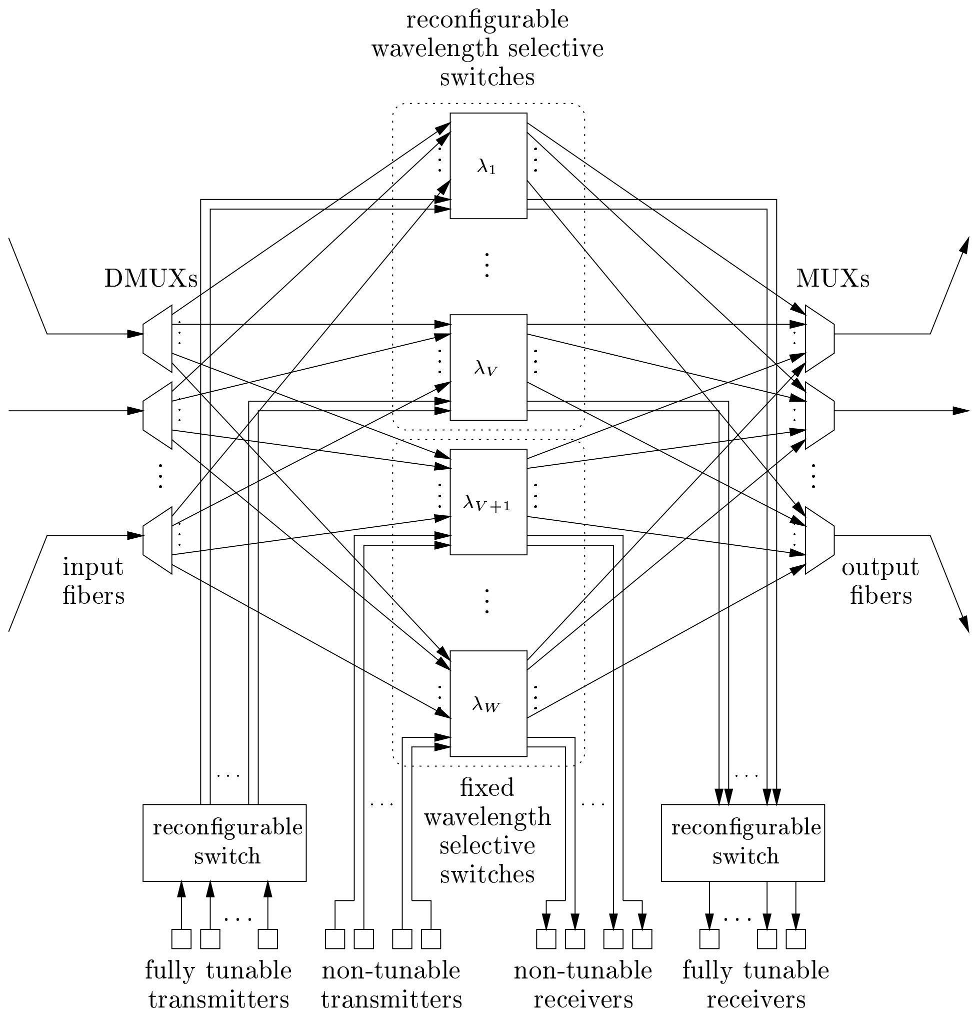


Figure 1-4: Reconfigurable switching node model in which wavelengths λ_{V+1}, ..., λ_W are used for dedicated static connections.

1.3 Switching of Traffic in Larger Granularities

As the amount of traffic among network nodes increases, it is more efficient to switch traffic in larger and larger traffic units. More specifically, we expect to switch traffic in units of wavelengths, bands of wavelengths, fibers, bundles of fibers, and so on. At each increment of the traffic unit, there is a potential cost saving from bypassing the processing of traffic in the smaller unit at intermediate nodes. For example, if we expect a wavelength to be a common traffic unit, then we can bypass electronic processing of traffic at intermediate nodes, possibly at the price of more wavelengths. If we expect a band of wavelengths to be a common traffic unit, then we can bypass wavelength-level optical MUXs and DMUXs, i.e. use only band-level optical MUXs and DMUXs, possibly at the price of more bands of wavelengths. Notice that, for different increments of the traffic unit, the logical problems of how to efficiently bypass the processing of traffic in the smaller unit are similar. The main differences lie first in a common traffic unit, and second in an available switching technology for that unit. By an appropriate scaling of the common traffic unit, a solution for the bypassing problem with one common traffic unit may be used for the bypassing problem with another common traffic unit.

However, the trade-offs between the reduction in switching cost and the increase in wavelengths can differ greatly in the bypassing problems with different common traffic units. For example, when a wavelength is a common traffic unit, optical bypassing of electronic processing can offer a significant saving in switching cost at a relatively small price of more wavelengths. On the other hand, when a band of wavelengths is a common traffic unit, bypassing of wavelength-level optical processing can reduce wavelength-level optical MUXs, DMUXs, and reconfigurable switches but may or may not justify a price of more bands of wavelengths. The detailed nature of these trade-offs are beyond the scope of this thesis. For the most part, we shall concentrate on the cases in which a wavelength is a common traffic unit and investigate how to switch wavelengths of traffic efficiently. In the last part of this thesis, we shall explore how to efficiently switch traffic in bands of wavelengths.

1.4 Core Network with Aggregated Traffic

We shall focus our attention on the design of a high-speed core network that interconnects subnetworks using electronic switches at the access nodes. Figure 1-5 shows an example of such a core network. With respect to the core network, the access nodes act as entry and exit points for traffic from individual end users in the subnetworks. The core network may have nodes that are not access nodes but are used to switch traffic. Electronic switches at the access nodes can be used to aggregate and deaggregate small-rate traffic sessions from individual end users in subnetworks. For large-rate sessions whose rates are approximately a wavelength, electronic switches can act as electronic wavelength changers which relax the wavelength continuity constraint between a core network link and a subnetwork link on a given lightpath.

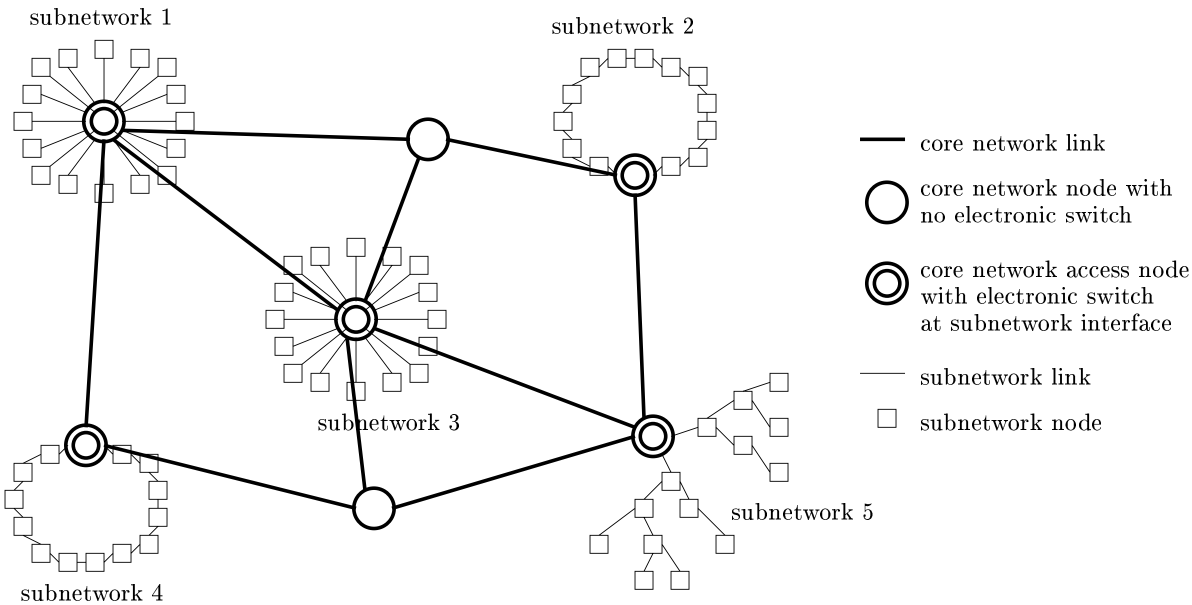


Figure 1-5: A core network interconnecting subnetworks through electronic switches at the access nodes.

In the core network, each traffic session is transmitted from one access node, referred to as the source node, to another access node, referred to as the destination node. A single session may result from traffic aggregation of a large number of small-rate sessions in a subnetwork. In this case, we expect each traffic session to be somewhat static and shall provide its route and wavelength in a static fashion. On the other hand, a single session may result from traffic aggregation of few large-rate sessions or even from a single large-rate session in a subnetwork. In this case, sessions might have short lifetimes, so it is necessary to change routes and wavelengths in a dynamic fashion. For the purpose of RWA algorithm designs, we can consider static provisioning of routes and wavelengths as if we were to support static traffic sessions. Throughout the thesis, we shall use the terms *static traffic* and *dynamic traffic* to refer to the cases in which we perform static and dynamic provisioning of routes and wavelengths respectively, even though each supported session is not static under static provisioning.

We shall adopt all-optical network architectures and aim to develop RWA algorithms to support both static and dynamic traffic in the core network. Note that, in an all-optical network, each traffic session is electronically processed only at the source node and the destination node. In both static and dynamic traffic models, we assume that each session has a rate equal to one wavelength unit. This assumption is reasonable for the design of high-speed core networks in which each pair of subnetworks have multiple wavelengths of traffic to communicate. In addition, this assumption allows us to neglect the additional wavelengths required for optical bypassing due to the traffic sessions whose rates are smaller than a wavelength.

1.5 Outline of the Thesis

The remaining parts of this thesis are organized as follows. Chapter 2 briefly discusses existing literature on the RWA problem in WDM networks. It also states our thesis objectives and presents our problem formulations. Chapter 3 discusses static RWA and presents our RWA algorithms for static traffic. Chapter 4 discusses dynamic RWA and presents our RWA algorithms for dynamic traffic. Chapter 5 explores further reduction in switching cost by performing switching in bands of wavelengths instead of in wavelengths. Finally, chapter 6 summarizes our achievements and points out some directions for future research.

Chapter 2

History and Problem Formulations

2.1 Existing Literature on the RWA Problem

Several papers investigate the routing and wavelength assignment (RWA) problem in a wavelength division multiplexing (WDM) network under the wavelength continuity constraint. A comprehensive overview of different problem formulations and solution approaches taken by researchers is available in [YB97, ZJM00]. We can categorize existing results into two groups based on whether static or dynamic provisioning of routes and wavelengths is performed. For static provisioning, the traffic to be supported is assumed known and fixed over time. The goal is often to minimize the number of wavelengths used in the network [BM96, RS96]. Alternatively, if the number of wavelengths is fixed in advance, one goal is to maximize the number of supported traffic sessions according to some known and fixed traffic demands [CGK92, RS95, ZA95, CB96]. These problems can be formulated as mixed integer linear programming (ILP) problems [RS95, ZA95, BM96, CB96, RS96], which are known to be NP-complete [CGK92]. Consequently, the RWA problems are frequently divided into two steps, the first for routing and the second for wavelength assignment. These two steps are then solved separately and suboptimally. In some cases, partial routing decisions are made at the time of wavelength assignment. For example, an RWA algorithm may assign a few routes in advance for each session with the final choice to be made at the time of wavelength assignment [RS95]. For some regular topologies and specific traffic, e.g. all-to-all uniform traffic in the bidirectional ring topology, the overall RWA problem can be solved to obtain closed form solutions [Elr93, Wil96]. For arbitrary mesh topologies, bounds on the optimal costs have been derived [RS95, BYC97] and several heuristics have been developed [RS95, ZA95, BM96, CB96, LL96, Muk+96].

Dynamic provisioning of routes and wavelengths gives us flexibility in supporting traffic which may change over time through session arrivals and session departures. To model dynamic traffic, session arrivals can be assumed to form stochastic processes [Bir96, SAS96]. In addition, session lifetimes are stochastic. The goal is usually to develop an on-line RWA algorithm which minimizes the average blocking probability for a new session request given a fixed number of wavelengths in the network. We refer to this type of problem formulation as the *blocking formulation*. Due to the complexity in computing blocking probabilities, some approximations are made to simplify the analysis. For example, session arrivals on different links are assumed to be independent [Bir96, BH96], or correlated among adjacent links in the same fashion throughout the network [SAS96]. Based on such approximations, several dynamic RWA heuristics are developed [LS99, ZRP00].

Another type of problem formulation, referred to as the *nonblocking formulation*, assumes prior knowledge of the set of all the traffic matrices, or equivalently the traffic demands, to be supported [Pan92, Ger+99, NLM02]. In [Ger+99], the set of traffic matrices is characterized by the maximum link load in the network. In [Pan92, NLM02], the set of traffic matrices is characterized by the number of tunable transmitters and tunable receivers at each end node. A new session is said to be *allowable* if its arrival results in a traffic matrix which is still in the set of supportable traffic. The goal is usually to develop an on-line RWA algorithm which does not block any allowable session and uses the minimum number of wavelengths.

If we allow some existing lightpaths to be rearranged in order to support a new session, the corresponding RWA algorithm is said to be *rearrangeably nonblocking*.[1] If we allow no rearrangement of any existing lightpath in order to support a new session, the corresponding RWA algorithm is said to be *wide-sense nonblocking*. Note that if an RWA algorithm is wide-sense nonblocking, it is also rearrangeably nonblocking. Therefore, for the same set of traffic matrices, the required number of wavelengths is higher for a wide-sense nonblocking RWA algorithm than for a rearrangeably

[1]The terminology comes from standard definitions in switching theory. A switching network is *rearrangeably nonblocking* if any allowable session can be supported, possibly after some rearrangements of existing sessions. A switching network is *wide-sense nonblocking* if any allowable session can be supported without rearrangement of existing sessions provided that all the existing sessions have been routed according to some algorithm. Finally, a switching network is *strict-sense nonblocking* if any allowable session can be supported without rearrangement of existing sessions. Notice that, in a strict-sense nonblocking network, we can support each allowable session by choosing any of the routes available at the time. By definition, a strict-sense nonblocking network is also wide-sense nonblocking. In addition, a wide-sense nonblocking network is also rearrangeably nonblocking.

nonblocking RWA algorithm.

Switching of traffic in multiple levels of granularity appears in several investigations on the *traffic grooming problem*. In the traffic grooming problem, the objective is to efficiently aggregate small-rate traffic sessions onto wavelengths using electronic switches and to perform optical bypassing to minimize the cost of electronic switches [BM00, CM00, GRS00]. Similar problems exist for larger levels of traffic granularity. In particular, as traffic demands increase, we expect to reduce the switching cost further by switching traffic in bands of wavelengths instead of in wavelengths when it is appropriate. In this case, the cost savings come from the reduction of optical switching resources. For convenience, we shall refer to a switch whose basic traffic unit is a wavelength as a *wavelength switch*. Accordingly, we shall refer to a switch whose basic traffic unit is a band of wavelengths as a *band switch*. In addition, we shall refer to the RWA problem with wavelengths and bands of wavelengths as the two levels of traffic granularity as the *band/wavelength RWA problem*.

Despite their similarities, there are some fundamental differences between the traffic grooming problem and the band/wavelength RWA problem. Since we still operate in the optical domain, the wavelength continuity constraint applies at the interface between a band switch and a wavelength switch, whereas there is no such constraint at the electronic interface. In addition, the cost structure of an optical switch is different from that of an electronic switch. More specifically, the cost of an electronic switch primarily depends on the total input traffic rate, while the cost of an optical switch may only depend on the total number of input ports. For example, with promising microelectromechanical system (MEMS) technologies, an optical switch can be constructed from a set of tiny mirrors used to reflect traffic streams in the form of light beams from input ports to output ports [RS01]. Such an optical switch can be used as a band switch or a wavelength switch without significant cost difference.

2.2 Thesis Objectives

In this thesis, we consider the RWA problem in a WDM network under the wavelength continuity constraint for both static and dynamic traffic. By static traffic, we refer to static provisioning of routes and wavelengths for traffic sessions. In a high-speed core network, such static provisioning of resources can be used to support aggregated traffic in which each individual session is not necessarily static but the combined traffic streams between each pair of access nodes are approximately static. By carefully choosing the locations of access nodes and the sizes of their corresponding subnetworks, we may be able to form a core network such that aggregated traffic streams among the access nodes are somewhat uniform. Such uniformity of traffic may not be achievable in practice. Nevertheless, we are interested in the case of providing one or a few wavelength paths between each pair of access nodes for basic all-to-all connectivity. In addition, having these dedicated wavelength paths between all pairs of nodes can simplify network operations since most small-rate sessions can be supported on dedicated paths and there is rarely a need to reconfigure the switching nodes as a result of a small traffic change. We view this static provisioning of routes and wavelengths as if we were to support static uniform all-to-all traffic. Our goal is to develop an off-line RWA algorithm which uses the minimum number of wavelengths for static uniform all-to-all traffic.

On the other hand, by dynamic traffic, we refer to dynamic provisioning of routes and wavelengths for traffic sessions. In a high-speed core network, dynamic provisioning of routes and wavelengths can be used to support traffic streams which cannot be well approximated as static due to insufficient aggregation. Adopting the nonblocking formulation, we assume that the traffic matrix changes from time to time but always belongs to a known traffic set. Our goal is to design an on-line RWA algorithms which can support all the traffic matrices in the known traffic set in a rearrangeably nonblocking fashion while using the minimum number of wavelengths and incurring few rearrangements of existing lightpaths, if any, for each traffic change.

Instead of trying to solve the RWA problem for an arbitrarily given network topology, we aim to investigate what topological properties contribute to good network architectures. To do so, we formulate RWA problems in a tractable fashion so that efficient solutions can be analytically derived. It is our hope that some of the analytical techniques developed in this thesis can contribute to greater understanding of network architectures. To build an analytical framework, we consider a few specific topologies including an arbitrary tree, a bidirectional ring, a two-dimensional (2D) torus, and a binary hypercube. Notice that these topologies are listed from the least densely connected to the most densely connected.

In the last part of this thesis, we perform preliminary study of the band/wavelength RWA problem in a WDM network under the wavelength continuity constraint. Our goal is to understand when and how individual wavelengths should be aggregated into bands of wavelengths to reduce the cost of optical switching. We present a two-level hierarchical network topology in which the top-level network nodes switch traffic in bands of wavelengths and the lower-level network nodes switch traffic in wavelengths.

In the remaining sections, we formulate in detail the static and the dynamic RWA problems of interest in this thesis.

2.3 RWA Problem for Static Traffic

This section formulates the RWA problem for static traffic. This problem is investigated in detail in chapter 3. Consider an all-optical WDM network with no optical wavelength conversion. In any given network topology, assume that adjacent nodes are connected by two fibers, one in each direction. Assume also that all fibers contain the same number of wavelengths, i.e. WDM channels. We shall refer to a network node which sources and sinks traffic as an *end node*. Let N be the number of end nodes in the network. In the context of a core network, an end node corresponds to an access node. Notice that there may be some network nodes which are not end nodes, e.g. a switching hub node in the star topology.

Define *l-uniform traffic* to be static traffic in which each end node transmits l wavelengths to, and receives l wavelengths from, each of the other end nodes.[2] Note that l-uniform traffic requires $l(N-1)$ transmitters and $l(N-1)$ receivers at each end node. Since the traffic is static, these transmitters and receivers need not be tunable. Moreover, at each switching node, we can use fixed optical switches instead of reconfigurable optical switches. The RWA problem for l-uniform traffic is given below.

Problem 1 (Off-Line RWA for l-Uniform Traffic) For a given network topology with N end nodes, let $W_{s,l}$ denote the minimum number of wavelengths which, if provided in each fiber, can support l-uniform traffic with no wavelength conversion. We want to find the value of $W_{s,l}$ and a corresponding off-line RWA algorithm.

In the above problem formulation, we model a traffic stream which is the aggregation of a large number of small-rate sessions as being static. The uniformity of static traffic may not be

[2] We reserve the terms *transmit* and *receive* for the end nodes which source and sink traffic sessions. Intermediate nodes which only switch traffic but neither source nor sink traffic are not considered transmitting or receiving traffic.

realistic. Nevertheless, we consider supporting l-uniform traffic for tractable analysis. In addition, supporting 1-uniform traffic is an interesting problem in how to provide minimal optical all-to-all connectivity among the end nodes.

2.4 RWA Problem for Dynamic Traffic

In this section, we formulate the RWA problem for dynamic traffic. We shall investigate this problem in chapter 4. As in the RWA problem for static traffic, we consider an all-optical WDM network with N end nodes and no optical wavelength conversion. Assume that node i, $1 \leq i \leq N$, is equipped with k_i fully tunable transmitters and k_i fully tunable receivers. At any time, node i can transmit at most k_i wavelengths and receive at most k_i wavelengths. Such a traffic matrix is said to belong to a set of **k**-*allowable traffic*, where $\mathbf{k} = [k_1, k_2, ..., k_N]$. Assume that each traffic session has a rate of one wavelength. We model dynamic traffic as a session-by-session arrival and departure process in which sessions arrive and depart one at a time. In other words, a transition from one traffic matrix to another is a result of either a single session arrival or a single session departure.

A new session request is allowable if the resultant traffic matrix is still in the set of **k**-allowable traffic. The definition implies that, for each allowable session request, there is a free transmitter at the source node and a free receiver at the destination node. We want to design a rearrangeably nonblocking RWA algorithm which can assign a lightpath to any allowable session, perhaps after some rearrangements of existing lightpaths. Our algorithms will be centralized in nature. We assume that traffic does not change too frequently and the RWA algorithms always have correct knowledge of the RWA in the network. In addition, we assume there is sufficient time for lightpath rearrangements between consecutive transitions of the traffic matrix.

Problem 2 (On-Line RWA for k-Allowable Traffic) For a given network topology with N end nodes, let $W_{d,\mathbf{k}}$ denote the minimum number of wavelengths which, if provided in each fiber, can support dynamic **k**-allowable traffic in a rearrangeably nonblocking fashion with no wavelength conversion. We want to find the value of $W_{d,\mathbf{k}}$ and a corresponding on-line RWA algorithm which uses minimal wavelengths and requires few, if any, lightpath rearrangements per new session request.

Note that the set of $\mathbf{k}$-allowable traffic represents the largest set of traffic matrices supportable by the given number of fully tunable transmitters and receivers in $\mathbf{k}$. In practice, past traffic history may suggest that we need to provide network resources only for a strict subset of $\mathbf{k}$-allowable traffic. Nevertheless, we shall concentrate on supporting the entire $\mathbf{k}$-allowable traffic set. It is clear that, for any network, the value of $W_{d,\mathbf{k}}$ is an upper bound on the minimum number of wavelengths required to support any strict subset of $\mathbf{k}$-allowable traffic.

To establish some connection between static and dynamic traffic, consider l-uniform traffic. When all the k_i's are equal to $l(N-1)$, l-uniform traffic belongs to the set of $\mathbf{k}$-allowable traffic. It follows that $W_{s,l} \leq W_{d,\mathbf{k}}$ in this case. In addition, a given dynamic RWA algorithm can be used to support l-uniform traffic. However, the number of wavelengths used by the algorithm will be higher than necessary.

Chapter 3

RWA for Static l-Uniform Traffic

In this chapter, we study the routing and wavelength assignment (RWA) problem for l-uniform traffic. In l-uniform traffic, each end node transmits l wavelengths to and receives l wavelengths from each of the other end nodes. While our goal includes understanding arbitrary mesh topologies, we solve the RWA problem in a few special cases. The specific topologies we shall consider are arbitrary tree topologies, a bidirectional ring, a two-dimensional (2D) torus, and a binary hypercube. Let $W_{s,l}$ denote the minimum number of wavelengths which, if provided in each fiber, can support l-uniform traffic with no wavelength conversion. In the future, we aim to extend our analytical techniques to obtain a good bound on the value of $W_{s,l}$ for any given topology.

Let $L_{s,l}$ denote the minimum number of wavelengths in a fiber required to support l-uniform traffic given full wavelength conversion at all network nodes. It is clear that $L_{s,l} \leq W_{s,l}$ for any given topology. We shall see that, in all the network topologies for which we can obtain the closed form expressions for $W_{s,l}$ and $L_{s,l}$, we can perform RWA efficiently to achieve $W_{s,l} = L_{s,l}$ without any wavelength converter in the network.

3.1 Arbitrary Tree Topologies

In this section, we solve the RWA problem for l-uniform traffic in an arbitrary tree topology. In a given tree topology, we assume there are $N > 2$ end nodes which are the leaf nodes of the tree.[1] We describe a tree by a set of nodes $\mathcal{N}$ and a set of bidirectional links $\mathcal{T}$. For the purpose of RWA,

[1] The RWA problem for a tree with two leaf nodes is trivial.

we can assume that each non-leaf node has degree at least 3.[2] Note that if a non-leaf node has degree less than 3, then it can be removed from the tree without changing the RWA problem, as illustrated in figure 3-1. Since there is a unique route for each traffic session, there is no routing problem in a tree topology. Thus, we only have to perform wavelength assignment (WA) in the RWA problem.

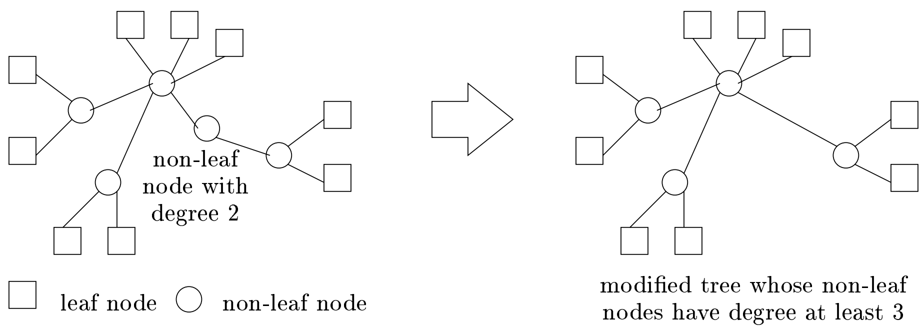


Figure 3-1: Removal of a non-leaf node with degree less than 3.

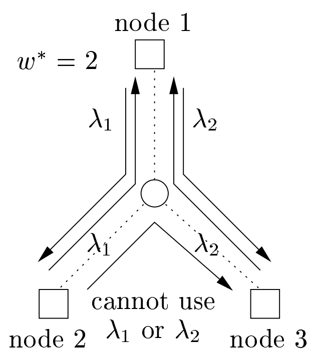
Let us consider the WA problem for 1-uniform traffic. The results are later extended, in a straightforward manner, to l-uniform traffic. Let $L_{s,1}$ denote the minimum number of wavelengths which, if provided in each fiber, can support 1-uniform traffic given full wavelength conversion at all nodes. Each link e in the tree corresponds to a cut which separates the N end nodes into two sets, denoted by $\mathcal{N}_{e,1}$ and $\mathcal{N}_{e,2}$. The amount of traffic (in wavelengths) on a fiber across link e is equal to $|\mathcal{N}_{e,1}||\mathcal{N}_{e,2}|$. Let w^* denote the maximum traffic over all the fibers. Clearly, $L_{s,1}$ is equal to w^*, as given below.

$$L_{s,1} \quad = \quad w^* \quad = \quad \max_{e \in \mathcal{T}} |\mathcal{N}_{e,1}||\mathcal{N}_{e,2}| \tag{3.1}$$

Let $W_{s,1}$ denote the minimum number of wavelengths which, if provided in each fiber, can support 1-uniform traffic with no wavelength conversion. We shall show that $W_{s,1}$ is bounded by $W_{s,1} \le w^*$, which implies $W_{s,1} = L_{s,1} = w^*$. We do so by constructing a WA algorithm. Figure 3-2 illustrates an example scenario in which a greedy WA algorithm fails to support 1-uniform traffic using w^* wavelengths. In this example, inspection shows that $w^* = 2$. Note that the same wavelength is assigned to the oppositely directed sessions between the same pair of nodes,

[2]Since we assume that each link consists of two fibers, one in each direction, the indegree and the outdegree of any given network node are the same. We simply refer to their value as the node degree.

e.g. sessions (1,2) and (2,1) on wavelength λ_1. After assigning wavelength λ_1 to sessions (1,2) and (2,1) and wavelength λ_2 to sessions (1,3) and (3,1), neither λ_1 nor λ_2 can be assigned to support session (2,3). It follows that more than $w^* = 2$ wavelengths are required. Therefore, this example scenario tells us that the design of a WA algorithm using w^* wavelengths is not trivial. Figure 3-2 also demonstrates that, in order to use the minimum number of wavelengths, we may need to support the oppositely directed sessions between the same pair of nodes on different wavelengths.



sequence of sessions for WA	corresponding sequence of WA steps
(1,2)	(1,2) on λ_1
(2,1)	(2,1) on λ_1
(1,3)	(1,3) on λ_2
(3,1)	(3,1) on λ_2
(2,3)	(2,3) not on λ_1 or λ_2

(i, j) denotes a session from node i to node j.

Figure 3-2: An example in which a greedy approach requires more than w^* wavelengths.

We now derive a few useful properties related to the minimum number of wavelengths w^*. Let e^* denote the link associated with w^*. Note that there may be multiple choices for e^*. The exact choice does not matter in the following discussion. We shall refer to e^* as the *bottleneck link* since it is the link with the maximum traffic on a fiber. Link e^* separates the leaf nodes into two sets $\mathcal{N}_{e^*,1}$ and $\mathcal{N}_{e^*,2}$. Without loss of generality, choose $\mathcal{N}_{e^*,1}$ such that $|\mathcal{N}_{e^*,1}| \leq |\mathcal{N}_{e^*,2}|$. Since we assume there are more than two leaf nodes, $\mathcal{N}_{e^*,2}$ must contain multiple leaf nodes. Define the *bottleneck node* v^* to be the end point of e^* opposite to $\mathcal{N}_{e^*,1}$, i.e. the subtree connected to v^* by e^* contains all the leaf nodes in $\mathcal{N}_{e^*,1}$, as illustrated in figure 3-3a.

We shall refer to each subtree connected to v^* as a *top-level subtree*. Note that a top-level subtree can be a single node. Figure 3-3b shows the top-level subtrees associated with the tree in figure 3-3a. Let d^* be the degree of v^*. Since v^* is a non-leaf node, $d^* \geq 3$. It follows that there are $d^* \geq 3$ top-level subtrees.

Let $\mathcal{S}_i$, $1 \leq i \leq d^*$, denote the set of all the leaf nodes in top-level subtree i, and $x_i = |\mathcal{S}_i|$. The following lemma provides useful properties of the top-level subtrees connected to v^* as well as bounds on the minimum number of wavelengths w^*.

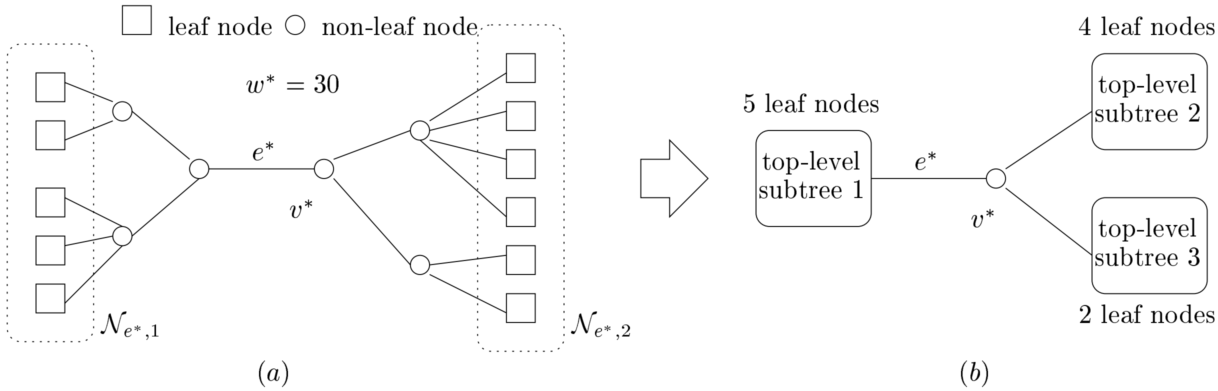


Figure 3-3: The bottleneck link e^* and the bottleneck node v^*.

Lemma 1 *Number the top-level subtree connected to the bottleneck node v^* by the bottleneck link e^* as top-level subtree 1, and the rest of the top-level subtrees from 2 to d^*, where d^* is the degree of v^*. Then,*

1. $x_i \leq x_1 \leq N/2$ *for all* $1 \leq i \leq d^*$, *and*
2. *the minimum number of wavelengths w^* is bounded by*

$$\frac{1}{d^*}\left(1-\frac{1}{d^*}\right)N^2 \quad \leq \quad w^* \quad \leq \quad \frac{N^2}{4}.$$

Proof:

1. Define $f(x) = x(N-x)$. Note that $f(x_i)$ is the traffic (in wavelengths) carried on each of the two fibers between the bottleneck node v^* and top-level subtree i. By the definition of e^*, $f(x_1) \geq f(x_i)$ for all $2 \leq i \leq d^*$. We now prove that $x_i \leq x_1$ for all $2 \leq i \leq d^*$ using contradiction. Assume that $x_i > x_1$ for some $i \neq 1$. Since $d^* \geq 3$, it follows that $x_1 + x_i < N$, yielding $x_i < N - x_1$. As illustrated in figure 3-4, $f(x)$ is concave and symmetric around the maximum value at $x = N/2$. Thus, the relation $x_1 < x_i < N - x_1$ implies that $f(x_i) > f(x_1)$, yielding a contradiction.

   Since top-level subtree 1 contains all the leaf nodes in $\mathcal{N}_{e^*,1}$ and $|\mathcal{N}_{e^*,1}| \leq |\mathcal{N}_{e^*,2}|$, it follows that $x_1 \leq N/2$. We conclude that $x_i \leq x_1 \leq N/2$ for all $1 \leq i \leq d^*$.

2. Note that $f(x) = x(N-x)$ has the maximum value of $N^2/4$ at $x = N/2$, as shown in figure 3-4. Since $w^* = f(x_1)$, it is clear that $w^* \leq N^2/4$. To prove the lower bound, note that

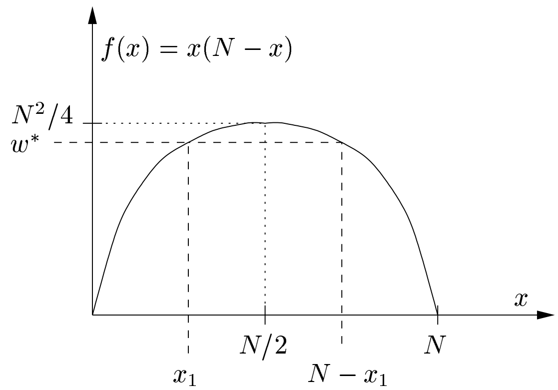


Figure 3-4: Graph of $f(x) = x(N-x)$.

$w^* = f(x_1)$ is an increasing function of x_1 for $0 < x_1 < N/2$. Thus, w^* is minimized when x_1 takes the lowest possible value which is equal to $\lceil N/d^* \rceil$. It follows that

$$w^* \quad \geq \quad f\left(\lceil N/d^* \rceil\right) \quad \geq \quad f\left(N/d^*\right),$$

which is the desired lower bound. □

Before describing our WA algorithm, we describe some of the ideas behind it. Define a *local session* to be a traffic session whose source and destination are in the same top-level subtree. Accordingly, a *non-local session* has its source and its destination in different top-level subtrees. Note that a non-local session has to travel through the bottleneck node v^*, whereas a local session does not have to travel all the way to v^* and back to its destination, i.e. each session never uses the same link twice in opposite directions.

Our WA algorithm first assigns wavelengths to all of the non-local sessions. It then assigns wavelengths to all the local sessions in each top-level subtree. Consider top-level subtree 1. Since there are in total $x_1(N-1)$ local and non-local sessions transmitted from nodes in this subtree while there are only $x_1(N-x_1)$ wavelengths available, it is clear that we need to reuse some wavelengths previously assigned to non-local sessions to support local sessions. Such wavelength reuse is the cause of the main complexity in the design of an efficient WA algorithm.

Let $n_{i,j}$ denote leaf node j in $\mathcal{S}_i$, where $1 \leq i \leq d^*$ and $1 \leq j \leq x_i$. With respect to $n_{i,j}$, define a *reusable wavelength* to be a wavelength used by $n_{i,j}$ to receive a non-local session (from a node in a different top-level subtree), but not used by $n_{i,j}$ to transmit a non-local session (to a node in a different top-level subtree). Figure 3-5 shows two examples in which λ_1 is a reusable wavelength with respect to $n_{i,j}$. The following lemma states a basic property of non-local sessions and reusable wavelengths.

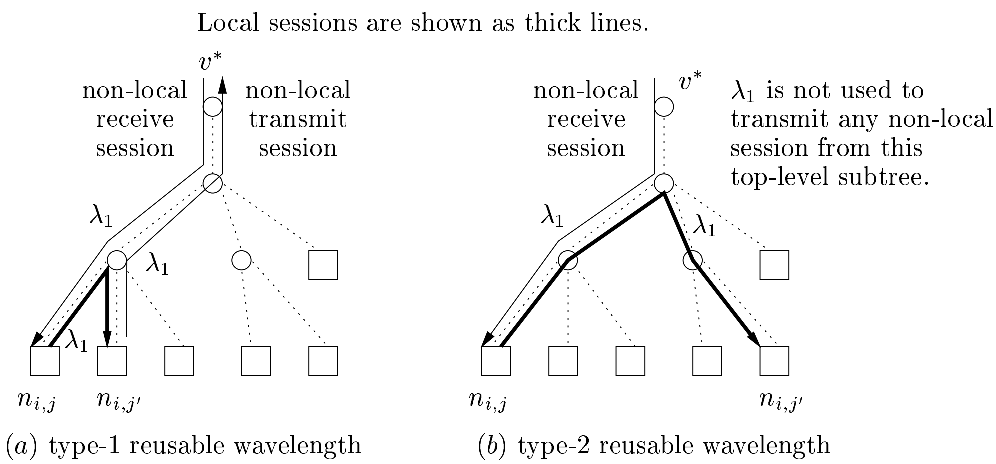


Figure 3-5: Reusable wavelength λ_1 with respect to node $n_{i,j}$.

Lemma 2 *In any given top-level subtree,*

1. *all the non-local sessions are received on distinct wavelengths,*
2. *all the non-local sessions are transmitted on distinct wavelengths, and*
3. *any two reusable wavelengths with respect to the same node or with respect to different nodes in the subtree are distinct.*

Proof:

1. Consider top-level subtree i, where $1 \leq i \leq d^*$. Any pair of non-local sessions which are received in this top-level subtree must traverse the fiber from the bottleneck node v^* to top-level subtree i. It follows that their wavelengths must be distinct, or else there would be a wavelength collision on this fiber.

2. The proof is identical to that of statement 1, except that we consider a pair of transmitted non-local sessions and the link from top-level subtree i to v^*.

3. Since any pair of reusable wavelengths are used to receive two non-local sessions, it follows from statement 1 that they must be distinct. $\square$

With respect to node $n_{i,j}$, define a *type-1* reusable wavelength to be a reusable wavelength which is also used by a different node in the same top-level subtree (i.e. top-level subtree i) to transmit a non-local session. For example, in figure 3-5a, with respect to $n_{i,j}$, λ_1 is a type-1 reusable wavelength. In addition, with respect to $n_{i,j}$, define a *type-1 local node* to be a different node in the same top-level subtree which transmits a non-local session on a reusable wavelength (with respect to $n_{i,j}$). For example, in figure 3-5a, with respect to $n_{i,j}$, $n_{i,j'}$ is a type-1 local node.

With respect to $n_{i,j}$, define a *type-2* reusable wavelength to be a reusable wavelength which is not type-1, i.e. it is not used by any other node in the same top-level subtree to transmit a non-local session. For example, in figure 3-5b, with respect to $n_{i,j}$, λ_1 is a type-2 reusable wavelength. In addition, with respect to $n_{i,j}$, define a *type-2 local node* to be a different node in the same top-level subtree which is not type-1, i.e. it does not transmit a non-local session on any reusable wavelength (with respect to $n_{i,j}$). For example, in figure 3-5b, with respect to $n_{i,j}$, if $n_{i,j'}$ does not use any reusable wavelength (with respect to $n_{i,j}$) to transmit a non-local session, then $n_{i,j'}$ is a type-2 local node.

Notice that, by the above definitions, with respect to any given node $n_{i,j}$, each node $n_{i,j'}$, $j' \neq j$, is either a type-1 or type-2 local node. The following lemma indicates one possible strategy of assigning wavelengths to the local sessions transmitted from $n_{i,j}$ using reusable wavelengths with respect to $n_{i,j}$

Lemma 3 *With respect to node $n_{i,j}$, we have the following properties.*

1. *Node $n_{i,j}$ can transmit a local session to type-1 local node $n_{i,j'}$ on a type-1 reusable wavelength (with respect to $n_{i,j}$) which is used by $n_{i,j'}$ to transmit a non-local session.*
2. *Node $n_{i,j}$ can transmit a local session to type-2 local node $n_{i,j'}$ on any type-2 reusable wavelength (with respect to $n_{i,j}$).*

Proof:

1. Figure 3-5a illustrates statement 1 of the lemma. Let λ_1 denote the reusable wavelength of interest. Let r denote the non-local session received by $n_{i,j}$ on λ_1. Let t denote the non-local session transmitted by $n_{i,j'}$ on λ_1. Let l denote the local session on λ_1 from $n_{i,j}$ to $n_{i,j'}$. We show below that these three sessions never share a fiber, and thus there is no wavelength collision.

Since all the fibers used by r are directed away from the bottleneck node v^* while all the fibers used by t are directed towards v^*, r and t never use the same fiber. We now show that r and l never use the same fiber. We proceed by contradiction. Assume that fiber e, i.e. unidirectional link e, is used by both r and l. Since e is used by r, e is necessarily directed away from v^* and towards $n_{i,j}$. If e is also used by l, then l must have traversed the link which contains e in the opposite direction, i.e. towards v^*, since there is a unique path from $n_{i,j}$ to the starting point of e. This contradicts the fact that no local session uses the same link twice in the opposite directions.

Similar arguments show that t and l never use the same fiber.

2. Figure 3-5b illustrates statement 2 of the lemma. The proof is identical to the proof for statement 1 that r and l never use the same fiber. We shall not repeat the details here. $\square$

Lemma 3 suggests the following method of assigning wavelengths to the local sessions. Consider the local sessions transmitted from node $n_{i,j}$ in top-level subtree $\mathcal{S}_i$. There are $x_i - 1$ such sessions. Let $\mathcal{P}_{i,j}^{(1)}$ and $\mathcal{P}_{i,j}^{(2)}$ be the sets of type-1 and type-2 local nodes with respect to $n_{i,j}$ respectively. Notice that type-1 local nodes (with respect to $n_{i,j}$) have associated with them distinct reusable wavelengths (with respect to $n_{i,j}$). From statement 1 of lemma 3, $n_{i,j}$ can use a distinct type-1 reusable wavelength (with respect to $n_{i,j}$) to transmit a local session to each type-1 local node in $\mathcal{P}_{i,j}^{(1)}$. It remains to provide wavelengths for the local sessions to type-2 local nodes (with respect to $n_{i,j}$).

We shall show shortly in our WA algorithm that it is always possible to assign wavelengths to the non-local sessions so that there are at least $|\mathcal{P}_{i,j}^{(2)}|$ type-2 reusable wavelengths with respect to each node $n_{i,j}$ in the tree. Given $|\mathcal{P}_{i,j}^{(2)}|$ type-2 reusable wavelengths with respect to $n_{i,j}$, statement 2 of lemma 3 implies that $n_{i,j}$ can use a distinct type-2 reusable wavelength (with respect to $n_{i,j}$) to transmit a local session to each type-2 local node in $\mathcal{P}_{i,j}^{(2)}$.

We repeat the same process for all the leaf nodes. From statement 3 of lemma 2, since all the reusable wavelengths (with respect to the same node or with respect to different nodes) in each top-level subtree are distinct, different local sessions (transmitted from the same node or from different nodes) never use the same wavelength.

In conclusion, the condition that there are at least $|\mathcal{P}_{i,j}^{(2)}|$ type-2 reusable wavelengths with respect to each node $n_{i,j}$ is a sufficient condition for the WA of all the local sessions to exist. We state this conclusion formally in the following lemma, which is later used to develop our WA algorithm.

Lemma 4 *If there are at least $|\mathcal{P}_{i,j}^{(2)}|$ type-2 reusable wavelengths with respect to node $n_{i,j}$ for all $1 \leq i \leq d^*$ and $1 \leq j \leq x_i$, then we can assign wavelengths to all the local sessions as follows. Consider the local sessions transmitted from $n_{i,j}$ in $\mathcal{S}_i$.*

1. *To transmit a local session to a type-1 local node in $\mathcal{P}_{i,j}^{(1)}$, $n_{i,j}$ uses a type-1 reusable wavelength (with respect to $n_{i,j}$) which is used by that node to transmit a non-local session.*
2. *To transmit a local session to a type-2 local node in $\mathcal{P}_{i,j}^{(2)}$, $n_{i,j}$ uses a distinct type-2 reusable wavelength (with respect to $n_{i,j}$).*

Our WA algorithm operates in three phases. In phase 1, we assign wavelength bands each of which is used by the non-local sessions from one top-level subtree to another. In phase 2, we perform WA for individual non-local sessions based on the wavelength bands obtained from phase 1. The goal of phase 2 is to assign wavelengths in such a way that enough type-1 and type-2 reusable wavelengths exist to support all local traffic. Finally, in phase 3, we perform WA for local sessions independently in each top-level subtree. The following is our WA algorithm for 1-uniform traffic in an arbitrary tree topology. The algorithm uses w^* wavelengths in each fiber. We shall refer to this algorithm as the *off-line tree WA algorithm.*

Algorithm 1 (Off-Line Tree WA Algorithm) (Use w^* wavelengths in each fiber.)
Number the top-level subtrees so that the numbers of leaf nodes, denoted by $x_1, ..., x_{d^*}$, satisfy $x_1 \geq x_2 \geq ... \geq x_{d^*}$. Note that $w^* = x_1(N - x_1)$.

Phase 1: Assign the wavelength band for the non-local sessions from one top-level subtree to another as follows. For convenience, let $\Lambda_{(i,i')}$ denote the wavelength band for the non-local sessions from $\mathcal{S}_i$ to $\mathcal{S}_{i'}$. Note that $\Lambda_{(i,i')}$ contains $x_i x_{i'}$ wavelengths. Figure 3-6 specifies the wavelength bands between all pairs of top-level subtrees. To obtain wavelength band $\Lambda_{(i,i')}$, where $i < i'$, follow the diagram in figure 3-6a. There are $d^* - 1$ rows of wavelength bands. In row i, $1 \leq i \leq d^* - 1$, we assign consecutive wavelengths starting from wavelength 1 (from left to right) to wavelength bands $\Lambda_{i,i+1}$, ..., $\Lambda_{i,d*}$. For example, the wavelength band $\Lambda_{(1,3)}$ contains 6 wavelengths with indices 10 to 15. On the other hand, to obtain wavelength band $\Lambda_{(i,i')}$, where $i' < i$, follow the diagram in figure 3-6b. There are $d^* - 1$ rows of wavelength bands. In row i', $1 \leq i' \leq d^* - 1$, we assign consecutive wavelengths starting from wavelength w^* (from right to left) to wavelength bands $\Lambda_{i'+1,i'}$, ..., $\Lambda_{d^*,i'}$. For example, the wavelength band $\Lambda_{(4,2)}$ contains 3 wavelengths with indices 10 to 12. Although a specific example is illustrated, the general scheme should be clear.

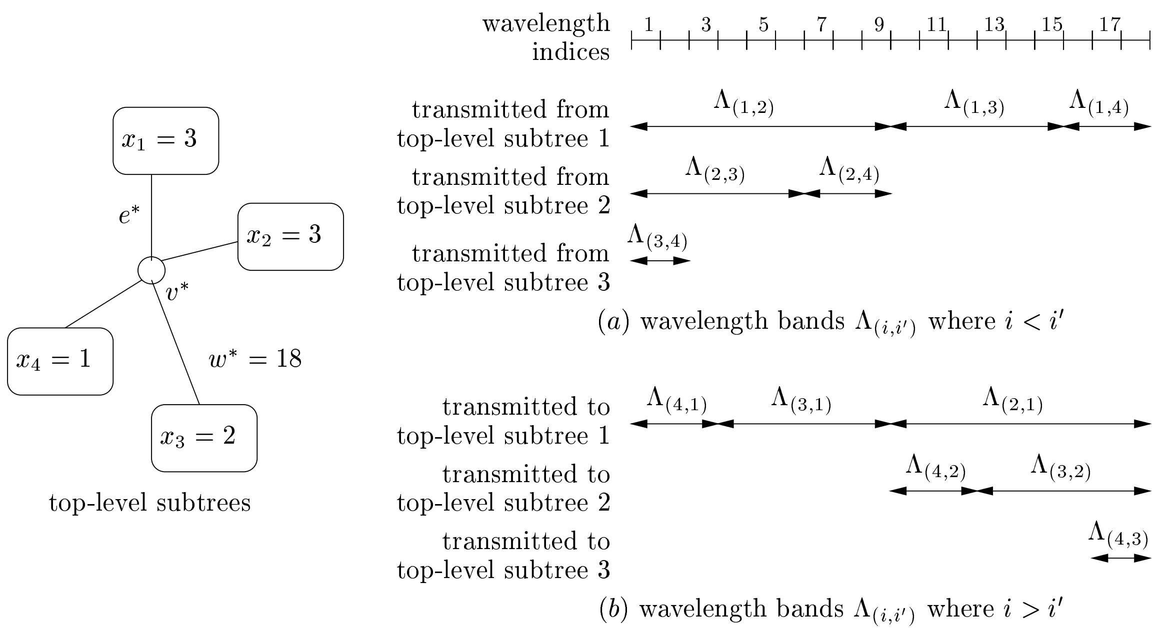


Figure 3-6: Phase 1 of the off-line tree WA algorithm.

We shall show that, in each top-level subtree, the assigned receive wavelength bands do not overlap, i.e. there is no wavelength collision between two non-local receive sessions in two different bands. In addition, the assigned transmit wavelength bands do not overlap. As a result, there is no wavelength collision among the non-local transmit sessions and among the non-local receive sessions in each top-level subtree.

As an example to show how the scheme works, consider two wavelength bands $\Lambda_{(1,4)}$ and $\Lambda_{(2,4)}$ for non-local receive sessions in top-level subtree 4. The highest wavelength index in $\Lambda_{(2,4)}$, denoted by $\lambda^+_{(2,4)}$, is $x_2x_3+x_2x_4$. The lowest wavelength index in $\Lambda_{(1,4)}$, denoted by $\lambda^-_{(1,4)}$, is $x_1x_2+x_1x_3+1$. Since $x_1 \geq x_2 \geq ... \geq x_d$, it follows that $x_1x_2 \geq x_2x_3$ and $x_1x_3 \geq x_2x_4$. Thus,

$$\lambda^{-}_{(1,4)} \quad = \quad x_1x_2 + x_1x_3 + 1 \quad > \quad x_2x_3 + x_2x_4 \quad = \quad \lambda^{+}_{(2,4)}.$$

It follows that a non-local session in wavelength band $\Lambda_{(1,4)}$ and a non-local session in wavelength band $\Lambda_{(2,4)}$ never share the same wavelength and therefore do not collide. A complete general proof is given later in the proof of algorithm correctness.

Phase 2: In this phase, we assign wavelengths to individual non-local sessions based on the wavelength bands obtained from phase 1. Our goal is to assign wavelengths so that there are at least $|\mathcal{P}^{(2)}_{i,j}|$ type-2 reusable wavelengths with respect to node $n_{i,j}$ for all $1 \leq i \leq d^*$ and $1 \leq j \leq x_i$, as suggested by lemma 4.

We first perform partial WA as follows. For each wavelength band $\Lambda_{(i,j)}$ containing x_ix_j wavelengths (used for the non-local sessions from $\mathcal{S}_i$ to $\mathcal{S}_j$), we break the band up into x_j subbands of x_i contiguous wavelengths. The first subband is assigned to be receive wavelengths for node $n_{j,1}$. The second subband is assigned to be receive wavelengths for $n_{j,2}$, and so on. For example, based on the example in figure 3-6, in top-level subtree 1, node $n_{1,1}$ receives three non-local sessions from top-level subtree 2 on the subband of $\Lambda_{(2,1)}$ containing wavelengths 10, 11, and 12. Notice that we have not specified which node in top-level subtree 2 uses a specific wavelength (10, 11, or 12) to transmit to $n_{1,1}$. Figure 3-7 illustrates the result of the partial WA in top-level subtree 1. Note that the partial WA also specifies the subbands used by the nodes in $\mathcal{S}_1$ to transmit to each node in $\mathcal{S}_{i'}$, $i' \neq 1$, as shown in figure 3-7b. For example, in $\Lambda_{(1,2)}$, wavelengths 1, 2, and 3 are used for the non-local sessions from $\mathcal{S}_1$ to $n_{2,1}$.

It remains to specify the source nodes for specific wavelengths in each subband, i.e. filling the empty slots in each subband in figure 3-7b with $n_{1,1}$, $n_{1,2}$, and $n_{1,3}$. Such specifications in top-level subtree 1 can be done independently from the similar specifications in all the other top-level subtrees since the lightpaths corresponding to each subband traverse the same set of fibers outside top-level subtree 1. In other words, the WA outside top-level subtree 1 looks the same regardless of how we fill the empty slots in figure 3-7b. Furthermore, such specifications yield, for each node, the corresponding type-1 and type-2 reusable wavelengths together with type-1 and type-2 local nodes.

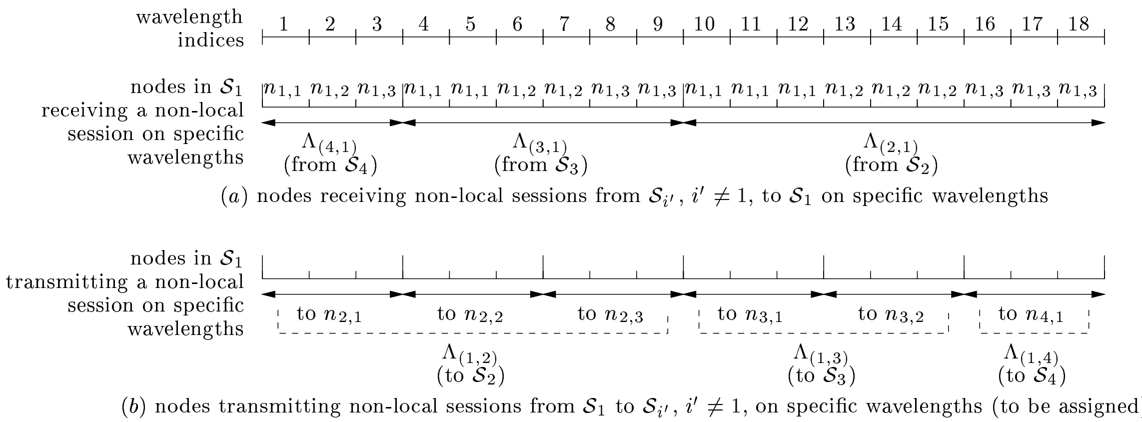


Figure 3-7: The result of the partial WA in phase 2 of the off-line tree WA algorithm for top-level subtree 1 in figure 3-6.

Assume for now that the set of wavelengths used to receive and to transmit non-local sessions are the same in a given top-level subtree i. (This is the case for top-level subtree 1 and any other subtree i with $x_i = x_1$. However, the assumption does not always hold, e.g. top-level subtree 3 in figure 3-6.) We show below how to assign the source nodes in each wavelength subband so that $|P_{i,j}^{(2)}| = 0$, i.e. no type-2 local node with respect to $n_{i,j}$, for each node $n_{i,j}$ in $\mathcal{S}_i$. Note that the condition $|P_{i,j}^{(2)}| = 0$ yields the sufficient condition in lemma 4 for the WA of all the local sessions in top-level subtree i to exist, i.e. there are at least $|P_{i,j}^{(2)}|$ type-2 reusable wavelengths with respect to each node $n_{i,j}$ in $\mathcal{S}_i$. The goal $|P_{i,j}^{(2)}| = 0$ is equivalent to $|P_{i,j}^{(1)}| = x_i - 1$. That is, we must ensure that, each node $n_{i,j'}$, $j' \neq j$, in $\mathcal{S}_i$ transmits at least one non-local session on one of the wavelengths used by $n_{i,j}$ to receive non-local sessions.

We can visualize the problem of assigning the source nodes in each subband using a bipartite graph. We consider each top-level subtree separately. For top-level subtree i, construct a *partial WA bipartite graph* denoted by $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$ as follows. The set $\mathcal{V}_1$ contains the $N - x_i$ leaf nodes outside top-level subtree i, i.e. $\{n_{i',j'} : i' \neq i, 1 \leq j' \leq x_{i'}\}$. The set $\mathcal{V}_2$ is equal to $\mathcal{S}_i$, i.e. $\{n_{i,j} : 1 \leq j \leq x_i\}$. In the set of edges $\mathcal{E}$, an edge joins $n_{i',j'}$ in $\mathcal{V}_1$ and $n_{i,j}$ in $\mathcal{V}_2$ for each wavelength that is used to receive a non-local session from a node in $\mathcal{S}_i$ by $n_{i',j'}$, and is used to receive a non-local session by $n_{i,j}$. There may be multiple edges between the same pair of nodes. For example, figure 3-8a shows the partial WA bipartite graph specified by the partial WA in top-level subtree 1 in figure 3-7. In particular, the edge between $n_{2,1}$ in $\mathcal{V}_1$ and $n_{1,1}$ in $\mathcal{V}_2$ corresponds to wavelength 1 which is used both to receive a non-local session from $\mathcal{S}_1$ by $n_{2,1}$ and to receive a non-local session by $n_{1,1}$. Two edges between $n_{2,3}$ in $\mathcal{V}_1$ and $n_{1,3}$ in $\mathcal{V}_2$ correspond to wavelengths 8 and 9 which are used both to receive a non-local session from $\mathcal{S}_1$ by $n_{2,3}$ and to receive a non-local session by $n_{1,3}$.

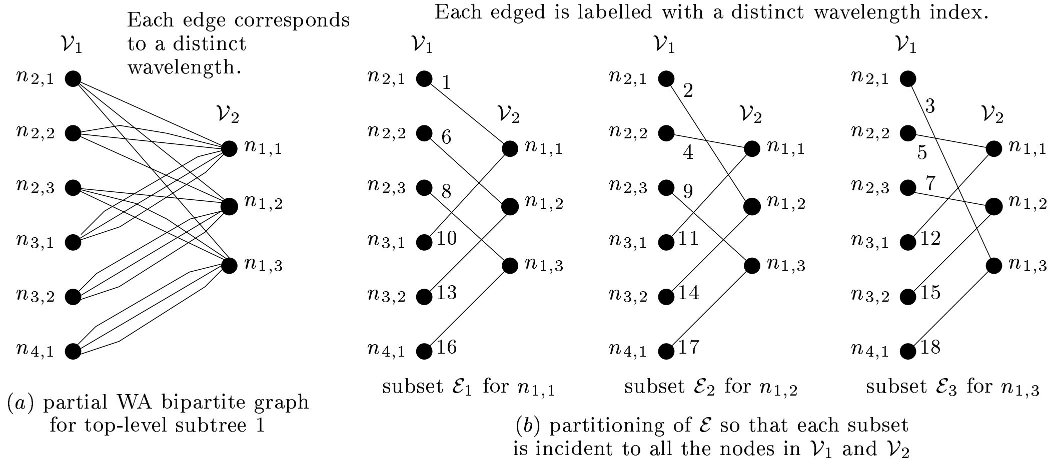


Figure 3-8: Partial WA bipartite graph for top-level subtree 1 in figure 3-6.

From the assumption that, in top-level subtree i, the set of non-local transmit wavelengths is equal to the set of non-local receive wavelengths, it follows that every non-local transmit wavelength corresponds one-to-one to an edge in the partial WA bipartite graph. Since each node $n_{i',j'}$ in $\mathcal{V}_1$ receives x_i non-local sessions from $\mathcal{S}_i$, each node $n_{i',j'}$ has degree x_i. Since each node $n_{i,j}$ in $\mathcal{V}_2$ receives $N - x_i$ non-local sessions, each node $n_{i,j}$ in $\mathcal{V}_2$ has degree $N - x_i$. In addition, there are in total $x_i(N - x_i)$ edges in the partial WA bipartite graph.

We next partition the set of edges, or equivalently the set of non-local transmit wavelengths from $\mathcal{S}_i$, into x_i subsets each with $N - x_i$ edges. Each subset of wavelengths are then used by some node $n_{i,j}$ in $\mathcal{S}_i$ (or equivalently $\mathcal{V}_2$) to transmit its $N - x_i$ non-local sessions to the $N - x_i$ nodes in $\mathcal{V}_1$. Thus, it is necessary that each subset of edges contains $N - x_i$ edges and is incident on all the nodes in $\mathcal{V}_1$, or else there would be a node in $\mathcal{V}_1$ not reachable from $\mathcal{S}_i$ in some subset of wavelengths. To achieve the goal of having $|\mathcal{P}_{i,j}^{(2)}| = 0$ for each $n_{i,j}$ in $\mathcal{S}_i$, we require in addition that each subset of edges is incident to all the nodes in $\mathcal{V}_2$. To see why this additional requirement is a sufficient condition for our goal, consider a given node $n_{i,j}$ in $\mathcal{S}_i$. Since every subset of edges is incident on $n_{i,j}$, it follows that each of the other nodes in $\mathcal{S}_i$ transmits a non-local session on a wavelength used by $n_{i,j}$ to receive a non-local session, and is thus a type-1 local node with respect to $n_{i,j}$. Therefore, with respect to each $n_{i,j}$ in $\mathcal{S}_i$, there is no type-2 local node, i.e. $|\mathcal{P}^{(2)}_{i,j}| = 0$.

Therefore, we want to partition $\mathcal{E}$ into x_i subsets of $N - x_i$ edges so that each subset is incident to all the nodes in $\mathcal{V}_1$ and $\mathcal{V}_2$. We shall show that this partitioning problem can be solved by reducing it to a bipartite matching problem. For example, for the partial WA bipartite graph in figure 3-8a, figure 3-8b shows one possible partitioning of $\mathcal{E}$ such that each subset of edges is incident to all the nodes in $\mathcal{V}_1$ and $\mathcal{V}_2$.

As mentioned above, after the partition of $\mathcal{E}$, we assign the wavelengths corresponding to each subset of $\mathcal{E}$ to each $n_{i,j}$ in $\mathcal{S}_i$ to transmit its non-local sessions. For example, according to figure 3-8b, we assign subsets $\mathcal{E}_1$, $\mathcal{E}_2$, and $\mathcal{E}_3$ to $n_{1,1}$, $n_{1,2}$, and $n_{1,3}$ respectively. Node $n_{1,1}$ transmits its non-local sessions on wavelengths 1, 6, 8, 10, 13, and 16 to $n_{2,1}$, $n_{2,2}$, $n_{2,3}$, $n_{3,1}$, $n_{3,2}$, and $n_{4,1}$ respectively. To complete the example in figure 3-7, we specify the source nodes in each transmit subband based on the partitioning of $\mathcal{E}$ in figure 3-8b. The final result of phase 2 for top-level subtree 1 is shown in figure 3-9b.

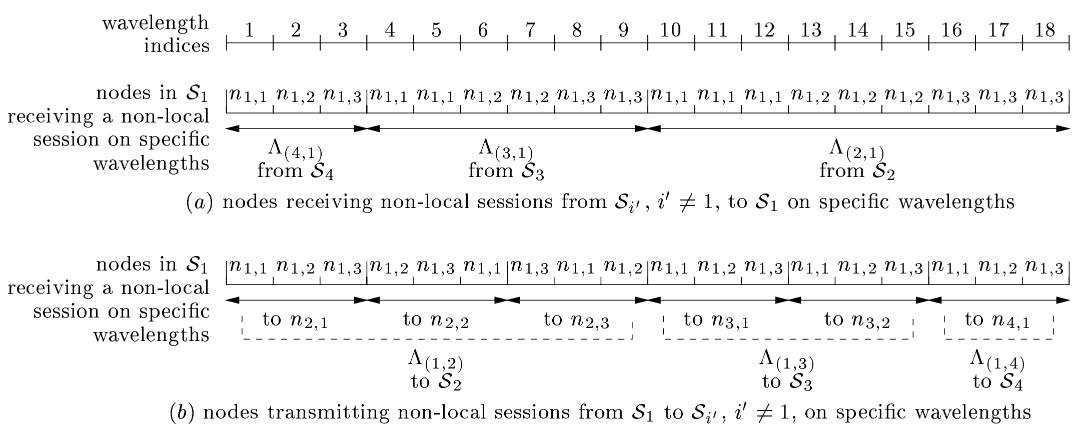


Figure 3-9: The final result of phase 2 of the off-line tree WA algorithm for top-level subtree 1 in figure 3-6.

It remains to consider the top-level subtrees which do not satisfy the previous assumption that the set of non-local transmit wavelengths is equal to the set of non-local receive wavelengths. As an example, consider top-level subtree 3 based on the same example in figure 3-6. The wavelength bands used for non-local sessions to and from top-level subtree 3 are shown in figure 3-10. Notice that non-local receive wavelengths 3, 10, 11, and 12 are not used as non-local transmit wavelengths.

By definition, each of these wavelengths is a type-2 reusable wavelength with respect to some node in $\mathcal{S}_3$.

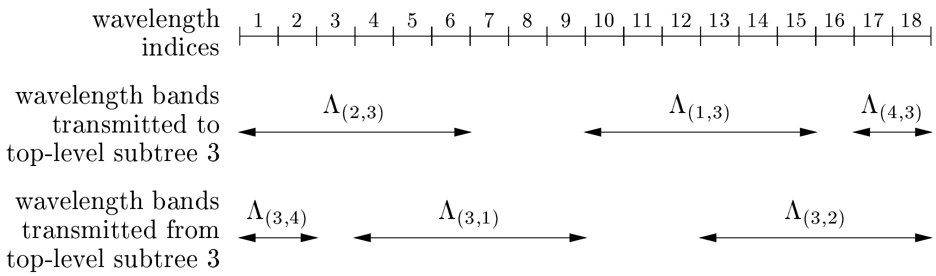


Figure 3-10: Wavelength bands to and from top-level subtree 3 in figure 3-6.

The result of the partial WA is shown in figure 3-11. From the given partial WA, we can create the partial WA bipartite graph for top-level subtree 3 in the same fashion as we have done for top-level subtree 1. This partial WA bipartite graph is the bipartite graph shown in figure 3-12a but with only the solid lines as its edges. Note that only the non-local transmit wavelengths which are also the non-local receive wavelengths in top-level subtree 3 correspond to the edges in the partial WA bipartite graph. For example, the solid edges in figure 3-12a correspond to wavelengths 1, 2, 4, 5, 6, 13, 14, 15, 17, and 18 which are both non-local transmit wavelengths and non-local receive wavelengths in top-level subtree 3. However, the non-local transmit wavelengths 7, 8, 9, and 16 do not correspond to any edge in the partial WA bipartite graph.

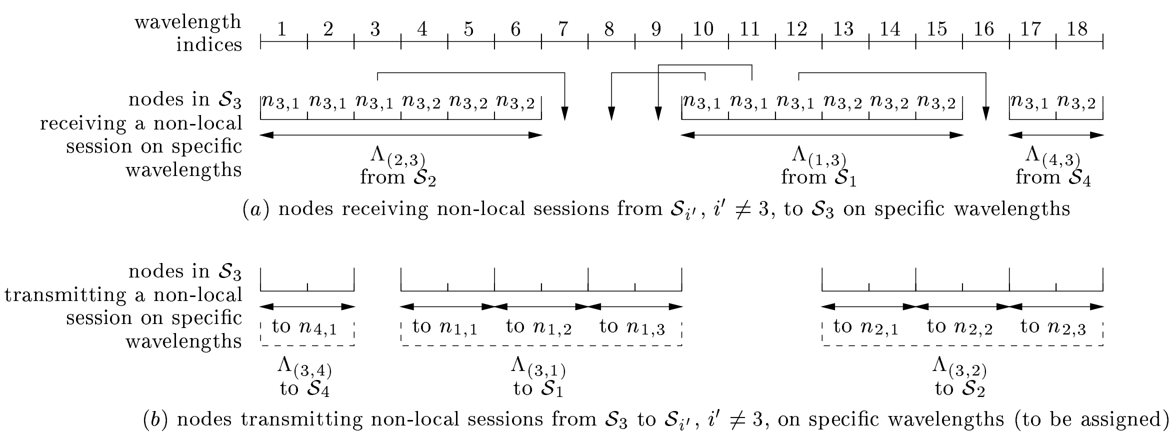


Figure 3-11: The result of the partial WA in phase 2 of the off-line tree WA algorithm for top-level subtree 3 in figure 3-6.

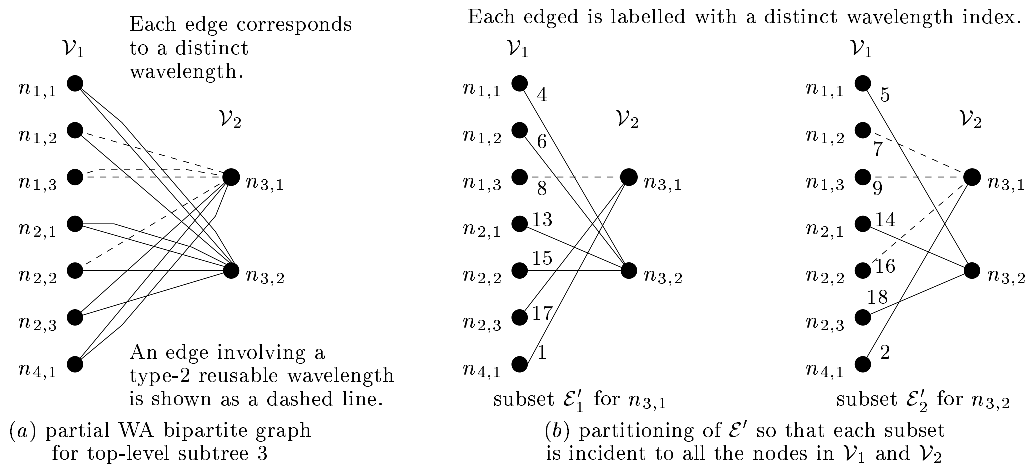


Figure 3-12: Partial WA bipartite graph for top-level subtree 3 in figure 3-6.

In general, given top-level subtree i which does not satisfy the assumption that the set of non-local transmit wavelengths is equal to the set of non-local receive wavelengths, we can perform the partial WA and construct the partial WA bipartite graph, as we have done for top-level subtree 3 in figure 3-12a. Since some non-local transmit wavelengths will not be present in the partial WA bipartite graph, we cannot partition the edges to assign the non-local transmit wavelengths to each node $n_{i,j}$ in $\mathcal{S}_i$ as we have done earlier for top-level subtree 1. To overcome this difficulty, we pair up in a one-to-one fashion each type-2 reusable wavelength with respect to some node in $\mathcal{S}_i$, i.e. a non-local receive wavelength which is not a non-local transmit wavelength, with a non-local transmit wavelength which is not a non-local receive wavelength. If k is the total number of type-2 reusable wavelengths in top-level subtree i, there are $k!$ ways to do this pairing. However, in what follows, it does not matter which way the pairing is carried out. For example, for top-level subtree 3 in figure 3-11a, we can pair up type-2 reusable wavelengths 3, 10, 11, and 12 with non-local transmit wavelengths 7, 8, 9, and 16 respectively.

We modify the set of edges $\mathcal{E}$ in the partial WA bipartite graph as follows. To create a new set of edges, denoted by $\mathcal{E}'$, we regard each type-2 reusable wavelength as being equivalent to its paired value, i.e. a non-local transmit wavelength. As before, an edge joins $n_{i',j'}$ in $\mathcal{V}_1$ and $n_{i,j}$ in $\mathcal{V}_2$ for each wavelength that is used both to receive a non-local session from $\mathcal{S}_i$ by $n_{i',j'}$ and to receive a non-local session by $n_{i,j}$.

It follows that the modified set of edges $\mathcal{E}'$ includes the original set of edges $\mathcal{E}$ together with some extra edges corresponding to all the remaining non-local transmit wavelengths previously not in the partial WA bipartite graph. For example, in figure 3-12a, the dashed edges correspond to non-local transmit wavelengths 7, 8, 9, and 16, which are previously not in the graph. Note that, at this point, each node $n_{i',j'}$ in $\mathcal{V}_1$ has degree x_i. Each node $n_{i,j}$ in $\mathcal{V}_2$ has degree $N - x_i$. In addition, there are in total $x_i(N - x_i)$ edges in the partial WA bipartite graph.

Since all the non-local transmit wavelengths now correspond to an edge in $\mathcal{E}'$, we next partition $\mathcal{E}'$ into x_i disjoint subsets each of which corresponds to $N - x_i$ wavelengths and is assigned to each node $n_{i,j}$ in $\mathcal{S}_i$ to transmit its non-local sessions. As before, to obtain the goal of having at least $|\mathcal{P}_{i,j}^{(2)}|$ type-2 reusable wavelengths with respect to each node $n_{i,j}$ in $\mathcal{S}_i$, we choose to partition $\mathcal{E}'$ such that each subset of edges is incident on all the nodes in $\mathcal{V}_1$ and $\mathcal{V}_2$. We then assign the non-local transmit wavelengths corresponding to each subset of edges to each node $n_{i,j}$ in $\mathcal{S}_i$ to transmit its non-local sessions. For example, in figure 3-12, the set $\mathcal{E}'$ is partitioned into two disjoint sets $\mathcal{E}_1'$ and $\mathcal{E}_2'$, which are then assigned to $n_{3,1}$ and $n_{3,2}$ respectively. In particular, $n_{3,1}$ transmits its non-local sessions on wavelengths 4, 6, 8, 13, 15, 17 and 1 to $n_{1,1}$, $n_{1,2}$, $n_{1,3}$, $n_{2,1}$, $n_{2,2}$, $n_{2,3}$, and $n_{4,1}$ respectively.

We now argue that this procedure yields the desired goal of having at least $|\mathcal{P}_{i,j}^{(2)}|$ type-2 reusable wavelengths with respect to each node $n_{i,j}$ in $\mathcal{S}_i$. Consider a given node $n_{i,j}$ in $\mathcal{S}_i$ and a specific subset of $\mathcal{E}'$ assigned to $n_{i,j'}, j' \neq j$. We know that this subset of $\mathcal{E}'$ is incident on $n_{i,j}$. Consider two cases.

1. In the subset of $\mathcal{E}'$ assigned to $n_{i,j'}$, if there is an edge in $\mathcal{E}$, i.e. a solid edge, incident on $n_{i,j}$, then $n_{i,j'}$ is a type-1 local node with respect to $n_{i,j}$ since $n_{i,j'}$ transmits a non-local session on the wavelength used by $n_{i,j}$ to receive a non-local session.

2. In the subset of $\mathcal{E}'$ assigned to $n_{i,j'}$, if there is no edge in $\mathcal{E}$, i.e. no solid edge, incident on $n_{i,j}$. Then $n_{i,j'}$ is a type-2 local node with respect to $n_{i,j}$ since $n_{i,j'}$ does not transmit any non-local session on the wavelength used by $n_{i,j}$ to receive a non-local session, i.e. $n_{i,j'}$ is not a type-1 local node with respect to $n_{i,j}$.

   For the reason explained below, we assign to $n_{i,j'}$ a unique type-2 reusable wavelength with respect to $n_{i,j}$ corresponding to one incident edge on $n_{i,j}$.

It follows that, with respect to $n_{i,j}$, each of the other nodes in $\mathcal{S}_i$ is either a type-1 local node or a type-2 local node with a unique type-2 reusable wavelength assigned to it. Clearly, there are at least $|\mathcal{P}_{i,j}^{(2)}|$ type-2 reusable wavelengths with respect to $n_{i,j}$. Thus, our goal in phase 2 is achieved.

To complete the example in figure 3-11, we specify the source nodes in each subband in figure 3-11b based on the partitioning of $\mathcal{E}'$ in figure 3-12b. The final result of phase 2 is shown in figure 3-13.

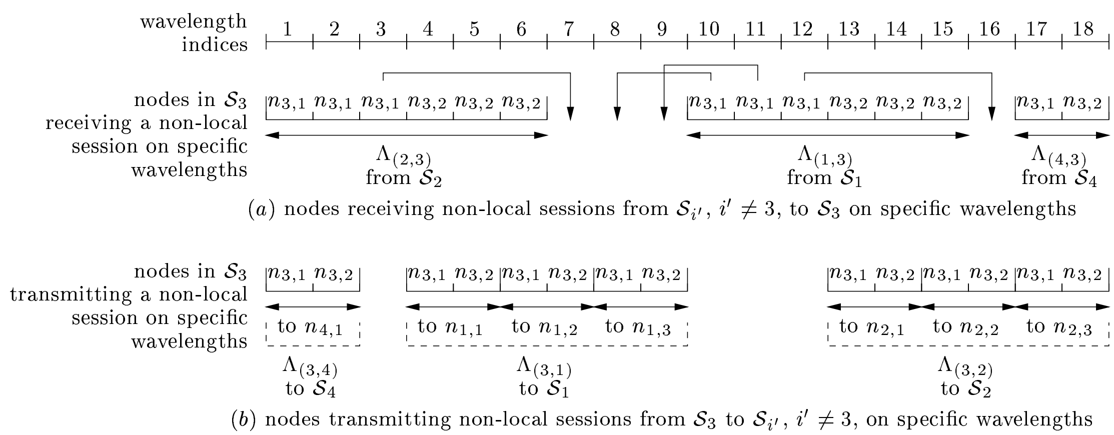


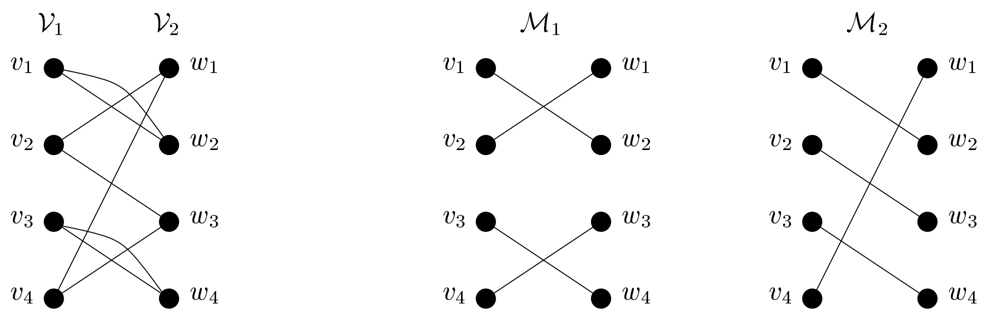
Figure 3-13: The final result of phase 2 of the off-line tree WA algorithm for top-level subtree 3 in figure 3-6.

Phase 3: In this phase, we assign wavelengths to local sessions in each top-level subtree. The assignment based on lemma 4 can be carried out independently in different top-level subtrees.

From phase 2, in top-level subtree i, there are at least $|\mathcal{P}_{i,j}^{(2)}|$ type-2 reusable wavelengths with respect to node $n_{i,j}$ for all $1 \leq j \leq x_i$. Thus, we can assign wavelengths to all the local sessions as follows. Consider the local sessions transmitted from $n_{i,j}$ in top-level subtree i.

1. To transmit a local session to a type-1 local node in $\mathcal{P}_{i,j}^{(1)}$, $n_{i,j}$ uses a type-1 reusable wavelength (with respect to $n_{i,j}$) which is used by that node to transmit a non-local session.
2. To transmit a local session to a type-2 local node in $\mathcal{P}_{i,j}^{(2)}$, $n_{i,j}$ uses a distinct type-2 reusable wavelength (with respect to $n_{i,j}$).

Before we prove the algorithm correctness, we state Hall's theorem and derive a few useful lemmas related to bipartite matchings. We denote a general bipartite graph with three components $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$, where $\mathcal{V}_1$ and $\mathcal{V}_2$ specify two disjoint sets of nodes, and $\mathcal{E}$ specifies a set of edges each of which connects a node in $\mathcal{V}_1$ to a node in $\mathcal{V}_2$. Figure 3-14a shows an example of a bipartite graph.



(a) bipartite graph $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$ (b) two perfect matchings in $\mathcal{E}$

Figure 3-14: Bipartite graph and its perfect matchings.

A matching in a bipartite graph, or in short a bipartite matching, is a subset $\mathcal{M}$ of $\mathcal{E}$ such that no two edges in $\mathcal{M}$ are adjacent. A matching $\mathcal{M}$ is said to *saturate* set $\mathcal{V}_1$ if, for every node in $\mathcal{V}_1$, there is an edge in $\mathcal{M}$ incident on that node. A matching $\mathcal{M}$ which saturates set $\mathcal{V}_1$ is called a *perfect matching*. In figure 3-14b, $\mathcal{M}_1$ and $\mathcal{M}_2$ are two different perfect matchings of $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$.

To describe Hall's theorem, for each subset $\mathcal{S}$ of $\mathcal{V}_1$, let $\mathcal{N}(\mathcal{S})$ denote the *neighborhood* of $\mathcal{S}$ defined as follows. The neighborhood $\mathcal{N}(\mathcal{S})$ is a subset of $\mathcal{V}_2$. Each node w in $\mathcal{V}_2$ is in $\mathcal{N}(\mathcal{S})$ if and only if there is a node v in $\mathcal{S}$ such that (v, w) is an edge in $\mathcal{E}$. For example, in figure 3-14a, if $\mathcal{S} = \{v_1, v_2\}$, then $\mathcal{N}(\mathcal{S}) = \{w_1, w_2, w_3\}$. Alternatively, if $\mathcal{S} = \{v_2, v_4\}$, then $\mathcal{N}(\mathcal{S}) = \{w_1, w_3\}$.

Hall's Theorem [Ber85] *In a bipartite graph $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$, there exists a perfect matching if and only if, for every subset $\mathcal{S}$ of $\mathcal{V}_1$, we have $|\mathcal{N}(\mathcal{S})| \geq |\mathcal{S}|$.*

The next lemma is a consequence of Hall's theorem and was proved in [Lin96]. Since it is less known than Hall's theorem, we provide the proof below.

Lemma 5 [Lin96] *In a bipartite graph $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$ in which each node in $\mathcal{V}_1$ and in $\mathcal{V}_2$ has degree m, the set $\mathcal{E}$ can be partitioned into m disjoint perfect matchings.*[3]

[3] The degree of a node in a bipartite graph $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$ is the number of distinct edges in $\mathcal{E}$ incident on that node. For example, in figure 3-14a, the degree of each node in $\mathcal{V}_1$ is 2.

Proof: We proceed by induction. If $m = 1$, then it is clear that $\mathcal{E}$ is a perfect matching. Assume the lemma holds for degree $m-1$. We now show that the lemma also holds for degree m.

We first show that the existence condition for a perfect matching in Hall's theorem is satisfied. We proceed by contradiction. Suppose there exists a subset $\mathcal{S}$ of $\mathcal{V}_1$ such that $|\mathcal{N}(\mathcal{S})| < |\mathcal{S}|$. There are $m|\mathcal{S}|$ edges incident on $\mathcal{S}$. These $m|\mathcal{S}|$ edges are also incident on $\mathcal{N}(\mathcal{S})$. Since $|\mathcal{N}(\mathcal{S})| < |\mathcal{S}|$, it follows that some node in $\mathcal{N}(\mathcal{S})$ must have degree greater than m, contradicting the assumption that all nodes have degree m. Thus, by Hall's theorem, a perfect matching exists in the bipartite graph of degree m.

Removing the edges corresponding to the above matching, we are left with a bipartite graph of degree $m-1$. By induction hypothesis, the set of edges can be partitioned into $m-1$ disjoint perfect matchings. Therefore, there are in total m disjoint perfect matchings in $\mathcal{E}$. Since $|\mathcal{E}| = m|\mathcal{V}_1|$, each edge in $\mathcal{E}$ belongs to one of these m perfect matchings. In conclusion, the set $\mathcal{E}$ can be partitioned into m disjoint perfect matchings. $\square$

We now prove the correctness of the off-line tree WA algorithm.

Proof of algorithm correctness: It remains to prove the two claims made earlier in the algorithm description. One claim is in phase 1 and the other is in phase 2.

Proof of the claim in phase 1: The claim in phase 1 states that, in each top-level subtree, the assigned receive (transmit) wavelength bands do not overlap. We shall prove the statement for the transmit wavelength bands in top-level subtree i, $1 \leq i \leq d^*$. Similar arguments can be used for the receive wavelength bands.

Define a *group-1 session* to be a session from top-level subtree i to top-level subtree i' where $i < i'$. Similarly, define a *group-2 session* to be a session from top-level subtree i to top-level subtree i' where $i > i'$. We shall show that, in top-level subtree i, (1) no two group-1 sessions from different bands collide, (2) no two group-2 sessions from different bands collide, and (3) no group-1 session collides with a group-2 session.

(1) It suffices to show that wavelength bands $\Lambda_{(i,i+1)}, \Lambda_{(i,i+2)}, ..., \Lambda_{(i,d^*)}$ do not overlap. Since these wavelength bands are specified on the same row in figure 3-6a, they contain distinct wavelengths and do not overlap.

(2) It suffices to show that wavelength bands $\Lambda_{(i,1)}, \Lambda_{(i,2)}, ..., \Lambda_{(i,i-1)}$ do not overlap. From figure 3-6, notice that wavelength band $\Lambda_{(i,i')}$ is a horizontal mirror image of wavelength band $\Lambda_{(i',i)}$. Thus, proving that wavelength bands $\Lambda_{(i,1)}, \Lambda_{(i,2)}, ..., \Lambda_{(i,i-1)}$ do not overlap is equivalent to proving that wavelength bands $\Lambda_{(1,i)}, \Lambda_{(2,i)}, ..., \Lambda_{(i-1,i)}$ do not overlap. For convenience, we shall prove the latter statement. For example, consider $i = 4$ in figure 3-6a, we see that wavelength bands $\Lambda_{1,4}, \Lambda_{2,4}$ and $\Lambda_{3,4}$ do not overlap.

We proceed by showing that, for $1 \leq i' \leq i-2$, the smallest wavelength index in $\Lambda_{(i',i)}$, denoted by $\lambda^-_{(i',i)}$, is strictly greater than the largest wavelength index in $\Lambda_{(i'+1,i)}$, denoted by $\lambda^+_{(i'+1,i)}$. We can express $\lambda^-_{(i',i)}$ and $\lambda^+_{(i'+1,i)}$ as

$$\lambda^-_{(i',i)} = \sum_{i'+1 \leq k \leq i-1} x_{i'} x_k + 1, \qquad \lambda^+_{(i'+1,i)} = \sum_{i'+2 \leq k \leq i} x_{i'+1} x_k.$$

To show that $\lambda^-_{(i',i)} > \lambda^+_{(i'+1,i)}$, we use the following inequality which results from the fact that $x_1 \geq x_2 \geq ... \geq x_d$.

$$\sum_{i'+1 \leq k \leq i-1} x_{i'} x_k \geq \sum_{i'+1 \leq k \leq i-1} x_{i'+1} x_k \geq \sum_{i'+2 \leq k \leq i} x_{i'+1} x_k$$

As a consequence of the above inequality, we show that $\lambda^-_{(i',i)} > \lambda^+_{(i'+1,i)}$ below.

$$\lambda^-_{(i',i)} - \lambda^+_{(i'+1,i)} = \left(\sum_{i'+1 \leq k \leq i-1} x_{i'} x_k - \sum_{i'+2 \leq k \leq i} x_{i'+1} x_k \right) + 1 \geq 1$$

Therefore, we have shown that wavelength band $\Lambda_{(i',i)}$, $1 \leq i' \leq i-2$, contains the wavelength indices all of which are greater than those in wavelength band $\Lambda_{(i'+1,i)}$. It follows that $\Lambda_{(1,i)}, \Lambda_{(2,i)}$, ..., $\Lambda_{(i-1,i)}$ do not overlap.

(3) It suffices to show that, among the non-local sessions transmitted from top-level subtree i, the wavelength index of any group-2 session is strictly greater than the wavelength index of any group-1 session.

The largest wavelength index of any group-1 session from top-level subtree i, denoted by λ^+_i, is in wavelength band $\Lambda_{(i,d^*)}$. The smallest wavelength index of any group-2 session from top-level subtree i, denoted by λ^-_i, is in wavelength band $\Lambda_{(i,1)}$. We can express λ^+_i and λ^-_i as

$$\lambda^+_i = \sum_{i+1 \leq k \leq d^*} x_i x_k, \qquad \lambda^-_i = w^* - \sum_{2 \leq k \leq i} x_k x_1 + 1.$$

We prove that $\lambda_i^- > \lambda_i^+$ as follows

$$\begin{aligned}\lambda_i^- - \lambda_i^+ &= w^* - \left(\sum_{2 \le k \le i} x_k x_1 + \sum_{i+1 \le k \le d^*} x_i x_k\right) + 1 \\ &\ge w^* - \left(\sum_{2 \le k \le i} x_k x_1 + \sum_{i+1 \le k \le d^*} x_1 x_k\right) + 1 \\ &= w^* - \sum_{2 \le k \le d^*} x_1 x_k + 1 \;=\; 1,\end{aligned}$$

where the last equality follows from the fact that $w^* = x_1(x_2 + x_3 + ... + x_{d^*})$. It follows that a group-1 session from top-level subtree i cannot collide with any group-2 session from top-level subtree i.

Proof of the claim in phase 2: The claim in phase 2 states that the set of edges $\mathcal{E}$ in the partial WA bipartite graph $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$ of top-level subtree i, $1 \le i \le d^*$, can be partitioned into x_i disjoint subsets each of which is incident to all the nodes in $\mathcal{V}_1$ and $\mathcal{V}_2$.

We first discuss basic properties of the partial WA bipartite graph of top-level subtree i. Consider the set $\mathcal{V}_1$. Notice that $|\mathcal{V}_1| = N - x_i$, and each node in $\mathcal{V}_1$ has degree x_i. Consider the set $\mathcal{V}_2$. Notice that $|\mathcal{V}_2| = x_i$, and each node in $\mathcal{V}_2$ has degree $N - x_i$. In addition, since $x_i \le N - x_i$, it follows that $|\mathcal{V}_1| \ge |\mathcal{V}_2|$.

If $|\mathcal{V}_1| = |\mathcal{V}_2|$, then each node in $\mathcal{V}_1$ and $\mathcal{V}_2$ has degree x_i. It follows from lemma 5 that $\mathcal{E}$ can be partitioned into x_i disjoint perfect matchings. By definition, each perfect matching is incident on the set $\mathcal{V}_1$. Moreover, each perfect matching must be incident on $\mathcal{V}_2$, or else there would be some adjacent edges in some matching. Thus, $\mathcal{E}$ can be partitioned into x_i disjoint subsets each of which is incident on all the nodes in $\mathcal{V}_1$ and $\mathcal{V}_2$.

It remains to consider the case with $|\mathcal{V}_1| > |\mathcal{V}_2|$. In this case, we can construct a new bipartite graph, denoted by $(\mathcal{V}_1, \mathcal{V}_2', \mathcal{E}')$ as follows. The set $\mathcal{V}_1$ is the same as before. Add $|\mathcal{V}_1| - |\mathcal{V}_2|$ dummy nodes to the set $\mathcal{V}_2$ to create the modified set of nodes $\mathcal{V}_2'$, i.e. $|\mathcal{V}_2'| = |\mathcal{V}_1|$. Label nodes in $\mathcal{V}_2'$ from 1 to $|\mathcal{V}_1|$ such that the dummy nodes are labeled from $|\mathcal{V}_2| + 1$ to $|\mathcal{V}_1|$. For $1 \le j \le |\mathcal{V}_2|$, we select from $\mathcal{E}$ a set of x_i edges incident on node j in $\mathcal{V}_2$ (in the original graph). We include these sets of edges in $\mathcal{E}'$ without any modification. This step is always possible since each node in $\mathcal{V}_2$ originally has degree $N - x_i \ge x_i$. For the remaining edges in $\mathcal{E}$, we reassign their end points originally in $\mathcal{V}_2$ to the dummy nodes in $\mathcal{V}_2'$ such that x_i edges are incident on each dummy node. This step is always possible since there are in total $|\mathcal{V}_2'|x_i$ edges in $\mathcal{E}$.

In the new bipartite graph $(\mathcal{V}_1, \mathcal{V}_2', \mathcal{E}')$, $|\mathcal{V}_2'| = |\mathcal{V}_1|$ and each node has degree x_i. From the above discussion, $\mathcal{E}'$ can be partitioned into x_i disjoint subsets $\mathcal{E}_1'$, ..., $\mathcal{E}_{x_i}'$ each of which is incident on $\mathcal{V}_1$ and $\mathcal{V}_2$. We can create the desired disjoint subsets of edges $\mathcal{E}_1$, ..., $\mathcal{E}_{x_i}$ in the original graph from $\mathcal{E}_1'$, ..., $\mathcal{E}_{x_i}'$ as described next. For $1 \leq j \leq x_i$, we construct part of $\mathcal{E}_j$ from $\mathcal{E}_j'$. From $\mathcal{E}_j'$, include in $\mathcal{E}_j$ the set of edges incident on nodes 1 to $|\mathcal{V}_2|$ in $\mathcal{V}_2'$ without any modification. For the remaining edges, their end points were previously reassigned. We include them in $\mathcal{E}_j$ after reassigning their end points to the original ones. By construction, it is clear that $\mathcal{E}_1$, ..., $\mathcal{E}_{x_i}$ are disjoint, and each $\mathcal{E}_j$ is incident on all the nodes in $\mathcal{V}_1$ and $\mathcal{V}_2$.

Finally, one standard algorithm for finding a perfect matching in a bipartite graph can be found in [CLR90]. Such an algorithm can be used successively for our task of finding x_i disjoint perfect matchings in a bipartite graph with node degree x_i. $\square$

The construction of the off-line tree WA algorithm implies the following theorem.

Theorem 1 *In an arbitrary tree topology with 1-uniform traffic among leaf nodes, $W_{s,1}$ is given by*

$$W_{s,1} \quad = \quad L_{s,1} \quad = \quad w^* = \quad \max_{e \in \mathcal{T}} |\mathcal{N}_{e,1}||\mathcal{N}_{e,2}|.$$

Theorem 1 tells us that wavelength conversion cannot decrease the wavelength requirement for 1-uniform traffic in an arbitrary tree topology. In addition, from statement 2 of lemma 1, the minimum value of w^* is at least $\frac{1}{d^*}(1 - \frac{1}{d^*})N^2$. The tree topologies with w^* close to this lower bound are the ones in which each top-level subtree has approximately N/d^* leaf nodes. Roughly speaking, it is desirable to have all the top-level subtrees support an equal amount of traffic.

It is a simple extension to establish that $W_{s,l} = lW_{s,1}$. First, we use the same argument as in the derivation of w^* in (3.1) to show that the bottleneck link e^* carries lw^* wavelengths in each fiber. Thus, $L_{s,l} \geq lw^*$. To show that $W_{s,l} \leq lw^*$, we apply the off-line tree WA algorithm l times on l disjoint sets each of which contains w^* wavelengths. We state the result formally as a corollary to theorem 1.

Corollary 1 *For an arbitrary tree topology with l-uniform traffic among leaf nodes, $W_{s,l}$ is given by*

$$W_{s,l} \quad = \quad L_l^s \quad = \quad lw^* = \quad l \max_{e \in \mathcal{T}} |\mathcal{N}_{e,1}||\mathcal{N}_{e,2}|.$$

The following example illustrates the resultant WA from the off-line tree WA algorithm.

Example 1 In this example, we shall present the overall WA for 1-uniform traffic in the example tree given in figure 3-6a. Although several parts of the WA are previously shown in the algorithm description, for completeness we shall present all the steps of the off-line tree WA algorithm below.

Figure 3-15 is identical to figure 3-6, which shows the wavelength bands $\Lambda_{(i,j)}$, $i \neq j$, $1 \leq i, j \leq 4$, for the non-local sessions among the four top-level subtrees. These bands are assigned in phase 1 of the algorithm. For example, band $\Lambda_{(2,4)}$ contains wavelengths 7, 8, and 9.

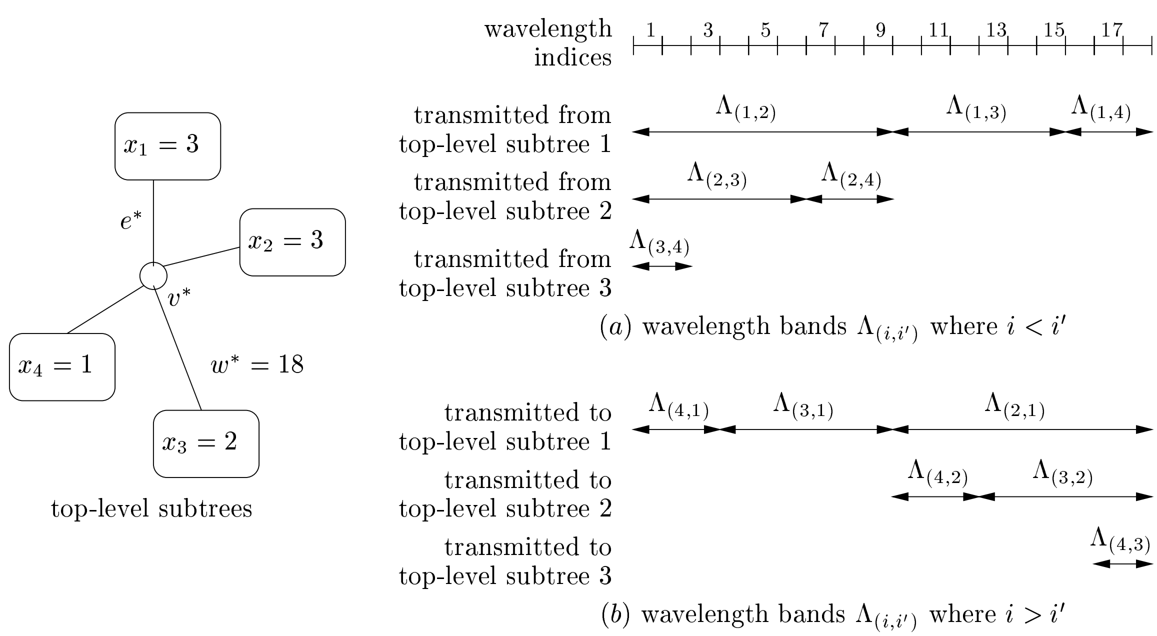


Figure 3-15: Phase 1 of the off-line tree WA algorithm for example 1.

Figures 3-16, 3-18, 3-20, and 3-22 show the results of phase 2 of the off-line tree WA algorithm for top-level subtrees 1, 2, 3, and 4 respectively. In each of these figures, we also present the underlying partial WA bipartite graph and the partition of its edges into disjoint subsets each of which are incident to all the nodes in the graph. For example, consider the result of phase 2 for top-level subtree 2 in figure 3-18. Node $n_{2,1}$ transmits its non-local sessions on wavelengths 10, 15, 17, 1, 4, and 7 to $n_{1,1}$, $n_{1,2}$, $n_{1,3}$, $n_{3,1}$, $n_{3,2}$, and $n_{4,1}$ respectively. Now consider the result of phase 2 for top-level subtree 4 in figure 3-22. Since there is only a single node in top-level subtree 4, the partial WA yields the complete WA for all the non-local sessions to and from $\mathcal{S}_4$. There is no need to create the partial WA bipartite graph and partition its edges as we have done for all the other three top-level subtrees. Moreover, since there is no local session in top-level subtree 4, we need not perform phase 3 for top-level subtree 4.

Figures 3-17, 3-19, and 3-21 show the results of phase 3 of the off-line tree WA algorithm for top-level subtrees 1, 2, and 3 respectively. For example, consider the result of phase 3 for top-level subtree 2 in figure 3-19. Node $n_{2,1}$ transmits its local sessions on wavelengths 13 and 14 to $n_{2,2}$ and $n_{2,3}$ respectively. Notice that the choice of the wavelengths for the local sessions may not be unique. From figure 3-18, since wavelengths 2 and 3 are non-local receive wavelengths for $n_{2,1}$ and are used as non-local transmit wavelengths for $n_{2,2}$ and $n_{2,3}$ respectively, $n_{2,1}$ may also use wavelengths 2 and 3 to transmit its local sessions to $n_{2,2}$ and $n_{2,3}$ respectively.

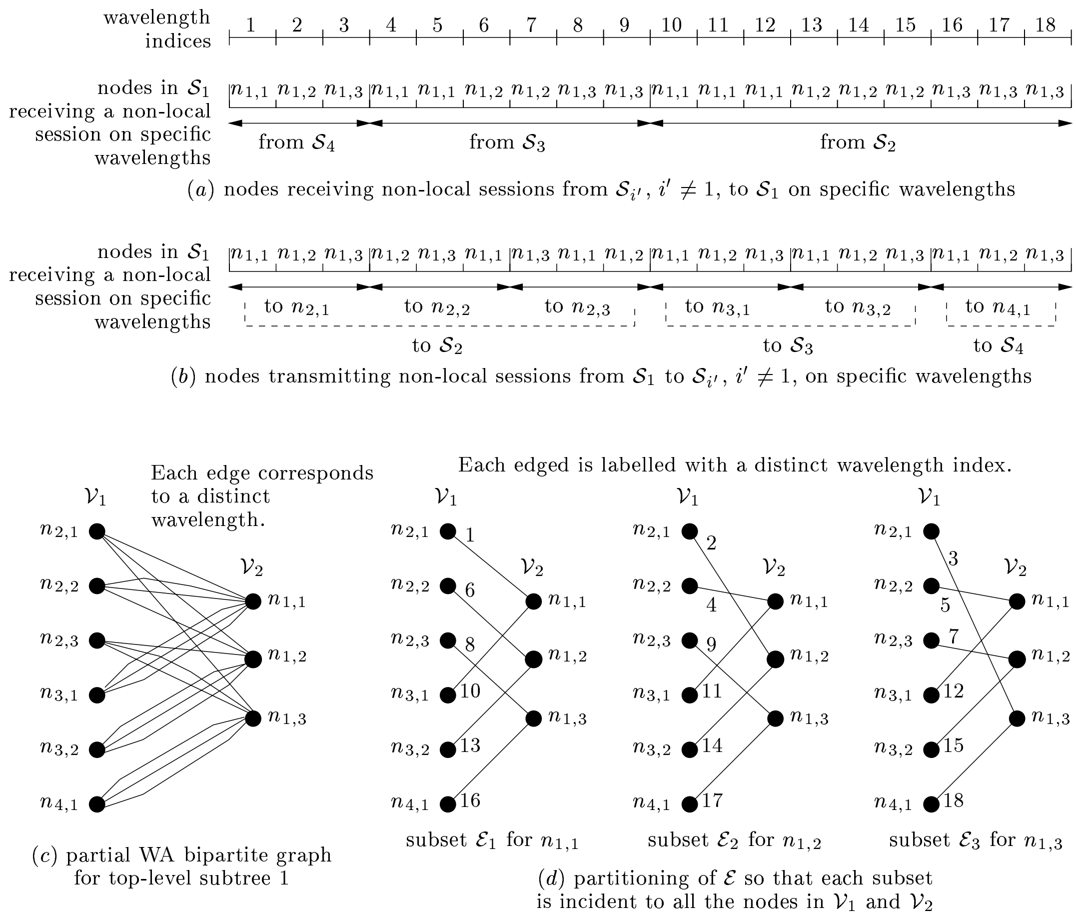


Figure 3-16: Phase 2 of the off-line tree WA algorithm for top-level subtree 1 in example 1.

	destination		
	$n_{1,1}$	$n_{1,2}$	$n_{1,3}$
source			
$n_{1,1}$	-	4	5
$n_{1,2}$	6	-	7
$n_{1,3}$	8	9	-

wavelengths for the local sessions in top-level subtree 1

Figure 3-17: Phase 3 of the off-line tree WA algorithm for top-level subtree 1 in example 1.

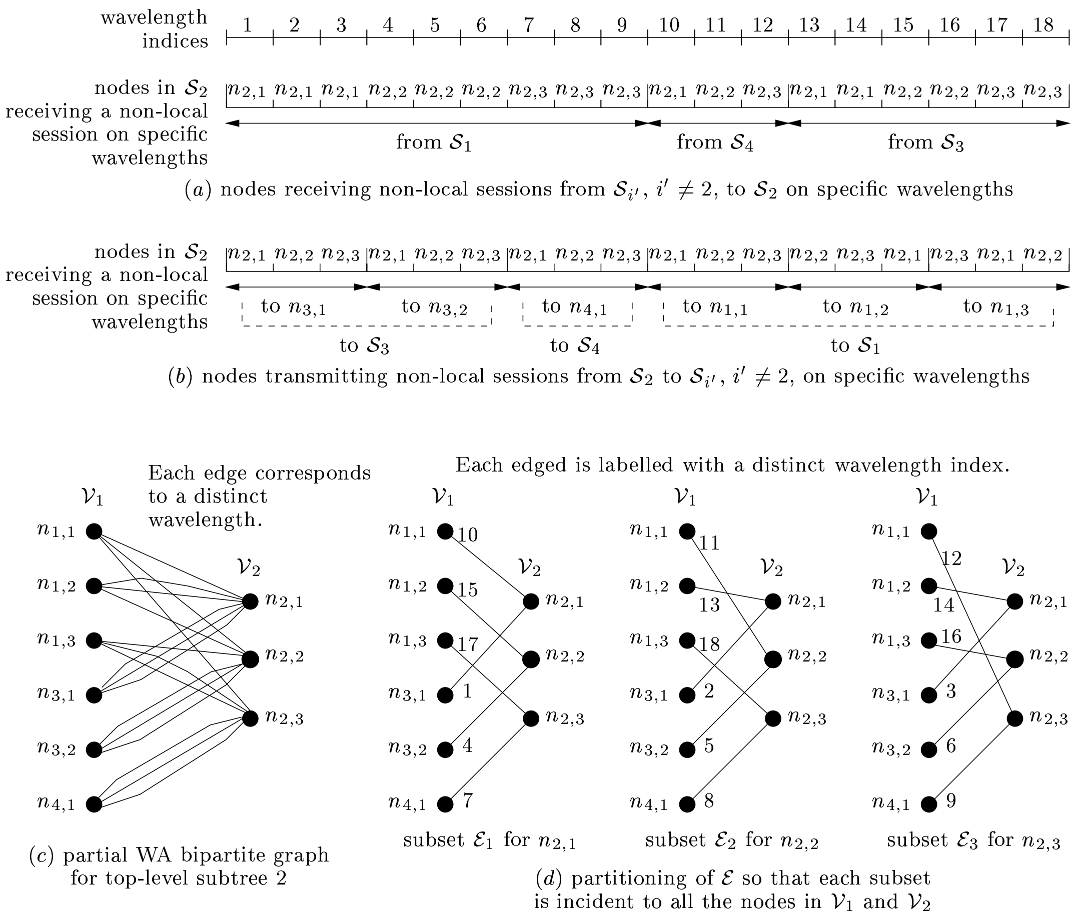


Figure 3-18: Phase 2 of the off-line tree WA algorithm for top-level subtree 2 in example 1.

	destination		
	$n_{2,1}$	$n_{2,2}$	$n_{2,3}$
source			
$n_{2,1}$	-	13	14
$n_{2,2}$	15	-	16
$n_{2,3}$	17	18	-

wavelengths for the local sessions in top-level subtree 2

Figure 3-19: Phase 3 of the off-line tree WA algorithm for top-level subtree 2 in example 1.

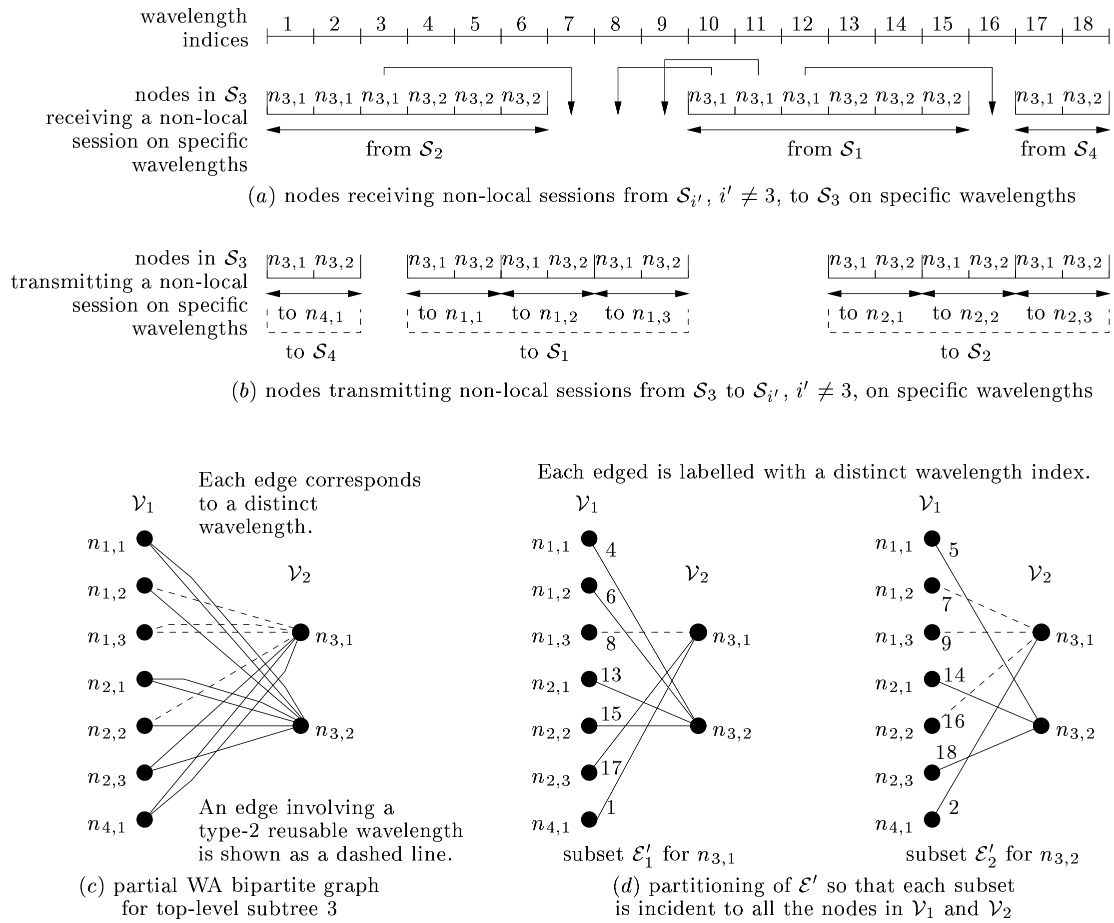


Figure 3-20: Phase 2 of the off-line tree WA algorithm for top-level subtree 3 in example 1.

	destination	
	$n_{3,1}$	$n_{3,2}$
source		
$n_{3,1}$	-	4
$n_{3,2}$	7	-

wavelengths for the local sessions in top-level subtree 3

Figure 3-21: Phase 3 of the off-line tree WA algorithm for top-level subtree 3 in example 1.

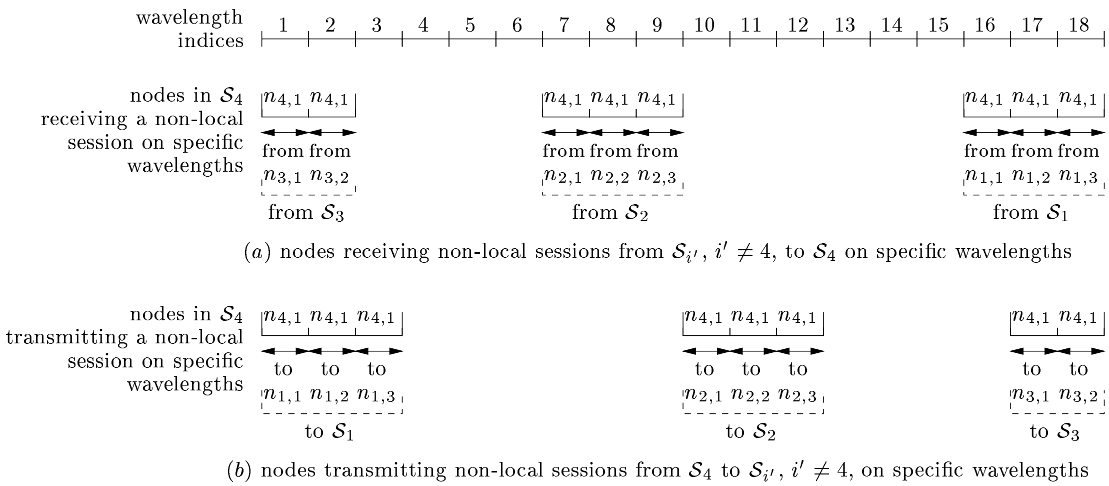


Figure 3-22: Phase 2 of the off-line tree WA algorithm for top-level subtree 4 in example 1.

3.1.1 Regular Tree Topologies

For some regular tree topologies, we can describe a WA scheme compactly as an algebraic expression which we shall refer to as a *WA code*. Let N be the number of end nodes, which we label as nodes $0, 1, ..., N-1$. For each of the N nodes, a code specifies a *WA vector*, which is an N-vector containing wavelength indices used to transmit to nodes $0, 1, ..., N-1$ respectively. We present two examples of WA codes below.

Example 2 (Star Topology) A star topology is a special case of an arbitrary tree topology. The star topology with N end nodes is shown in figure 3-23a. For 1-uniform traffic, $w^* = N - 1$. Since each top-level subtree is a single node, only phase 1 of the off-line tree WA algorithm is required.

For example, when $N = 4$, the resultant WA is illustrated in figure 3-23b. From this WA, we can write down the WA vector $\mathbf{v}_j$ for node j, $0 \le j \le 3$, as follows

$$\mathbf{v}_0 = \begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \end{bmatrix}, \quad \mathbf{v}_1 = \begin{bmatrix} 3 \\ 0 \\ 1 \\ 2 \end{bmatrix}, \quad \mathbf{v}_2 = \begin{bmatrix} 2 \\ 3 \\ 0 \\ 1 \end{bmatrix}, \quad \mathbf{v}_3 = \begin{bmatrix} 3 \\ 0 \\ 1 \\ 2 \end{bmatrix},$$

where we use wavelength 0 as a dummy wavelength index for the self-traffic entries. More generally, for node j, $0 \le j \le N$, the WA vector $\mathbf{v}_j$ can be expressed compactly as

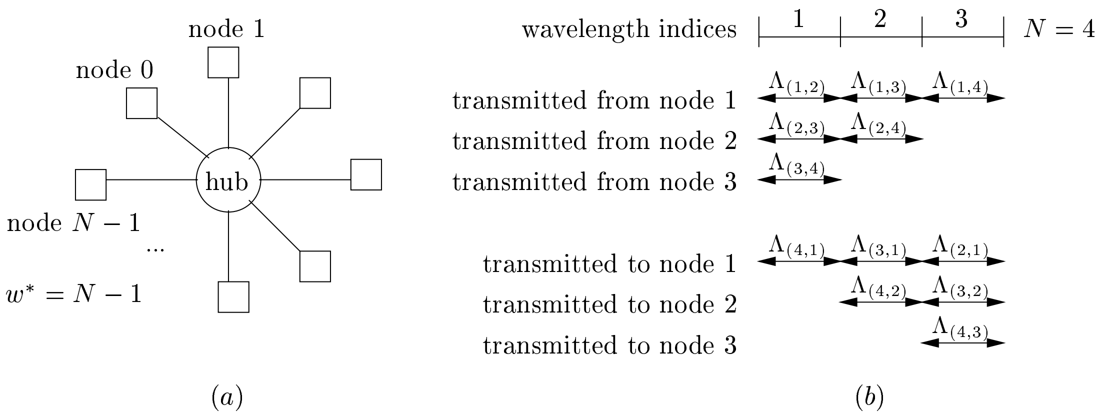


Figure 3-23: Star topology and its WA

$$\mathbf{v}_j \;=\; \mathbf{c}_N - j\mathbf{e}_N \bmod N,$$

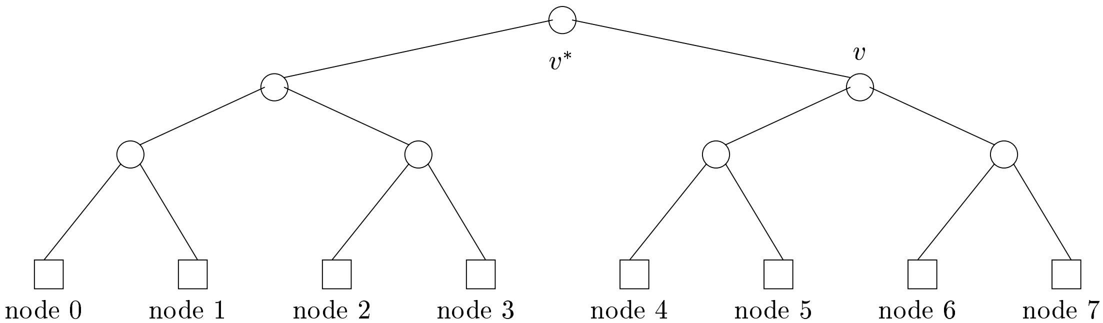
where $\mathbf{c}_N = [0, 1, ..., N-1]$ and $\mathbf{e}_N = [1, 1, ..., 1]$.

Example 3 (Binary Tree Topology) Consider a binary tree topology containing $N = 2^n$ end nodes for some positive integer n, as illustrated in figure 3-24. For 1-uniform traffic, $w^* = N^2/4$. Although one WA code can be obtained from the off-line tree WA algorithm by choosing node v in figure 3-24 as the bottleneck node with three top-level subtrees, we can obtain a different WA code which can be expressed more compactly by choosing the root node as the bottleneck node v^*.

With the root node as the bottleneck node v^*, there are only two top-level subtrees, violating the previous assumption of having at least three top-level subtrees. However, in this special case in which the bottleneck link e^* separates leaf nodes into two equal sets, i.e. $x_1 = x_2 = N/2$, the off-line tree WA algorithm can still be applied. When there are only two top-level subtrees with $x_1 > x_2$, the algorithm breaks down since each node in top-level subtree 1 may not be able to possess up to $x_1 - 1$ reusable wavelengths in phase 2. To see this, note that each reusable wavelength corresponds to a non-local receive session. When $x_1 - 1 > x_2$, there are strictly less than $x_1 - 1$ non-local receive wavelengths with respect to each node in top-level subtree 1. Thus, phase 2 of the algorithm cannot terminate with $x_1 - 1$ or more reusable wavelengths with respect to each leaf node. With only two top-level subtrees but with $x_1 = x_2 = N/2$, such a problem does not occur.

Number leaf nodes $n_{1,1}$, $n_{1,2}$, ..., $n_{1,\frac{N}{2}}$, $n_{2,1}$, $n_{2,2}$, ..., $n_{2,\frac{N}{2}}$ from 0 to $N-1$. By applying the off-line tree WA algorithm with two top-level subtrees, the WA code is shown in figure 3-24 for

$N=8$ and $w^*=16$. We use 0 as a dummy wavelength index for the self-traffic entries.



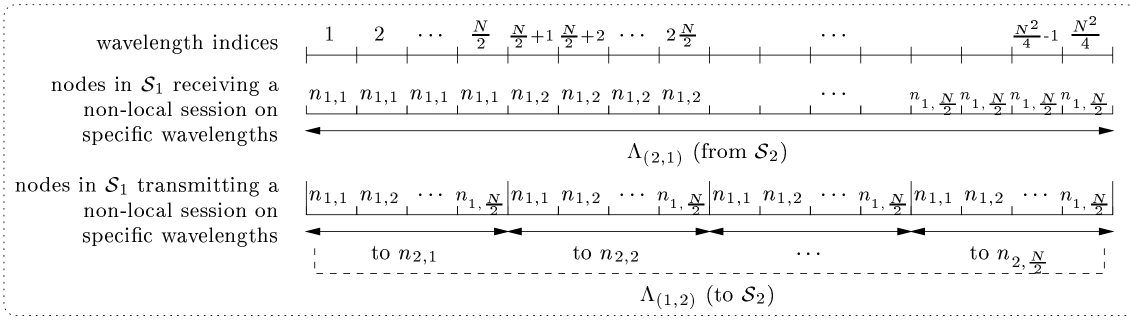
$$
\mathbf{v}_0 = \begin{bmatrix} 0\\2\\3\\4\\1\\5\\9\\13 \end{bmatrix}
\mathbf{v}_1 = \begin{bmatrix} 5\\0\\7\\8\\2\\6\\10\\14 \end{bmatrix}
\mathbf{v}_2 = \begin{bmatrix} 9\\10\\0\\12\\3\\7\\11\\15 \end{bmatrix}
\mathbf{v}_3 = \begin{bmatrix} 13\\14\\15\\0\\4\\8\\12\\16 \end{bmatrix}
\mathbf{v}_4 = \begin{bmatrix} 1\\5\\9\\13\\0\\2\\3\\4 \end{bmatrix}
\mathbf{v}_5 = \begin{bmatrix} 2\\6\\10\\14\\5\\0\\7\\8 \end{bmatrix}
\mathbf{v}_6 = \begin{bmatrix} 3\\7\\11\\15\\9\\10\\0\\12 \end{bmatrix}
\mathbf{v}_7 = \begin{bmatrix} 4\\8\\12\\16\\13\\14\\15\\0 \end{bmatrix}
$$

Figure 3-24: Binary tree topology and its WA code

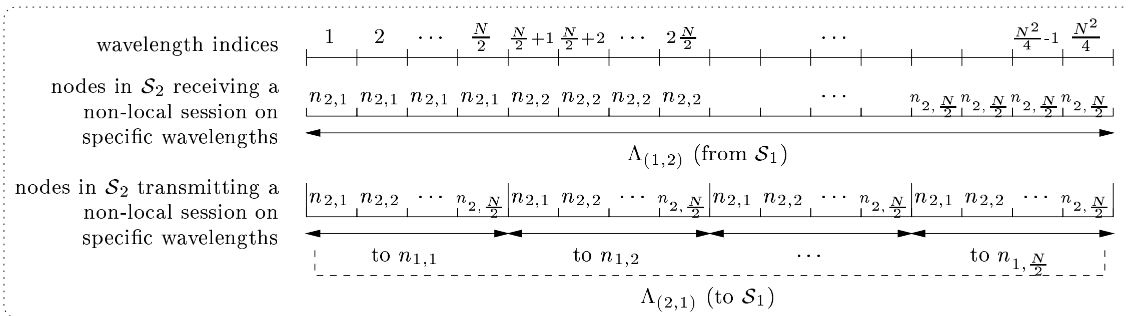
For a general value of n, phase 2 of the off-line tree WA algorithm is illustrated in figure 3-25. For example, consider the non-local sessions transmitted by node $n_{1,j}$, where $1 \le j \le \frac{N}{2}$. Node $n_{1,j}$ transmits to $n_{2,1}$, $n_{2,2}$, ..., $n_{2,\frac{N}{2}}$ on wavelengths j, $\frac{N}{2}+j$, ..., $(\frac{N}{2}-1)\frac{N}{2}+j$ respectively. With respect to $n_{1,j}$, wavelength $(j-1)\frac{N}{2}+k$, where $k \neq j$, is a type-1 reusable wavelength used by $n_{1,k}$ to transmit a non-local session. Thus, in phase 3 of the off-line tree WA algorithm, node $n_{1,j}$ transmits a local session to $n_{1,k}$ on wavelength $(j-1)\frac{N}{2}+k$.

From the above discussion, WA vectors $\mathbf{v}_0$, $\mathbf{v}_1$, ..., and $\mathbf{v}_{\frac{N}{2}-1}$ for the leaf nodes in top-level subtree 1 are given by

$$
\begin{bmatrix} | & | & & | \\ \mathbf{v}_0 & \mathbf{v}_1 & \cdots & \mathbf{v}_{\frac{N}{2}-1} \\ | & | & & | \end{bmatrix}
=
\left[\begin{array}{ccccc}
0 & \frac{N}{2}+1 & 2\frac{N}{2}+1 & & (\frac{N}{2}-1)\frac{N}{2}+1 \\
2 & 0 & 2\frac{N}{2}+1 & & (\frac{N}{2}-1)\frac{N}{2}+2 \\
3 & \frac{N}{2}+2 & 0 & \cdots & (\frac{N}{2}-1)\frac{N}{2}+3 \\
\vdots & \vdots & \vdots & & \vdots \\
\frac{N}{2} & 2\frac{N}{2} & 3\frac{N}{2} & & \frac{N^2}{4} \\
\hline
1 & 2 & 3 & \cdots & \frac{N}{2} \\
\frac{N}{2}+1 & \frac{N}{2}+2 & \frac{N}{2}+3 & \cdots & 2\frac{N}{2} \\
 & & \vdots & & \\
(\frac{N}{2}-1)\frac{N}{2}+1 & (\frac{N}{2}-1)\frac{N}{2}+2 & (\frac{N}{2}-1)\frac{N}{2}+3 & \cdots & 0
\end{array}\right].
$$



WA for top-level subtree 1



WA for top-level subtree 2

Figure 3-25: Phase 2 of the off-line tree WA algorithm for a binary tree topology.

Note that the first $\frac{N}{2}$ entries in each WA vector correspond to local sessions, while the last $\frac{N}{2}$ entries in each WA vector correspond to non-local sessions.

To express the WA code compactly, define matrix C_k to be a $k \times k$ matrix whose ith row, $0 \leq i \leq k-1$, is $[ik+1, ..., ik+k]$. In addition, define the matrix $\mathsf{C}_k^{\mathsf{T0}}$ to be the transpose of C_k with all the diagonal entries set to 0. For example,

$$\mathsf{C}_4 = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix}, \quad \mathsf{C}_4^{\mathsf{T0}} = \begin{bmatrix} 0 & 5 & 9 & 13 \\ 2 & 0 & 10 & 14 \\ 3 & 7 & 0 & 15 \\ 4 & 8 & 12 & 0 \end{bmatrix}.$$

In terms of matrices $\mathsf{C}_{N/2}$ and $\mathsf{C}_{N/2}^{\mathsf{T0}}$, the above WA vectors $\mathbf{v}_0$, ..., and $\mathbf{v}_{\frac{N}{2}-1}$ can be expressed as

$$\begin{bmatrix} | & | & & | \\ \mathbf{v}_0 & \mathbf{v}_1 & \cdots & \mathbf{v}_{\frac{N}{2}-1} \\ | & | & & | \end{bmatrix} = \begin{bmatrix} \mathsf{C}_{N/2}^{\mathsf{T0}} \\ \mathsf{C}_{N/2} \end{bmatrix}.$$

To construct WA vectors $\mathbf{v}_{\frac{N}{2}}$, ..., and $\mathbf{v}_{N-1}$ for the leaf nodes in top-level subtree 2, consider again phase 2 of the off-line tree WA algorithm in figure 3-25. Notice that the WA in top-level subtree 2 differs from that in top-level subtree 1 only in the source and destination node indices. It follows that WA vectors $\mathbf{v}_{\frac{N}{2}}, ..., \mathbf{v}_{N-1}$ are the same as $\mathbf{v}_0, ..., \mathbf{v}_{\frac{N}{2}-1}$ but with the exchange of the first $\frac{N}{2}$ rows and the last $\frac{N}{2}$ rows, i.e.

$$\begin{bmatrix} | & | & & | \\ \mathbf{v}_{\frac{N}{2}} & \mathbf{v}_1 & \cdots & \mathbf{v}_{N-1} \\ | & | & & | \end{bmatrix} = \begin{bmatrix} \mathsf{C}_{N/2} \\ \mathsf{C}_{N/2}^{\mathsf{T0}} \end{bmatrix}.$$

In conclusion, we can express the WA code compactly as

$$\begin{bmatrix} | & | & & | \\ \mathbf{v}_0 & \mathbf{v}_1 & \cdots & \mathbf{v}_{N-1} \\ | & | & & | \end{bmatrix} = \begin{bmatrix} \mathsf{C}_{N/2}^{\mathsf{T0}} & \mathsf{C}_{N/2} \\ \mathsf{C}_{N/2} & \mathsf{C}_{N/2}^{\mathsf{T0}} \end{bmatrix}.$$

3.2 Bidirectional Ring Topologies

In this section, we discuss the RWA for l-uniform traffic in a bidirectional ring topology. Figure 3-26 shows a bidirectional ring topology with $N > 2$ end nodes.[4] Unlike section 3.1 on arbitrary tree topologies, we assume that each node in the network is an end node.

Let $L_{s,l}$ denote the minimum number of wavelengths which, if provided in each fiber, can support l-uniform traffic given full wavelength conversion at all nodes. To obtain a lower bound on $L_{s,l}$, we use the argument referred to as the *link counting bound* in [Pan92]. Let H be the sum

[4]The RWA problem for the ring with two nodes is trivial.

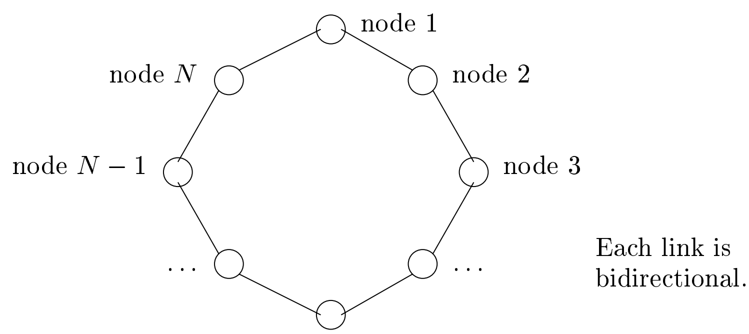


Figure 3-26: The bidirectional ring topology with $N > 2$ nodes.

of the number of hops traversed by each of the sessions under shortest path routing, and F be the number of fibers in the network. Then some fiber must support at least $\lceil H/F \rceil$ wavelengths, i.e. $L_{s,l} \geq \lceil H/F \rceil$. For l-uniform traffic in the N-node ring, it is straightforward to derive H as shown below.

$$H = \begin{cases} lN(N^2-1)/4, & N \text{ odd}, \\ lN^3/4, & N \text{ even} \end{cases}$$

Since $F = 2N$, it follows that

$$L_{s,l} \geq \left\lceil \frac{H}{F} \right\rceil = \begin{cases} l(N^2-1)/8, & N \text{ odd}, \\ \lceil lN^2/8 \rceil, & N \text{ even}. \end{cases}$$

Note that, for N odd, $(N^2-1)/8 = (N-1)(N+1)/8$ is always an integer since one of the factors $(N-1)$ and $(N+1)$ is divisible by 4 while the other is divisible by 2.

Let $W_{s,l}$ denote the minimum number of wavelengths which, if provided in each fiber, can support l-uniform traffic with no wavelength conversion. There are some known results about the value of $W_{s,l}$. For N odd, $W_{s,l} = l(N^2-1)/8$ [Elr93, Wil96]. In addition, for N even and $l = 1$, $W_{s,1} = N^2/8$ if N is divisible by 4, and $W_{s,1} = N^2/8 + 1/2$ if N is not divisible by 4 [Wil96]. In [Wil96], an explicit RWA algorithm is given as a proof on the value of $W_{s,1}$. Since the proof in [Wil96] is rather involved, an alternative and simple proof based on induction was suggested as an exercise in [RS01] to show that $W_{s,l} \leq l(N^2-1)/8$ for N odd. In what follows, we use the idea of the inductive proof to derive a general expression for the upper bound

$$W_{s,l} \leq \begin{cases} l(N^2-1)/8, & N \text{ odd}, \\ \lceil lN^2/8 \rceil, & N \text{ even}, \end{cases}$$

which implies $W_{s,l} = L_{s,l}$.

We first consider N odd and $l = 1$. For $N = 3$, it is easy to see that, under shortest path routing, $W_{s,1} = 1$. For a higher value of N, we find the RWA by inserting two new nodes at a time and updating the RWA, starting from the 3-node ring. More specifically, given a k-node ring, where k is odd, we add two new nodes so that they are $(k+1)/2$ hops apart, as shown in figure 3-27 for $k = 3$ and $k = 5$.

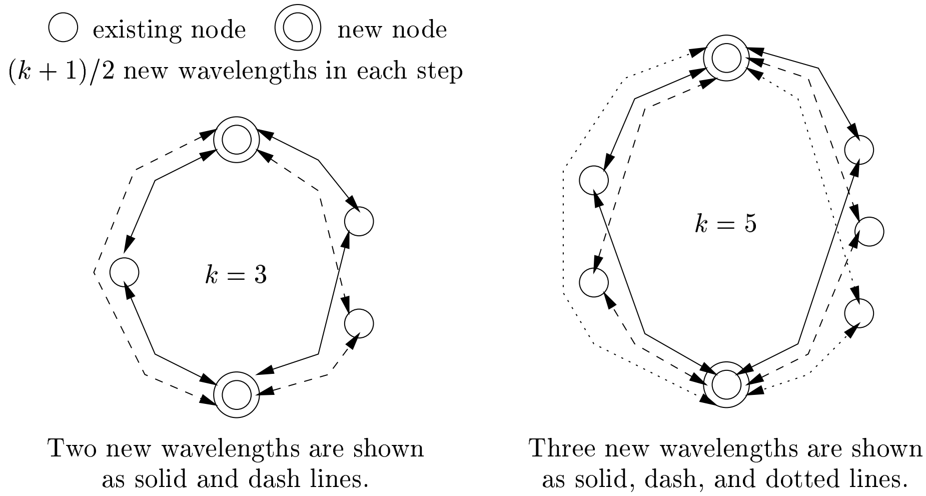


Figure 3-27: RWA update step for the k-node ring, where k is odd. Only the new sessions are shown.

The RWA of the sessions not terminated, i.e. transmitted and/or received, at the new nodes remain the same, although their path lengths may increase. The RWA of the sessions terminated at the new nodes is shown in figure 3-27. In particular, the RWA is chosen based on shortest path routing and efficient wavelength reuse such that each new wavelength is used on every fiber. Note that each RWA update step, i.e. adding two new nodes to the k-node ring, requires $(k + 1)/2$ new wavelengths. By repeating the update step until we obtain the N-node ring, it is clear that all the $N(N - 1)$ sessions in 1-uniform traffic are assigned some wavelength. Accordingly, the number of wavelengths used is

$$1 + \left(\frac{3+1}{2}\right) + \left(\frac{5+1}{2}\right) + ... + \left(\frac{(N-2)+1}{2}\right) = \frac{N^2 - 1}{8}.$$

It follows that $W_{s,1} \leq (N^2 - 1)/8$.

For N odd and $l > 1$, we can repeat the above RWA l times on l disjoint sets of wavelengths.

Thus, $W_{s,l} \leq lW_{s,1} \leq l(N^2-1)/8$.

We now consider N even and $l=1$. For $N=2$, it is trivial that $W_{s,1}=1$. We choose to route the two sessions in the clockwise (CW) ring direction, as shown in figure 3-28a, for the reason to be explained shortly. For a higher value of N, we update the RWA starting from the 2-node ring by inserting two new nodes in each step. Given a k-node ring, where k is even, we add two new nodes so that they are $k/2+1$ hops apart, as shown in figure 3-28b for $k=2$ and $k=4$.

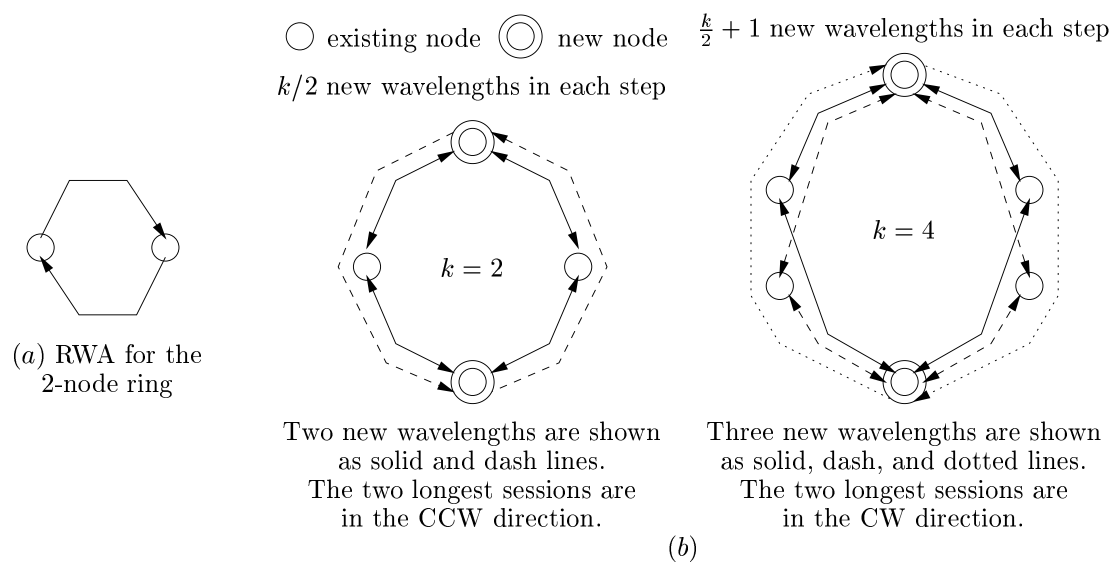


Figure 3-28: RWA update step for the k-node ring, where k is even.

The RWA of the new sessions (terminated at the new nodes) is shown in figure 3-28b. In particular, the RWA is based on shortest path routing and efficient wavelength reuse such that each new wavelength is used on every fiber, except for the wavelength used by the two longest sessions (between the two new nodes). We choose to route the two longest sessions in the counterclockwise (CCW) direction when k is not divisible by 4 and in the CW direction when k is divisible by 4. Notice that, in each step when k is not divisible by 4, we can reuse the wavelength which is used only in the CW direction (for the two longest sessions) in the previous step. This is the reason for the above RWA for the 2-node ring. It follows that the number of new wavelengths used in each step is $k/2+1$ if k is divisible by 4, and $k/2$ if k is not divisible by 4.

By repeating the update step until we obtain the N-node ring, the number of wavelengths used is, for N divisible by 4,[5]

$$1+\left(\frac{2}{2}\right)+\left(\frac{4}{2}+1\right)+\left(\frac{6}{2}\right)+\left(\frac{8}{2}+1\right)+...+\left(\frac{N-2}{2}\right) \;=\; \frac{N^2}{8} \;=\; \left\lceil\frac{N^2}{8}\right\rceil.$$

It follows that $W_{s,1} \leq \lceil N^2/8 \rceil$ for N even and divisible by 4.

For N not divisible by 4, the number of wavelengths used is [6]

$$1+\left(\frac{2}{2}\right)+\left(\frac{4}{2}+1\right)+\left(\frac{6}{2}\right)+\left(\frac{8}{2}+1\right)+...+\left(\frac{N-2}{2}+1\right) \;=\; \frac{N^2}{8}+\frac{1}{2} \;=\; \left\lceil\frac{N^2}{8}\right\rceil.$$

Therefore, $W_{s,1} \leq \lceil N^2/8 \rceil$ for all N even.

For N even and $l \geq 1$ odd, we use a procedure similar to the case with $l = 1$. In this case, we route l session pairs instead of one session pair between each node pair. For the 2-node ring, we route $(l+1)/2$ session pairs in the CW direction and the other $(l-1)/2$ session pairs in the CCW direction. In each RWA update step, for k divisible by 4, we route $(l+1)/2$ of the longest session pairs in the CW direction and the other $(l-1)/2$ of the longest session pairs in the CCW direction. For k not divisible by 4, we route $(l+1)/2$ of the longest session pairs in the CCW direction and the other $(l-1)/2$ of the longest session pairs in the CW direction. When k is divisible by 4, there is one new wavelength used only in the CW direction. When k is not divisible by 4, we can reuse the wavelength used only in the CW direction in the previous step. It follows that the number of new wavelengths used in each step is $lk/2+(l+1)/2$ if k is divisible by 4, and $lk/2+(l-1)/2$ if k is not divisible by 4. After repeating the update step until we obtain the N-node ring, the number of wavelengths used is, for N divisible by 4,

$$\frac{l+1}{2}+\left(l\frac{2}{2}+\frac{l-1}{2}\right)+\left(l\frac{4}{2}+\frac{l+1}{2}\right)+...+\left(l\frac{N-2}{2}+\frac{l-1}{2}\right) \;=\; l\frac{N^2}{8} \;=\; \left\lceil l\frac{N^2}{8}\right\rceil.$$

For N not divisible by 4, the number of wavelengths used is

$$\frac{l+1}{2}+\left(l\frac{2}{2}+\frac{l-1}{2}\right)+\left(l\frac{4}{2}+\frac{l+1}{2}\right)+...+\left(l\frac{N-2}{2}+\frac{l+1}{2}\right) \;=\; l\frac{N^2}{8}+\frac{1}{2} \;=\; \left\lceil l\frac{N^2}{8}\right\rceil.$$

Therefore, $W_{s,l} \leq \lceil lN^2/8 \rceil$ for all N even and l odd.

---

[5]When N is divisible by 4, N^2 is divisible by 8. Thus $N^2/8$ is an integer, i.e. $\lceil N^2/8 \rceil = N^2/8$.

[6]When N is not divisible by 4, $N = 4m+2$ for some positive integer m. We can express $N^2/8$ as $(4m+2)^2/8 = 2m^2+2m+1/2$, from which it is easy to see that $\lceil N^2/8 \rceil = N^2/8+1/2$.

Finally, for N even and $l \geq 1$ even, we use a procedure similar to the case with l odd. However, in each step, we can route $l/2$ of the longest session pairs in the CW direction and the other $l/2$ of the longest session pairs in the CCW direction. Thus, each new wavelength can be used on every fiber. It follows that the number of new wavelengths used in each step is $lk/2+l/2$. After repeating the update step until we obtain the N-node ring, the number of wavelengths used is

$$\frac{l}{2} + \left(l\frac{2}{2} + \frac{l}{2}\right) + \left(l\frac{4}{2} + \frac{l}{2}\right) + \ldots + \left(l\frac{N-2}{2} + \frac{l}{2}\right) \;=\; l\frac{N^2}{8} \;=\; \left\lceil l\frac{N^2}{8} \right\rceil .$$

In conclusion, we have shown that $W_{s,l} \leq \lceil lN^2/8 \rceil$ for all N even and all $l \geq 1$.

We summarize the discussion in this section in the following theorem.

Theorem 2 *In the bidirectional ring topology with l-uniform traffic among N nodes, $W_{s,l}$ is given by*

$$W_{s,l} \;=\; L_{s,l} \;=\; \begin{cases} l(N^2-1)/8, & N \text{ odd,} \\ \lceil lN^2/8 \rceil, & N \text{ even.} \end{cases}$$

Theorem 2 tells us that wavelength conversion cannot decrease the wavelength requirement for l-uniform traffic in a bidirectional ring topology.

3.3 2D Torus Topologies

In this section, we discuss the RWA for l-uniform traffic in a two-dimensional (2D) torus topology. Figure 3-29 shows the $R \times C$ torus topology with $N = RC$ end nodes, where R and C are the numbers of rows and columns respectively.

Let $L_{s,l}$ denote the minimum number of wavelengths which, if provided in each fiber, can support l-uniform traffic given full wavelength conversion at all nodes. To derive a lower bound on $L_{s,l}$, we use the link counting bound described in section 3.2. Let H be the sum of the number of hops traversed by each of the sessions under shortest path routing, and F be the number of fibers in the network. Then some fiber must support at least $\lceil H/F \rceil$ wavelengths, i.e. $L_{s,l} \geq \lceil H/F \rceil$. For l-uniform traffic in the $R \times C$ torus topology, it is straightforward to derive H as shown below.

$R = 4$,
$C = 5$,
$N = 20$

Each link is bidirectional.

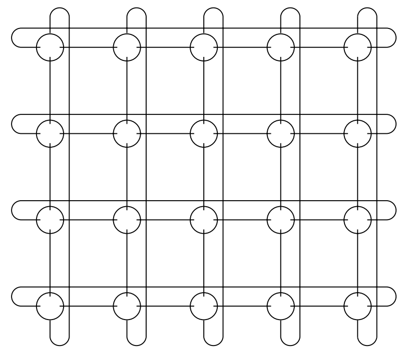


Figure 3-29: The $R \times C$ torus topology with $N = RC$ end nodes.

$$H = \begin{cases} lR^3C^2/4 + lR^2C^3/4, & R \text{ even}, C \text{ even}, \\ lR^3C^2/4 + lR^2C(C^2-1)/4, & R \text{ even}, C \text{ odd}, \\ lR(R^2-1)C^2/4 + lR^2C^3/4, & R \text{ odd}, C \text{ even}, \\ lR(R^2-1)C^2/4 + lR^2C(C^2-1)/4, & R \text{ odd}, C \text{ odd}. \end{cases}$$

Since $F = 4RC$, it follows that

$$L_{s,l} \geq \left\lceil \frac{H}{F} \right\rceil = \begin{cases} \lceil lRC(R+C)/16 \rceil, & R \text{ even}, C \text{ even}, \\ \lceil lRC(R+C-\frac{1}{C})/16 \rceil, & R \text{ even}, C \text{ odd}, \\ \lceil lRC(R+C-\frac{1}{R})/16 \rceil, & R \text{ odd}, C \text{ even}, \\ \lceil lRC(R+C-\frac{1}{R}-\frac{1}{C})/16 \rceil, & R \text{ odd}, C \text{ odd}. \end{cases} \quad (3.2)$$

Let $W_{s,l}$ denote the minimum number of wavelengths which, if provided in each fiber, can support l-uniform traffic with no wavelength conversion. The derivation of $W_{s,l}$ was previously studied in [Mar+93]. In [Mar+93], the authors claim that, if both R and C are divisible by 4, then there exists, by construction, an RWA scheme which uses the number of wavelengths equal to the lower bound of $L_{s,l}$ given in (3.2), i.e. $W_{s,l} = L_{s,l} = \lceil lRC(R+C)/16 \rceil = lRC(R+C)/16$. The derivation of $W_{s,l}$ for general values of R and C remains to be investigated.

3.4 Binary Hypercube Topologies

In this section, we solve the RWA problem for l-uniform traffic in a binary hypercube topology. A binary hypercube topology contains $N = 2^n$ end nodes for some positive integer n. Figure 3-30 illustrates the cases with $N = 4$ and $N = 8$. The N end nodes can be labeled using n-bit binary strings. Two nodes are adjacent if their labels differ in only one bit. Unlike all the previous topologies in which the maximum node degree can be kept constant as N grows large, a binary hypercube has its node degree equal to n, which increases logarithmically with N.

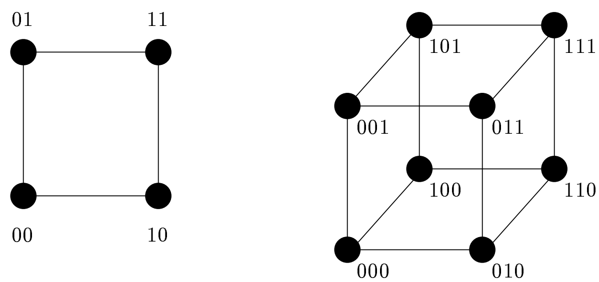


Figure 3-30: Binary hypercube topologies

Let us consider the RWA problem for 1-uniform traffic. The results can later be extended to l-uniform traffic in a straightforward fashion. Let $L_{s,1}$ denote the minimum number of wavelengths which, if provided in each fiber, can support 1-uniform traffic given full wavelength conversion at all nodes. We first derive a lower bound on $L_{s,1}$. Partition the nodes into two disjoint subsets, one with the nodes whose labels start with bit 0 and the other with the nodes whose labels start with bit 1. Note that each subset contains $N/2$ nodes. There are $N/2$ fibers leaving from one subset to the other. For 1-uniform traffic, the amount of traffic from one subset to the other is $N/2 \times N/2 = N^2/4$ wavelengths. It follows that one fiber connecting the two sets of nodes must support at least $(N^2/4)/(N/2) = N/2$ wavelengths. Therefore, $L_{s,1} \geq N/2$.

Let $W_{s,1}$ denote the minimum number of wavelengths which, if provided in each fiber, can support 1-uniform traffic with no wavelength conversion. To derive an upper bound on $W_{s,1}$, we construct an RWA algorithm. Our algorithm, which uses $N/2$ wavelengths in each fiber, implies that $W_{s,1} = L_{s,1} = N/2$.

To route each session, we use a fixed routing scheme which we refer to as *label matching routing*. In label matching routing, a route from a source to a destination is obtained by changing the source label one bit at a time from the most significant bit to obtain the destination label. For example, a route from source 000 to destination 111 goes through the following nodes: 000 $\rightarrow$ 100 $\rightarrow$ 110 $\rightarrow$ 111.

Using label matching routing, it is still necessary to perform wavelength assignment (WA). We shall express our WA scheme as a WA code.[7] To do so, we make a few useful observations. Define a *type-i link*, $0 \leq i \leq n-1$, to be a unidirectional link, or equivalently a fiber, between two nodes whose labels differ only in the i^{th} significant bit. For example, the link from node 100 to node 101 is a type-0 link, while the link from node 110 to node 010 is a type-2 link. Figure 3-31 illustrates the following observations.

1. Based on label matching routing, a type-i link is used to reach 2^i destinations by any source which utilizes it.
2. The type-i link from node n_1 to node n_2 is used by the 2^{n-i-1} sources whose lowest $i+1$ bits are the same as those in the label of n_1. Consequently, the labels of those 2^{n-i-1} sources, when viewed as integers, are each separated by an integer multiple of 2^{i+1}.

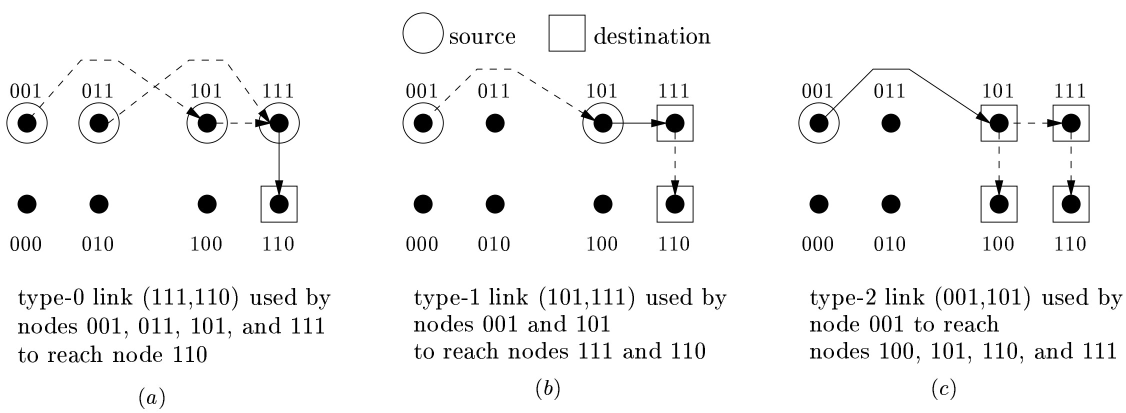


Figure 3-31: Properties of type-i links.

The above observations suggest the following WA code construction. From the above observations, a fiber can be shared only by sources which all have the least significant bit equal to 0, or some sources all of which have the least significant bit equal to 1. In other words, a source with the least significant bit equal to 0 never shares the same fiber with any source with the least significant bit equal to 1. For example, source 000 and source 001 never use the same fiber since their least

[7] The definitions of a WA code and a WA vector are given in section 3.1.1.

significant bits differ. It follows that wavelength collision is avoided even if we assign the set of WA vectors $\mathbf{v}_0, \mathbf{v}_2, ..., \mathbf{v}_{N-2}$ independently from the set of WA vectors $\mathbf{v}_1, \mathbf{v}_3, ..., \mathbf{v}_{N-1}$. We shall construct $\mathbf{v}_0, \mathbf{v}_2, ..., \mathbf{v}_{N-2}$ and use the same construction for $\mathbf{v}_1, \mathbf{v}_3, ..., \mathbf{v}_{N-1}$.

We shall use $N/2$ wavelengths in each fiber. For simplicity, we choose each WA vector to contain consecutive wavelength indices in an increasing order modulo $N/2$, e.g. $[1, 2, ..., \frac{N}{2}, 1, 2, ..., \frac{N}{2}]$. Notice that each wavelength index may appear more than once in a given WA vector. This multiplicity does not pose a problem since, according to label matching routing, the routes from each source to the nodes whose labels start with bit 0 never overlap with the routes from the same source to the nodes whose labels start with bit 1. Therefore, we can repeat the same set of entries twice in each WA vector. We choose to assign the first WA vector as $\mathbf{v}'_0 = [1, 2, ..., \frac{N}{2}, 1, 2, ..., \frac{N}{2}]$. We use the notation $\mathbf{v}'_0$ instead of $\mathbf{v}_0$ because we shall eventually construct $\mathbf{v}_0$ from $\mathbf{v}'_0$ by using 0 as a dummy wavelength index for the self-traffic entry, i.e. $\mathbf{v}_0 = [0, 2, ..., \frac{N}{2}, 1, 2, ..., \frac{N}{2}]$.

From the above observations, each type-i link must carry 2^i wavelengths from each of the sources whose labels, when viewed as integers, are each separated by an integer multiple of 2^{i+1}. This observation suggests shifting the entries of $\mathbf{v}'_0$ by 1 unit to create $\mathbf{v}'_2$, shift the entries of $\mathbf{v}'_2$ by 1 unit to create $\mathbf{v}'_4$, and so on. More explicitly,

$$\mathbf{v}'_0 = \begin{bmatrix} 1 \\ 2 \\ \vdots \\ \frac{N}{2} \\ 1 \\ 2 \\ \vdots \\ \frac{N}{2} \end{bmatrix}, \quad \mathbf{v}'_2 = \begin{bmatrix} 2 \\ \vdots \\ \frac{N}{2} \\ 1 \\ 2 \\ \vdots \\ \frac{N}{2} \\ 1 \end{bmatrix}, \quad \cdots, \quad \mathbf{v}'_{N-2} = \begin{bmatrix} \frac{N}{2} \\ 1 \\ 2 \\ \vdots \\ \frac{N}{2} \\ 1 \\ \vdots \\ \frac{N}{2} - 1 \end{bmatrix}.$$

It follows that two sources which are 2^{i+1} units apart have their WA vectors offset by 2^i units. Thus, based on this scheme, no wavelength collision occurs on any type-i link, $0 \leq i \leq n-1$.

We use the same construction to assign $\mathbf{v}'_1 = \mathbf{v}'_0, \mathbf{v}'_3 = \mathbf{v}'_2, ..., \mathbf{v}'_{N-1} = \mathbf{v}'_{N-2}$. Given a square matrix C, let C^0 denote the same matrix but with all its diagonal entries set to 0. The WA code can be expressed as

$$
\begin{bmatrix} | & | & & | \\ \mathbf{v}_0 & \mathbf{v}_1 & \cdots & \mathbf{v}_{N-1} \\ | & | & & | \end{bmatrix} = \begin{bmatrix} | & | & & | \\ \mathbf{v}'_0 & \mathbf{v}'_1 & \cdots & \mathbf{v}'_{N-1} \\ | & | & & | \end{bmatrix}^{0} = \begin{bmatrix} 1 & 1 & 2 & 2 & & \frac{N}{2} & \frac{N}{2} \\ \vdots & \vdots & \vdots & \vdots & & 1 & 1 \\ & & \frac{N}{2} & \frac{N}{2} & & \vdots & \vdots \\ \frac{N}{2} & \frac{N}{2} & 1 & 1 & \cdots & \frac{N}{2} & \frac{N}{2} \\ 1 & 1 & \vdots & \vdots & & 1 & 1 \\ \vdots & \vdots & \frac{N}{2} & \frac{N}{2} & & \vdots & \vdots \\ \frac{N}{2} & \frac{N}{2} & 1 & 1 & & \frac{N}{2}-1 & \frac{N}{2}-1 \end{bmatrix}^{0} .
$$

To express the above WA code more compactly, define $\tilde{\mathbf{c}}_N = [1, ..., \frac{N}{2}, 1, ..., \frac{N}{2}]$. In addition, for $0 \le j \le N-1$, define $\tilde{\mathbf{c}}_N^{(j)}$ to be the vector with the entries of $\tilde{\mathbf{c}}_N$ shifted up by j units. For example, $\tilde{\mathbf{c}}_8^{(0)} = [1,2,3,4,1,2,3,4]$ and $\tilde{\mathbf{c}}_8^{(3)} = [4,1,2,3,4,1,2,3]$. Using this notation, we can express the WA code as

$$
\begin{bmatrix} | & | & & | \\ \mathbf{v}_0 & \mathbf{v}_1 & \cdots & \mathbf{v}_{N-1} \\ | & | & & | \end{bmatrix} = \begin{bmatrix} | & | & | & | & & | & | \\ \tilde{\mathbf{c}}_N^{(0)} & \tilde{\mathbf{c}}_N^{(0)} & \tilde{\mathbf{c}}_N^{(1)} & \tilde{\mathbf{c}}_N^{(1)} & \cdots & \tilde{\mathbf{c}}_N^{(N/2-1)} & \tilde{\mathbf{c}}_N^{(N/2-1)} \\ | & | & | & | & & | & | \end{bmatrix}^{0} .
$$

The construction of our RWA algorithm implies the following theorem.

Theorem 3 *In a binary hypercube topology with 1-uniform traffic among N nodes, where $N = 2^n$ for some positive integer n, $W_{s,1}$ is given by*

$$
W_{s,1} \;=\; L_{s,1} \;=\; N/2.
$$

Let $L_{s,l}$ and $W_{s,l}$ denote the minimum number of wavelengths which, if provided in each fiber, can support l-uniform traffic with full wavelength conversion at all nodes and without wavelength conversion respectively. It is a simple extension to establish that $W_{s,l} = lW_{s,1} = lN/2$. First, we can use the same argument as in the derivation for the lower bound of $L_{s,1}$ to show that one of the fibers connecting the set of nodes whose labels start with bit 0 and the set of nodes whose labels start with bit 1 must carry at least $lN/2$ wavelengths. Thus, $L_{s,l} \ge lN/2$. To show that $W_{s,l} \le N/2$, we apply the RWA algorithm l times on l different sets each with $N/2$ wavelengths. We state the result formally as a corollary to theorem 3.

Corollary 2 *In a binary hypercube topology with l-uniform traffic among N nodes, where $N = 2^n$ for some positive integer n, $W_{s,l}$ is given by*

$$W_{s,l} \;=\; L_{s,l} \;=\; lN/2.$$

The following example illustrates our RWA algorithm in detail.

Example 4 Consider a binary hypercube with $N = 8$. Theorem 3 states that 4 wavelengths suffice to support 1-uniform traffic. With label matching routing, the corresponding WA code based on our RWA algorithm is given below.

$$\begin{bmatrix} | & | & & | \\ \mathbf{v}_0 & \mathbf{v}_1 & \cdots & \mathbf{v}_7 \\ | & | & & | \end{bmatrix} = \begin{bmatrix} 0 & 1 & 2 & 2 & 3 & 3 & 4 & 4 \\ 2 & 0 & 3 & 3 & 4 & 4 & 1 & 1 \\ 3 & 3 & 0 & 4 & 1 & 1 & 2 & 2 \\ 4 & 4 & 1 & 0 & 2 & 2 & 3 & 3 \\ 1 & 1 & 2 & 2 & 0 & 3 & 4 & 4 \\ 2 & 2 & 3 & 3 & 4 & 0 & 1 & 1 \\ 3 & 3 & 4 & 4 & 1 & 1 & 0 & 2 \\ 4 & 4 & 1 & 1 & 2 & 2 & 3 & 0 \end{bmatrix}$$

Figure 3-32 explicitly illustrates the routes and wavelengths of the sessions transmitted by node 001. For example, node 001 transmits to node 000 on wavelength 1. Node 001 transmits to node 101 on wavelength 2.

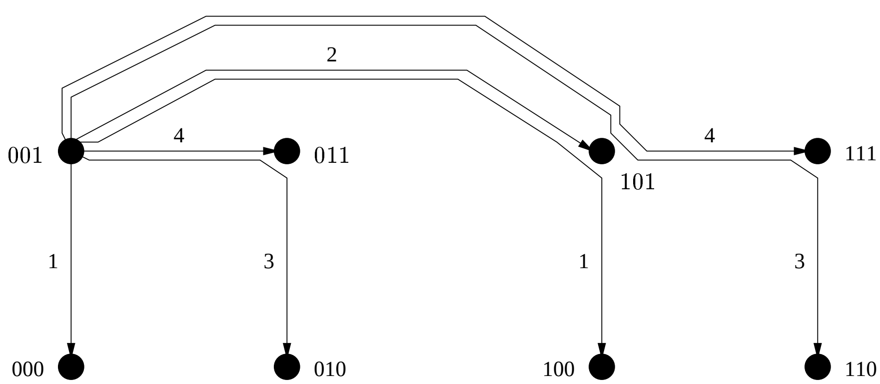


Figure 3-32: Routes and wavelengths of the sessions from node 001.

3.5 Arbitrary Topologies

In this section, we discuss the RWA problem for l-uniform traffic in an arbitrary topology. Let $L_{s,l}$ and $W_{s,l}$ denote the minimum number of wavelengths which, if provided in each fiber, can support l-uniform traffic with full wavelength conversion at all nodes and without wavelength conversion respectively.

We shall describe two lower bounds on $L_{s,l}$ and two upper bounds on $W_{s,l}$. Since $L_{s,l} \leq W_{s,l}$, given a lower bound on $L_{s,l}$ and an upper bound on $W_{s,l}$, the actual value of $W_{s,l}$ lies between the two bounds.

3.5.1 Lower Bound on $L_{s,l}$: the Link Counting Bound

To derive a lower bound on $L_{s,l}$, we can use the link counting bound from [Pan92] which is described in section 3.2. Let H be the sum of the number of hops traversed by each of the sessions under shortest path routing, and F be the number of fibers in the network. Then some fiber must support at least $\lceil H/F \rceil$ wavelengths, and thus $L_{s,l} \geq \lceil H/F \rceil$.

The link counting bound is reasonably tight when there exists a routing scheme which distributes traffic evenly on all the fibers. For example, for l-uniform traffic in a bidirectional ring, an RWA scheme described in section 3.2 uses effectively the same number of wavelengths on all the fibers. Thus, the link counting bound is tight in this case.

As an example in which the link counting bound is not tight, consider the N-node binary tree topology in example 3, where $N = 2^n$ for some positive integer n. From corollary 1, we know that $L_{s,l} = lN^2/4$. To use the link counting bound, it is straightforward to derive H as shown below

$$H = 2lN[N(\log_2 N - 1) + 1].$$

Since $F = 4(N-1)$, it follows that

$$L_{s,l} \geq \left\lceil \frac{H}{F} \right\rceil = \frac{l}{2}\frac{N}{N-1}[N(\log_2 N - 1) + 1],$$

which is approximately $(lN \log_2 N)/2$ for large N. Since $L_{s,l} = lN^2/4$, the link counting bound is not tight in this example.

3.5.2 Lower Bound on $L_{s,l}$: the Cut Set Bound

A cut set in a connected network is a subset of (bidirectional) links whose removal results in two disjoint connected subgraphs.[8] A lower bound on $L_{s,l}$ can be obtained by forming a cut set in the network and determine the minimum amount of traffic that one fiber across the cut has to support [BB97].

Consider a cut set $\mathcal{C}$ which separates the end nodes into two sets $\mathcal{N}_{\mathcal{C},1}$ and $\mathcal{N}_{\mathcal{C},2}$. The amount of traffic (in wavelengths) across this cut from $\mathcal{N}_{\mathcal{C},1}$ to $\mathcal{N}_{\mathcal{C},2}$ is $l|\mathcal{N}_{\mathcal{C},1}||\mathcal{N}_{\mathcal{C},2}|$. Since there are $|\mathcal{C}|$ fibers from $\mathcal{N}_{\mathcal{C},1}$ to $\mathcal{N}_{\mathcal{C},2}$, one fiber across this cut must support at least $\lceil l|\mathcal{N}_{\mathcal{C},1}||\mathcal{N}_{\mathcal{C},2}|/|\mathcal{C}|\rceil$ wavelengths, i.e. $L_{s,l} \geq \lceil l|\mathcal{N}_{\mathcal{C},1}||\mathcal{N}_{\mathcal{C},2}|/|\mathcal{C}|\rceil$. To tighten the bound, we search for the cut which yields the maximum lower bound, i.e.

$$L_{s,l} \;\geq\; \max_{\mathcal{C}} \left\lceil \frac{l|\mathcal{N}_{\mathcal{C},1}||\mathcal{N}_{\mathcal{C},2}|}{|\mathcal{C}|} \right\rceil . \tag{3.3}$$

We shall refer to the above lower bound of $L_{s,l}$ as the *cut set bound.* Notice that, in section 3.2, we use the cut set bound to define the value of w^* in (3.1). In a tree topology, the cut set is a single link, and the bottleneck link yields the cut set bound. From corollary 1, we know that the cut set bound is tight for a tree topology.

Interestingly, for other topologies we consider, the cut set bound is also tight. For example, the cut set bound for an N-node bidirectional ring is given by

$$L_{s,l} \;\geq\; \begin{cases} \left\lceil \frac{l\frac{N-1}{2}\frac{N+1}{2}}{2} \right\rceil \;=\; l\frac{N^2-1}{8}, & \text{N odd,} \\ \left\lceil \frac{l\frac{N}{2}\frac{N}{2}}{2} \right\rceil \;=\; \left\lceil l\frac{N^2}{8} \right\rceil, & \text{N even.} \end{cases}$$

From theorem 2, the cut set bound is tight. As another example, in section 3.4, we have used the cut set bound argument to derive the lower bound $L_{s,l} \geq lN/2$ for the N-node binary hypercube. From corollary 2, the cut set bound is tight.

To our knowledge, there is no known topology for which the cut set bound is not tight for l-uniform traffic. On the other hand, there is no known proof that the cut set bound is tight for l-uniform traffic in an arbitrary topology. We state this problem as an open problem for future research below.

[8] Equivalently, a cut set in a connected network is a subset of links such that the network is no longer connected after its removal, but is still connected after a removal of its strict subset.

Problem 3 *For l-uniform traffic in an arbitrary topology, determine whether or not the cut set bound in (3.3) is always tight.*

3.5.3 Upper Bound on $W_{s,l}$: the Embedded Tree Bound

In this subsection, we shall return to the wavelength assignment (WA) problem for l-uniform traffic in an arbitrary tree topology considered in section 3.1 and relax the assumption that only leaf nodes are end nodes. This relaxation allows us to embed a tree topology in an arbitrary connected topology. The off-line tree WA algorithm can then be used to derive an upper bound on $W_{s,l}$. As a specific example, figure 3-33a shows an arbitrary topology. One possible embedded tree is shown in figure 3-33b. Note that nodes 2, 4, and 5 are non-leaf nodes.

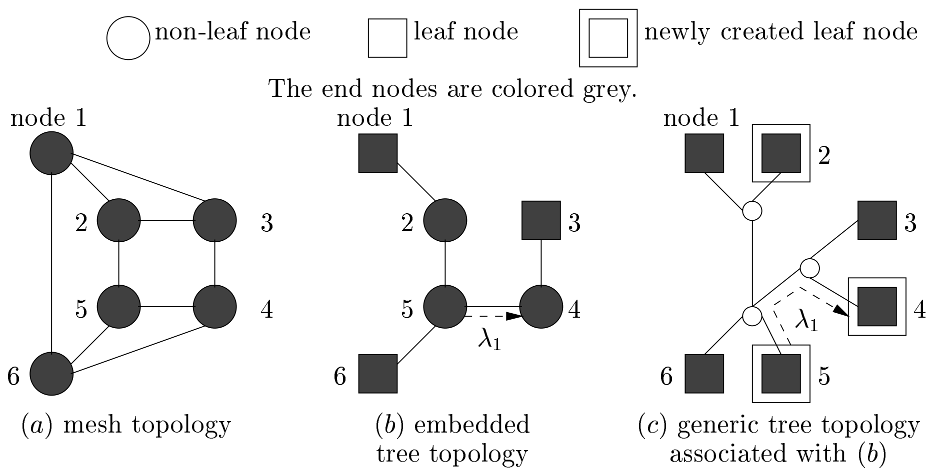


Figure 3-33: Embedded tree topology and its associated generic tree topology.

Given an embedded tree topology with non-leaf end nodes, we can create the associated *generic tree topology* with no non-leaf end node as follows. For each non-leaf end node, create a new leaf node attached to it. The new leaf node is an end node, while the existing non-leaf node is no longer an end node. For example, figure 3-33c shows the generic tree topology associated with the embedded tree topology in figure 3-33b. In particular, there are three new leaf nodes in figure 3-33c created from the three non-leaf end nodes in figure 3-33b.

The following theorem states that the minimum number of wavelengths for l-allowable traffic for the generic tree, denoted by $W_{s,l,g}$, is the same as for the embedded tree, denoted by $W_{s,l,e}$.

Theorem 4 *For l-uniform traffic, the wavelength requirements for an embedded tree and for its associated generic tree are the same, i.e.* $W_{s,l,e} = W_{s,l,g}$.

Proof: We first argue that $W_{s,l,e} \leq W_{s,l,g}$. Observe that, for the same traffic matrix, the WA for the generic tree can be used for the embedded tree as described next. Each lightpath in the generic tree can be mapped to an identical lightpath in the embedded tree except for all the newly created links in the generic tree. For example, the three-hop lightpath on wavelength λ_1 from leaf node 5 to leaf node 4 in figure 3-33c is mapped to the one-hop lightpath on λ_1 from node 5 to node 4 in figure 3-33b. It follows that $W_{s,l,e} \leq W_{s,l,g}$.

We now argue that $W_{s,l,e} \geq W_{s,l,g}$. From the definition of w^* for a generic tree given in (3.1), we claim that the bottleneck link e^* in the generic tree can always be chosen so that it is not one of the newly created links as compared with the embedded tree. To see this, note that any newly created link separates a single end node from all the other end nodes, and the flow across a fiber in this link is equal to $l[1(N-1)] = l(N-1)$ wavelengths, which is the minimum possible fiber load in a tree with l-uniform traffic among N end nodes. Thus, if a newly created link can serve as the bottleneck link, so can any existing link. (In fact, in this case, the generic tree topology is necessarily a star.) With the above choice of the bottleneck link e^*, link e^* exists in the embedded tree and up to lw^* wavelengths of traffic can traverse across it in one direction. It follows that $W_{s,l,e} \geq lw^*$. Since $W_{s,l,g} = lw^*$, we have shown that $W_{s,l,e} \geq W_{s,l,g}$. In conclusion, we have proved that $W_{s,l,e} = W_{s,l,g}$. □

Theorem 4 tells us that the definition of w^* in (3.1) yields the minimum number of wavelengths for l-uniform traffic in an arbitrary tree topology with non-leaf end nodes. After we embed a tree topology in a given arbitrary topology, the value of $W_{s,l,e}$ for the embedded tree can be used in an upper bound on $W_{s,l}$. We summarize the discussion below as a corollary to theorem 4.

Corollary 3 *For l-uniform traffic, the generic tree associated with the embedded tree can be used to obtain the embedded tree bound in place of the embedded tree, i.e.* $W_{s,l} \leq W_{s,l,g} = W_{s,l,e}$.

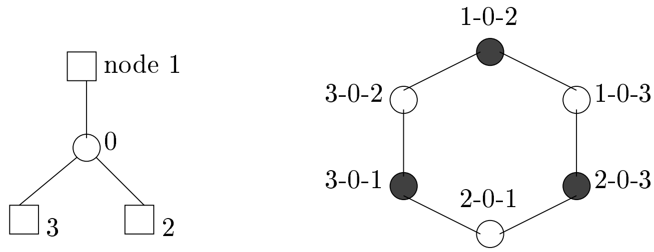
For example, the value of $W_{s,l,g}$ for the generic tree associated with the embedded tree in figure 3-33b is equal to $8l$. Thus, for the topology given in figure 3-33a, $W_{s,l} \leq 8l$.

We shall refer to the upper bound on $W_{s,l}$ obtained in this fashion as the *embedded tree bound.* The embedded tree bound is a reasonable estimate on $W_{s,l}$ when the network nodes are sparsely connected. However, for a densely connected network, it can perform poorly. For example, consider the N-node binary hypercube. We know from corollary 2 that $W_{s,l} = lN/2$. From statement 2 of lemma 1, any embedded tree with N end nodes has $w^* \geq \frac{1}{d^*}(1 - \frac{1}{d^*})N^2$, where d^* is the degree of the bottleneck node. Since d^* in the N-node binary hypercube is equal to $\log_2 N$, it follows that the embedded tree bound lw^* is at least $\frac{1}{\log_2 N}(1 - \frac{1}{\log_2 N})lN^2$, which is approximately $lN^2/(\log_2 N)$ for large N. Since $W_{s,l} = lN/2$, the embedded tree bound is not tight in this example.

3.5.4 Upper Bound on $W_{s,l}$ in term of $L_{s,l}$: the Graph Coloring Bound

In this section, we discuss an upper bound of $W_{s,l}$ in term of $L_{s,l}$ using a known argument in [Agg+96]. Given the routing assignment for all the sessions, i.e. the routes of all the lightpaths, such that the maximum load in a fiber is $L_{s,l}$ wavelengths, we derive an upper bound on $W_{s,l}$ by keeping the same routing assignment and performing wavelength assignment (WA). In [CGK92], it is shown that the WA problem can be reduced to a graph coloring problem in which we try to color all the nodes in the new graph so that no adjacent nodes have the same color using the minimum number of colors. More specifically, given a network topology and the routes of all the lightpaths, we can create the corresponding *path graph* as follows. Each lightpath is mapped one-to-one to a node in the path graph. Two nodes in the path graph are connected if and only if the two corresponding lightpaths share a fiber. For example, consider the 3-node star network with 1-uniform traffic in figure 3-34a. Note that there is no routing problem in this example. The corresponding path graph is shown in figure 3-34b. In the path graph, there are in total six nodes corresponding to the six lightpaths under 1-uniform traffic. We denote each node in the path graph by its route, e.g. node 1-0-2 refers to the lightpath of session (1,2). Node 1-0-2 is adjacent to node 1-0-3 since they share the fiber from on link 1-0.

In this specific example, the graph coloring problem is to color all the six nodes so that no adjacent nodes have the same color using the minimum number of colors. It is easy to see that the minimum number of colors required in this example is 2. After coloring the nodes in the path graph, we map the node colors one-to-one to the wavelengths which we assign to the corresponding lightpaths. For example, figure 3-34b shows the node colors after the graph coloring problem is



(a) 3-node star network (b) path graph for 1-uniform traffic

Figure 3-34: The path graph for the 3-node star with 1-uniform traffic.

solved. From the node colors, we assign the first wavelength to lightpaths 1-0-2, 2-0-3, and 3-0-1, and the second wavelength to lightpaths 1-0-3, 2-0-1, and 3-0-2.

In general, the graph coloring problem is hard to solve and is known to be NP-complete [GJ79]. However, it is known that any graph with maximum node degree d can be colored with $d+1$ colors [Ber85]. Given the maximum fiber load $L_{s,l}$ and the length (in hops) of the longest lightpath h, each lightpath shares a fiber with at most $h(L_{s,l}-1)$ other lightpaths. It follows that the maximum node degree in the path graph is $h(L_{s,l}-1)$. Therefore, $h(L_{s,l}-1)+1$ wavelengths are sufficient to support l-uniform traffic, i.e.

$$W_{s,l} \leq h(L_{s,l}-1)+1. \tag{3.4}$$

We shall refer to the above upper bound on $W_{s,l}$ as the *graph coloring bound*. Unfortunately, the graph coloring bound tends to be quite pessimistic for l-uniform traffic. For example, consider the N-node bidirectional ring topology. We know from theorem 2 that $W_{s,l} = L_{s,l}$. However, the graph coloring bound in (3.4) yields

$$W_{s,l} \leq \begin{cases} \frac{N}{2}(L_{s,l}-1)+1, & \text{N even,} \\ \frac{N-1}{2}(L_{s,l}-1)+1, & \text{N odd,} \end{cases}$$

which is clearly not tight.

Interestingly, for all the topologies in which we can obtain closed form expressions for $L_{s,l}$ and $W_{s,l}$, we see that $W_{s,l} = L_{s,l}$. In particular, we have seen that $W_{s,l} = L_{s,l}$ for arbitrary tree, bidirectional ring, and binary hypercube topologies.

To our knowledge, there is no known topology with $W_{s,l} > L_{s,l}$. On the other hand, there is no known proof that $W_{s,l} = L_{s,l}$ in any arbitrary topology. We state this problem as an open problem for future research below.

Problem 4 *For l-uniform traffic in an arbitrary topology, determine whether or not* $W_{s,l} = L_{s,l}$.

Chapter 4

RWA for Dynamic k-Allowable Traffic

In this chapter, we study the routing and wavelength assignment (RWA) problem for $\mathbf{k}$-allowable traffic, where $\mathbf{k} = [k_1, k_2, ..., k_N]$ and N is the number of end nodes in the network. In $\mathbf{k}$-allowable traffic, node i, $1 \leq i \leq N$, transmits at most k_i wavelengths and receives at most k_i wavelengths. Let $W_{d,\mathbf{k}}$ denote the minimum number of wavelengths which, if provided in each fiber, can support dynamic $\mathbf{k}$-allowable traffic in a rearrangeably nonblocking fashion with no wavelength conversion. As in the case of l-uniform traffic, we solve the RWA problem in a few special cases with the hope of extending our analytical techniques to obtain a good general bound on the value of $W_{d,\mathbf{k}}$ for any given topology. The specific topologies we shall consider include arbitrary tree topologies, a bidirectional ring, a two-dimensional (2D) torus, and a binary hypercube.

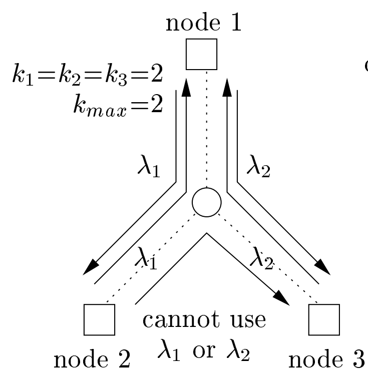
Let $L_{d,\mathbf{k}}$ denote the minimum number of wavelengths which, if provided in each fiber, can support dynamic $\mathbf{k}$-allowable traffic in a rearrangeably nonblocking fashion given full wavelength conversion at all nodes. It is clear that $L_{d,\mathbf{k}} \leq W_{d,\mathbf{k}}$ for any given network topology. For convenience, define *symmetric k-allowable traffic* to be the $\mathbf{k}$-allowable traffic in which all the k_i's are equal to k. Throughout the chapter, we make the following assumption on $\mathbf{k}$-allowable traffic.

Assumption 1 *Let $k_{max} = \max_{1\leq i\leq N} k_i$. Assume that $k_{max} \leq \left(\sum_{1\leq i\leq N} k_i\right)/2$.*

Assumption 1 is reasonable since the node with k_{max} fully tunable transmitters (receivers) can transmit (receive) at most $(\sum_{1\leq i\leq N} k_i) - k_{max}$ wavelengths to (from) all the other nodes. Therefore, k_{max} need be no greater than $(\sum_{1\leq i\leq N} k_i) - k_{max}$, yielding the condition in assumption 1.

4.1 Star Topologies

In this section, we solve the RWA problem for $\mathbf{k}$-allowable traffic in a star topology. In the next section, we extend the results to the case of arbitrary tree topologies. Figure 4-1 shows an example of a star topology with 3 end nodes connected through a central hub. Since there is a unique route for each traffic session, there is no routing problem. Thus, we only have to perform wavelength assignment (WA) in the RWA problem.



sequence of session arrivals	corresponding sequence of WA steps
(1,2)	(1,2) on λ_1
(2,1)	(2,1) on λ_1
(1,3)	(1,3) on λ_2
(3,1)	(3,1) on λ_2
(2,3)	(2,3) not on λ_1 or λ_2

(i, j) denotes a session from node i to node j.

Figure 4-1: An example in which a greedy approach requires more than k_{max} wavelengths.

Let $L_{d,\mathbf{k}}$ and $W_{d,\mathbf{k}}$ denote the minimum number of wavelengths which, if provided in each fiber, can support $\mathbf{k}$-allowable traffic with full wavelength conversion at all nodes and without wavelength conversion respectively. It is clear that $L_{d,\mathbf{k}} \leq W_{d,\mathbf{k}}$. Notice that $L_{d,\mathbf{k}}$ and $W_{d,\mathbf{k}}$ are the number of wavelengths required to support *any* traffic matrix in the $\mathbf{k}$-allowable set. Thus, for a specific traffic matrix, we may need fewer wavelengths than in the worst-case. To derive $L_{d,\mathbf{k}}$, consider the fiber from the node with traffic parameter k_{max} to the hub node. This fiber must support up to k_{max} wavelengths, which is the maximum link load. It follows that $L_{d,\mathbf{k}} = k_{max}$.

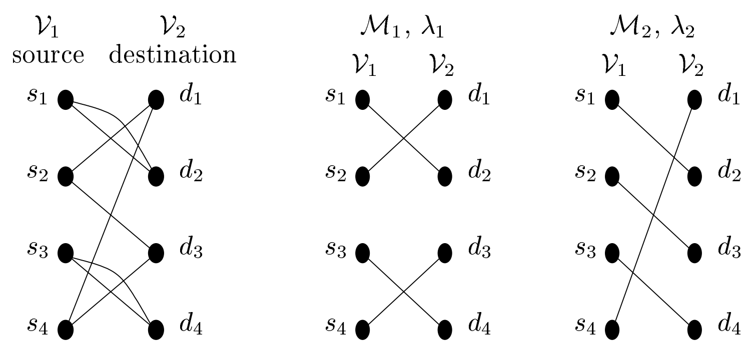
We shall show that $W_{d,\mathbf{k}} \leq k_{max}$, which implies $W_{d,\mathbf{k}} = L_{d,\mathbf{k}} = k_{max}$. We do so by constructing an on-line WA algorithm. Figure 4-1 illustrates an example scenario in which an on-line greedy WA algorithm fails to support an instance of $\mathbf{k}$-allowable traffic using k_{max} wavelengths. In this example, $N = 3$, $\mathbf{k} = [2, 2, 2]$, and the traffic matrix to be supported is uniform all-to-all traffic, i.e. each node sends one wavelength to each of the other two nodes. As shown in figure 4-1, the same wavelength is assigned to the oppositely directed sessions between the same pair of nodes, e.g. sessions (1,2) and (2,1) on wavelength λ_1. After assigning wavelength λ_1 to sessions (1,2) and

(2,1) and wavelength λ_2 to sessions (1,3) and (3,1), neither λ_1 nor λ_2 can be assigned to support session (2,3). It follows that more than $k_{max} = 2$ wavelengths are required. Therefore, this example scenario tells us that the WA algorithm design using k_{max} wavelengths is not trivial. Figure 4-1 also demonstrates that, to use the minimum number of wavelengths, we may need to support the oppositely directed sessions between the same pair of nodes on different wavelengths.

Our algorithm is based on bipartite matchings. For a given traffic matrix, we construct the *traffic bipartite graph*, denoted by $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$, as follows. For convenience, we consider each leaf node as one distinct source node and one distinct destination node. The set of nodes $\mathcal{V}_1$ contains the N source nodes. The set of nodes $\mathcal{V}_2$ contains the N destination nodes. In the set of edges $\mathcal{E}$, an edge between node i in $\mathcal{V}_1$ and node j in $\mathcal{V}_2$ exists for each traffic session from source i to destination j. Figure 4-2a shows an example of the traffic bipartite graph and its traffic matrix. Note that there may be multiple edges between the same pair of nodes. For example, since there are two sessions from source 1 to destination 2, there are two parallel edges between s_1 in $\mathcal{V}_1$ and d_2 in $\mathcal{V}_2$ in figure 4-2a.

destination

$$\begin{array}{cc} & \begin{array}{cccc} d_1 & d_2 & d_3 & d_4 \end{array} \\ \text{source} & \\ \begin{array}{c} s_1 \\ s_2 \\ s_3 \\ s_4 \end{array} & \begin{bmatrix} 0 & 2 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 2 \\ 1 & 0 & 1 & 0 \end{bmatrix} \end{array}$$



(a) traffic matrix and its traffic bipartite graph

(b) bipartite matchings $\mathcal{M}_1$ and $\mathcal{M}_2$ assigned to wavelengths λ_1 and λ_2

Figure 4-2: Traffic bipartite graph and its matchings.

Figure 4-2b shows one partition of the set $\mathcal{E}$ into two disjoint bipartite matchings $\mathcal{M}_1$ and $\mathcal{M}_2$. Observe that the sessions in a bipartite matching can be supported on a single wavelength without wavelength collision. To see this, note that, in a matching, at most one edge is incident on each source (destination) node. Thus, in each fiber to (from) the hub node, every wavelength is used at most once. Our algorithm will assign a single bipartite matching to a single wavelength. In what follows, we shall refer to the matching in the traffic bipartite graph which is assigned to wavelength

λ_1 simply as the bipartite matching of λ_1. Figure 4-2b shows an example of bipartite matchings of specific wavelengths.

Before presenting our on-line WA algorithm, we derive a few useful lemmas related to bipartite matchings. These lemmas are consequences of Hall's theorem and lemma 5 introduced in section 3.1. The first lemma is a more general version of lemma 5.

Lemma 6 *In a bipartite graph* $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$ *with* $|\mathcal{V}_1| = |\mathcal{V}_2| = V$*, if each node has degree at most* m*, the set* $\mathcal{E}$ *can be partitioned into* m *disjoint bipartite matchings.*

Proof: If all nodes have degree m, then lemma 5 can be applied. It remains to consider the cases in which some node has degree less than m.

When some node has degree less than m, we can add extra edges to make each node have degree m. Such extra edges can be added one by one as follows. Label nodes in $\mathcal{V}_1$ and in $\mathcal{V}_2$ from 1 to V. Find the lowest-index node in $\mathcal{V}_1$ with degree less than m. Add an edge from this node to the lowest-index node in $\mathcal{V}_2$ with degree less than m. Repeat the process until all nodes in $\mathcal{V}_1$ have degree m. Since the sum of the degrees of the nodes in $\mathcal{V}_1$ is equal to the sum in $\mathcal{V}_2$, there is always a node in $\mathcal{V}_2$ for each extra edge. When all nodes in $\mathcal{V}_1$ have degree m, there are mV edges incident on nodes in $\mathcal{V}_2$. Since we never add an extra edge to a node in $\mathcal{V}_2$ with degree m, all nodes in $\mathcal{V}_2$ also have degree m in the modified graph.

By lemma 5, the edges in the modified bipartite graph can be partitioned into m perfect matchings. By removing the extra edges from each matching, we obtain our desired m disjoint bipartite matchings. $\square$

Lemma 6 can be used to argue that k_{max} wavelengths are sufficient to support any traffic matrix in the **k**-allowable set. Given a traffic matrix, we can write down the corresponding traffic bipartite graph in which each node has degree at most k_{max}. By lemma 6, the set of edges can be partitioned into k_{max} disjoint bipartite matchings. The sessions in each matching can be supported on a single wavelength. Thus, k_{max} wavelengths are sufficient to support any **k**-allowable traffic matrix.

The main idea of our on-line WA algorithm involves keeping k_{max} disjoint bipartite matchings of k_{max} wavelengths such that each traffic session corresponds to an edge in one bipartite matching. When a session departs, we simply remove its corresponding lightpath from the network. When a new session arrives, we update the WA by finding up to two wavelengths whose bipartite matchings can be reassigned to include the new session. Instead of finding k_{max} disjoint bipartite matchings every time a new session arrives as suggested by lemma 6, our on-line WA algorithm needs to find only two disjoint bipartite matchings.

In particular, suppose (i, j) is the new session, i.e. a new session is from source i to destination j. Note that (s_i, d_j) is the corresponding new edge in the traffic bipartite graph. If there is a bipartite matching of some wavelength, say λ_0, which is incident on neither s_i in $\mathcal{V}_1$ nor d_j in $\mathcal{V}_2$, i.e. λ_0 is used by neither source i nor destination j, then the new session can be added to this bipartite matching so that the resultant set of edges is still a matching. In this case, the new session can be supported on λ_0 without any rearrangement of existing lightpaths. If such a wavelength λ_0 does not exist, then we find two bipartite matchings of two wavelengths, say λ_1 and λ_2, such that the bipartite matching of λ_1 is not incident on s_i, i.e. λ_1 is not used by source i, whereas the bipartite matching of λ_2 is not incident on d_j, i.e. λ_2 is not used by destination j. In this case, we partition the edges in the bipartite matchings of λ_1 and λ_2 as well as the new edge (s_i, d_j) into two disjoint matchings. We then assign one matching to λ_1 and the other to λ_2. For $|\mathcal{V}_1| = |\mathcal{V}_2| = V$, λ_1 and λ_2 each contain at most $V - 1$ existing lightpaths, so the number of lightpath rearrangements is bounded above by $2(V - 1)$. The following lemma makes the above discussion rigorous and states a tighter upper bound on the number of lightpath rearrangements.

Lemma 7 *In a bipartite graph $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$ with $|\mathcal{V}_1| = |\mathcal{V}_2| = V$, given a new edge (s_i, d_j), $s_i \in \mathcal{V}_1$, $d_j \in \mathcal{V}_2$, a matching $\mathcal{M}_1$ of wavelength λ_1 which is not incident on s_i, and a matching $\mathcal{M}_2$ of wavelength λ_2 which is not incident on d_j, there exist two disjoint bipartite matchings which contain all the edges in $\mathcal{M}_1$ and $\mathcal{M}_2$ as well as the new edge (s_i, d_j).*

In addition, these two disjoint bipartite matchings can be assigned to λ_1 and λ_2 so that the number of lightpath rearrangements is at most $V - 1$.

Proof: Consider the bipartite graph $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E}')$ whose set of edges $\mathcal{E}'$ contains all of the edges in $\mathcal{M}_1$ and $\mathcal{M}_2$ as well as the new edge (s_i, d_j). Observe that each node has degree at most 2. From lemma 6 with $m = 2$, there exist two disjoint bipartite matchings, denoted by $\mathcal{M}'_1$ and $\mathcal{M}'_2$, which contain all the edges.

Without loss of generality, assume that (s_i, d_j) belongs to $\mathcal{M}'_1$. Let set $\mathcal{P}$ contain the edges in $\mathcal{M}_1$ assigned to $\mathcal{M}'_2$ and the edges in $\mathcal{M}_2$ assigned to $\mathcal{M}'_1$. Let set $\mathcal{Q}$ contain the edges in $\mathcal{M}_1$

assigned to $\mathcal{M}_1'$ and the edges in $\mathcal{M}_2$ assigned to $\mathcal{M}_2'$. Notice that $\mathcal{P}$ and $\mathcal{Q}$ contain all the edges in $\mathcal{M}_1$ and $\mathcal{M}_2$. Since there are at most $2V-2$ edges in $\mathcal{M}_1$ and $\mathcal{M}_2$, it follows that $|\mathcal{P}|+|\mathcal{Q}| \leq 2V-2$.

If $|\mathcal{P}| \leq V-1$, assigning $\mathcal{M}_1'$ to λ_1 and $\mathcal{M}_2'$ to λ_2 yields the desired result that the number of lightpath rearrangements, which is equal to the sum of the number of edges in $\mathcal{M}_1$ assigned to $\mathcal{M}_2'$ and the number of edges in $\mathcal{M}_2$ assigned to $\mathcal{M}_1'$, is at most $V-1$. Otherwise, it is true that $|\mathcal{Q}| \leq V-1$. In this case, assigning $\mathcal{M}_1'$ to λ_2 and $\mathcal{M}_2'$ to λ_1 yields the desired result. □

A general algorithm for bipartite matching is available in [CLR90]. In particular, the general algorithm in [CLR90] is based on converting a bipartite matching problem into a maximum flow problem. For a bipartite graph $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$, the corresponding running time is proportional to the product VE, where $V = |\mathcal{V}_1| = |\mathcal{V}_2|$ and $E = |\mathcal{E}|$. For our purpose of partitioning the edges in a bipartite graph with maximum node degree 2 into two disjoint matchings, the running time is $O(V^2)$ for the general algorithm.[1] In appendix A, we provide an efficient specialized procedure to find such two disjoint bipartite matchings with the running time $O(V)$.

The following is our on-line WA algorithm for a star topology with **k**-allowable traffic which uses k_{max} wavelengths in each fiber, is rearrangeably nonblocking, and requires at most $N-1$ lightpath rearrangements per new session request. We shall refer to this algorithm as the *on-line star WA algorithm.*

Algorithm 2 (On-Line Star WA Algorithm) (Use k_{max} wavelengths in each fiber.)

Session termination: When a session terminates, simply remove its associated lightpath from the network without any further lightpath rearrangement.

Session arrival: When a new session arrives and the resultant traffic matrix is still **k**-allowable, proceed as follows. Assume that the new session is from source i to destination j.

Step 1: If there is a wavelength, denoted by λ_0, which is used by neither source i nor destination j, i.e. its matching in the traffic bipartite graph $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$ is incident on neither s_i nor d_j, then assign the new session to λ_0. In this case, no lightpath rearrangement is required. Otherwise, proceed to step 2.

[1]By running time $O(g(n))$, we mean the running time can be expressed as a function $f(n)$ of the problem size n such that there exist a positive real constant c and a positive integer n_0 satisfying $0 \leq f(n) \leq cg(n)$ for all $n \geq n_0$.

Step 2: Find a wavelength, denoted by λ_1, which is not used by source i, i.e. its bipartite matching is not incident on s_i, and another wavelength, denoted by λ_2, which is not used by destination j, i.e. its bipartite matching is not incident on d_j. Since the new session is allowable, there are at most $k_{max} - 1$ sessions from source i. Since there are k_{max} available wavelengths, it follows that λ_1 exists. By the same argument, λ_2 always exists.

Modify the WA of only the sessions on λ_1 and λ_2. Construct the traffic bipartite graph $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E}')$ in which the set of edges $\mathcal{E}'$ contains the bipartite matchings of λ_1 and λ_2 as well as the new edge (s_i, d_j). From lemma 7, we can partition the set $\mathcal{E}'$ into two disjoint bipartite matchings. In addition, since $|\mathcal{V}_1| = |\mathcal{V}_2| = N$, lemma 7 tells us that the two matchings can be assigned to λ_1 and λ_2 such that at most $N - 1$ existing lightpaths need to be rearranged.

The construction of the on-line star WA algorithm implies the following theorem.

Theorem 5 *For the star topology with N nodes and $\mathbf{k}$-allowable traffic, $W_{d,\mathbf{k}}$ is given by*

$$W_{d,\mathbf{k}} \;=\; L_{d,\mathbf{k}} \;=\; k_{max} \;=\; \max_{1 \le i \le N} k_i.$$

In addition, there exists, by construction, an on-line WA algorithm which uses k_{max} wavelengths in each fiber and requires at most $N - 1$ lightpath rearrangements per new session request.

The following example illustrates the operations of the on-line star WA algorithm.

Example 5 Consider a 4-node star network with the traffic matrix given in figure 4-2a. Note that $W_{d,\mathbf{k}} = 2$. Assume that the WA is given by the two bipartite matchings of wavelengths λ_1 and λ_2 as shown in figure 4-2b. Now assume the following changes in the traffic matrix.

1. Existing session (3,4) on λ_1 terminates.
2. Existing session (4,1) on λ_2 terminates.
3. A new session (3,1) arrives.

After the second session termination, the bipartite matchings of λ_1 and λ_2 are shown in figure 4-3a. To support the new session, the star WA algorithm performs step 2. In particular, it creates a traffic bipartite graph whose edges are the bipartite matchings of λ_1 and λ_2 as well as the new edge (s_3, d_1). The algorithm then partitions the set of edges into two disjoint bipartite matchings and assigns them to λ_1 and λ_2, as shown in figure 4-3b. In particular, session (3,4) on λ_2 is reassigned to λ_1, and the new session is then assigned to λ_2. In this example, one rearrangement of an existing lightpath is made to support the new session.

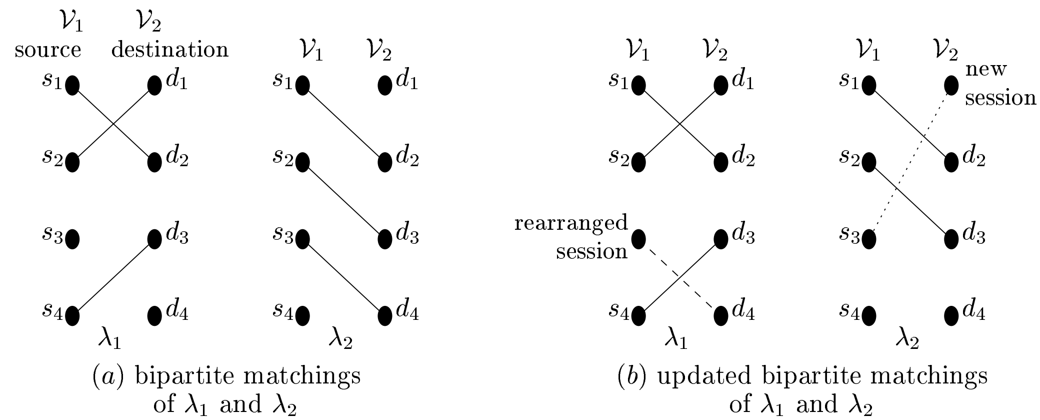


Figure 4-3: Example operations of the on-line star WA algorithm.

The next example demonstrates that the on-line star WA algorithm may perform up to $N-1$ lightpath rearrangements to support a new session request. Consider the following WA scenario. Assume that each wavelength supports one of the two bipartite matchings shown in figure 4-4a.

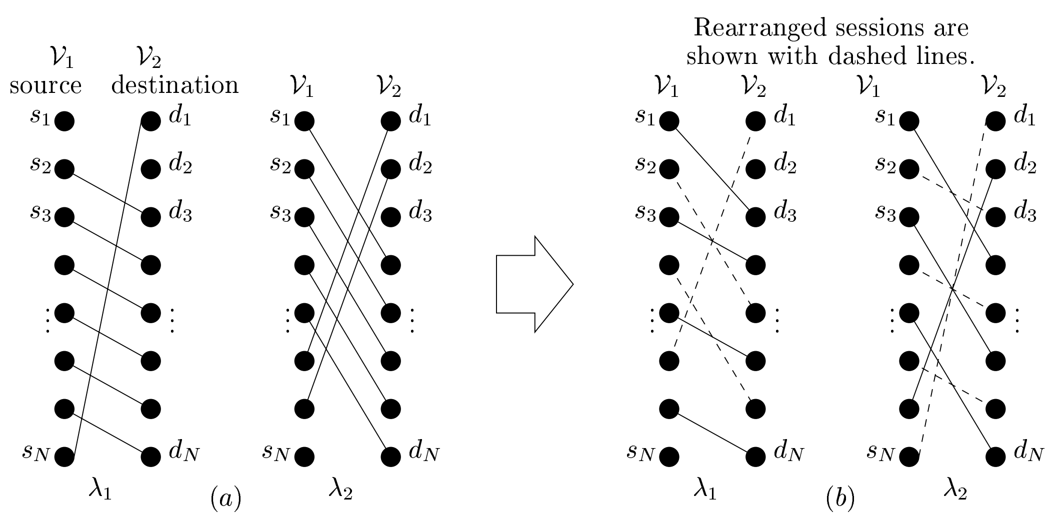


Figure 4-4: An example case in which $N-1$ lightpath rearrangements are made to support a new session.

Suppose the new session is transmitted from source 1 to destination 3. In this case, the on-line star WA algorithm needs to perform step 2. After choosing two bipartite matchings of wavelengths

λ_1 and λ_2, as shown in figure 4-4a, the algorithm creates a traffic bipartite graph whose edges are all the edges in the bipartite matchings of λ_1 and λ_2 as well as the new edge (s_1, d_3). The algorithm then partitions the set of edges into two disjoint bipartite matchings and assign them to λ_1 and λ_2, as shown in figure 4-4b. In this example, the algorithm needs to perform $N-1$ lightpath rearrangements to support the new session.

In the next section, we shall extend the on-line star WA algorithm to create an on-line WA algorithm for an arbitrary tree topology.

4.2 Arbitrary Tree Topologies

In this section, we solve the RWA problem for $\mathbf{k}$-allowable traffic in an arbitrary tree topology. Since there is a unique route for each traffic session, there is no routing problem in a tree topology. Thus, we only have to perform wavelength assignment (WA) in the RWA problem. We shall extend the on-line star WA algorithm to create an on-line WA algorithm for an arbitrary tree topology. In a given tree topology, assume there are $N > 2$ end nodes which are the leaf nodes of the tree.[2] We shall ignore all the non-leaf nodes with degree 2 since their removal does not change the WA problem. We describe a tree by a set of nodes $\mathcal{N}$ and a set of bidirectional links $\mathcal{T}$.

Let $L_{d,\mathbf{k}}$ denote the minimum number of wavelengths which, if provided in each fiber, can support $\mathbf{k}$-allowable traffic given full wavelength conversion at all nodes. We first determine $L_{d,\mathbf{k}}$. Each link e in the tree corresponds to a cut which separates N leaf nodes into two sets, denoted by $\mathcal{N}_{e,1}$ and $\mathcal{N}_{e,2}$. The maximum possible traffic, in wavelength units, in a fiber across this link is equal to $\min(\sum_{i\in\mathcal{N}_{e,1}} k_i, \sum_{i\in\mathcal{N}_{e,2}} k_i)$. The maximum over all links of the traffic on a fiber is denoted by w^*. This is the value of $L_{d,\mathbf{k}}$, as given below.

$$L_{d,\mathbf{k}} \quad = \quad w^* \quad = \quad \max_{e\in\mathcal{T}} \min\left(\sum_{i\in\mathcal{N}_{e,1}} k_i, \sum_{i\in\mathcal{N}_{e,2}} k_i\right) \tag{4.1}$$

Let $W_{d,\mathbf{k}}$ denote the minimum number of wavelengths which, if provided in each fiber, can support $\mathbf{k}$-allowable traffic with no wavelength conversion. We shall show that $W_{d,\mathbf{k}} \leq w^*$, which implies $L_{d,\mathbf{k}} = W_{d,\mathbf{k}} = w^*$. We do so by constructing an on-line WA algorithm. We shall refer

[2] The WA problem for a tree with two leaf nodes is trivial.

to w^* as the *worst-case number of wavelengths* since w^* wavelengths are necessary and sufficient to support *any* traffic matrix in the $\mathbf{k}$-allowable traffic set. Since a star topology is also a tree topology, figure 4-1 illustrates that the WA algorithm design using w^* wavelengths is not trivial for an arbitrary tree topology.

We now derive a few useful properties related to the worst-case number of wavelengths w^*. Let e^* denote the link associated with w^*. Note that there may be multiple choices for e^*. When there are multiple choices for e^*, the exact choice does not matter in the following discussion. We shall refer to e^* as the *bottleneck link* since it is the link with the maximum load under the worst-case traffic.

Link e^* separates the leaf nodes into two sets $\mathcal{N}_{e^*,1}$ and $\mathcal{N}_{e^*,2}$. Without loss of generality, choose $\mathcal{N}_{e^*,1}$ such that the sum of k_i's in this set is w^*. We assume for now that $\mathcal{N}_{e^*,2}$ contains multiple leaf nodes, as illustrated in figure 4-5. Define the *bottleneck node* v^* to be the end point of e^* opposite to $\mathcal{N}_{e^*,1}$, i.e. the subtree connected to v^* by e^* has the sum of k_i's equal to w^*, as illustrated in figure 4-5.

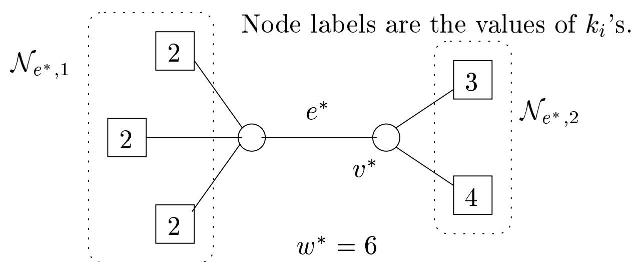


Figure 4-5: Definition of the bottleneck node v^*.

We shall refer to each subtree connected to v^* as a *top-level subtree*. Note that a top-level subtree can be a single node. Let d^* be the degree of v^*.[3] Since v^* is a non-leaf node, $d^* \geq 3$. It follows that there are $d^* \geq 3$ top-level subtrees, as illustrated in figure 4-6a.

If the set $\mathcal{N}_{e^*,2}$ contains a single node, we have the scenario illustrated in figure 4-6b. In this case, assumption 1 implies that the value of k_i for the leaf node in $\mathcal{N}_{e^*,2}$ is equal to w^*. We argue that, with $N > 2$ leaf nodes, this scenario can be transformed to the scenario in figure 4-6a by exchanging the roles of $\mathcal{N}_{e^*,1}$ and $\mathcal{N}_{e^*,2}$. After the exchange, the set $\mathcal{N}_{e^*,2}$ will contain multiple

[3]Since we assume that each link consists of two fibers, one in each direction, the indegree and the outdegree of any given network node are the same. We simply refer to their value as the node degree.

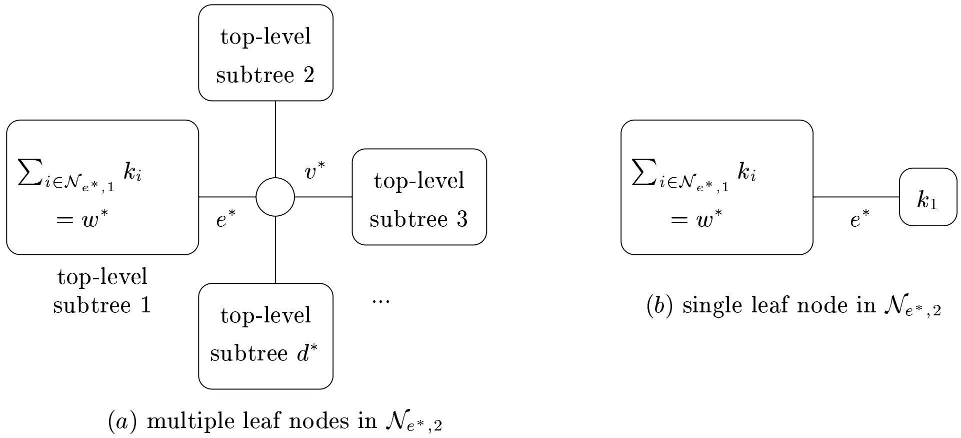


Figure 4-6: Illustrations of the bottleneck node v^* and the top-level subtrees.

nodes, and we have a scenario as illustrated in figure 4-6a. Therefore, we shall consider only the scenarios in which v^* exists and $d^* \geq 3$, as illustrated in figure 4-6a.

Note that the location of the bottleneck node v^* depends on the specific tree topology and the traffic vector $\mathbf{k}$, but not on the current traffic matrix being supported. The following lemma provides useful properties of the top-level subtrees connected to v^* as well as bounds on the worst-case number of wavelengths w^*.

Lemma 8 *Under assumption 1, the following properties hold.*

1. *Let $K_j, 1 \leq j \leq d^*$, denote the sum of k_i's in top-level subtree j. For all $1 \leq j \leq d^*$, $K_j \leq w^*$.*
2. *Let $K = \sum_{1\leq i\leq N} k_i$. The worst-case number of wavelengths w^* is bounded by*

$$K/d^* \leq w^* \leq K/2.$$

Proof:

1. Number the d^* top-level subtrees from 1 to d^* such that top-level subtree 1 is connected to v^* by e^*. By the definition of v^*, we know that $K_1 = w^*$. For $2 \leq j \leq d^*$, consider the link e_j which isolates top-level subtree j from v^*. Let $\mathcal{N}_{e_j,1}$ contain the leaf nodes in top-level subtree j, and $\mathcal{N}_{e_j,2}$ contain all the other leaf nodes. Consequently, $\sum_{i\in\mathcal{N}_{e_j,1}} k_i = K_j$. In addition, $\sum_{i\in\mathcal{N}_{e_j,2}} k_i = K - K_j > K_1 = w^*$ since there are at least three top-level subtrees. From the definition of w^* in (4.1), we must have that $K_j \leq w^*$, or else e_j instead of e^* would be the bottleneck link. It follows that $K_j \leq w^*$ for $2 \leq j \leq d^*$. Thus, $K_j \leq w^*$ for all $1 \leq j \leq d^*$.

2. From the definition of w^*, it is clear that $w^* \leq K/2$. To prove the lower bound, we use statement 1 of the lemma, i.e. $K_j \leq w^*$ for all $1 \leq j \leq d^*$, to show that

$$K \;=\; \sum_{1 \leq j \leq d^*} K_j \;\leq\; d^* w^*.$$

The above inequality yields the desired lower bound $w^* \geq K/d^*$. □

As in the on-line star WA algorithm, the algorithm in this section is based on bipartite matchings. The main difference has to do with what a node in a bipartite graph represents. In the on-line star WA algorithm, a node represents a single source or a single destination. In this section, a node represents a set of sources or a set of destinations in a top-level subtree.

For a given traffic matrix, we construct the *top-level subtree bipartite graph*, denoted by $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$, as follows. We consider each leaf node as one distinct source and one distinct destination. Number the d^* top-level subtrees from 1 to d^*. The set $\mathcal{V}_1$ contains d^* abstract nodes, denoted by $\mathcal{S}_1$, $\mathcal{S}_2$, ..., $\mathcal{S}_{d^*}$. Node $\mathcal{S}_i$, $1 \leq i \leq d^*$, represents the set of sources contained in top-level subtree i. Similarly, the set $\mathcal{V}_2$ contains d^* abstract nodes, denoted by $\mathcal{D}_1$, $\mathcal{D}_2$, ..., $\mathcal{D}_{d^*}$. Node $\mathcal{D}_j$, $1 \leq j \leq d^*$, represents the set of destinations contained in top-level subtree j. In the set of edges $\mathcal{E}$, an edge from node $\mathcal{S}_i$ in $\mathcal{V}_1$ to node $\mathcal{S}_j$ in $\mathcal{V}_2$ exists for each traffic session from a source in top-level subtree i to a destination in top-level subtree j. Figure 4-7 shows an example of the top-level subtree bipartite graph and its traffic matrix. Note that there may be multiple edges between the same pair of nodes. For example, since there are two sessions from top-level subtree 3 to top-level subtree 4, there are two parallel edges between the set of sources $\mathcal{S}_3$ and the set of destinations $\mathcal{D}_4$ in figure 4-7d.

Define a *local session* to be a traffic session whose source and destination are in the same top-level subtree. Accordingly, a *non-local session* has its source and its destination in different top-level subtrees. A non-local session has to travel through the bottleneck node v^*, whereas a local session does not have to travel all the way to v^* and back to its destination, i.e. each session does not use the same link twice in the opposite directions. A non-local session corresponds to an edge from some node $\mathcal{S}_i$ in $\mathcal{V}_1$ and some node $\mathcal{D}_j$ in $\mathcal{V}_2$, where $i \neq j$. On the other hand, a local session corresponds to an edge between some node $\mathcal{S}_i$ in $\mathcal{V}_1$ and node $\mathcal{D}_i$ in $\mathcal{V}_2$. For example, the top-level subtree bipartite graph in figure 4-7d contains seven non-local sessions and one local session. The local session is from a source in top-level subtree 2 to a destination in the same top-level subtree.

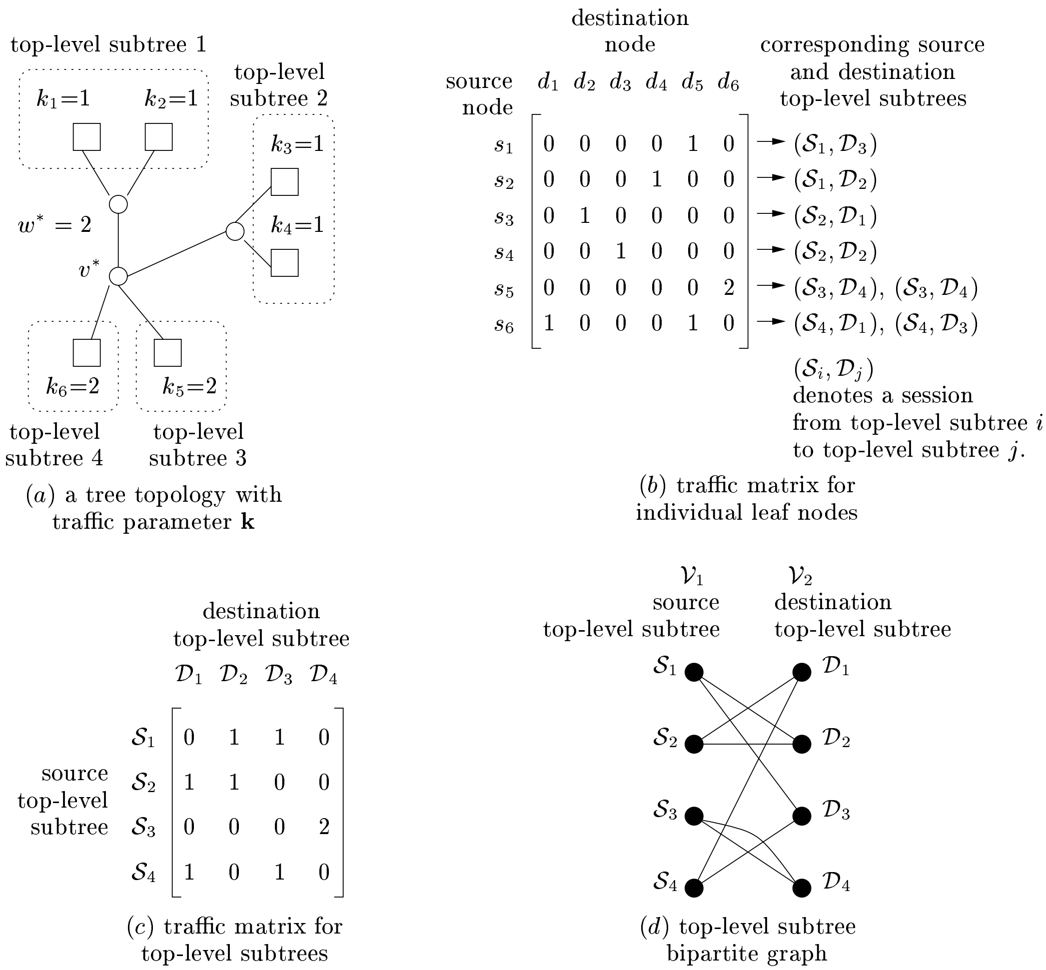


Figure 4-7: Top-level subtree bipartite graph.

Observe that the sessions belonging to a matching in the top-level subtree bipartite graph can be supported on a single wavelength without wavelength collision. To see this, note that any two sessions in a bipartite matching are transmitted from different top-level subtrees and to different top-level subtrees. Consequently, if these two sessions travel in the same top-level subtree, one session must be transmitted from that subtree while the other session must be received in that subtree. It follows that these two sessions always traverse links belonging to the same top-level subtree in the opposite directions and do not collide.

Our algorithm will assign a single bipartite matching to a single wavelength. We shall refer to the matching assigned to wavelength λ_1 as the bipartite matching of λ_1. Figure 4-8 shows example bipartite matchings of specific wavelengths.

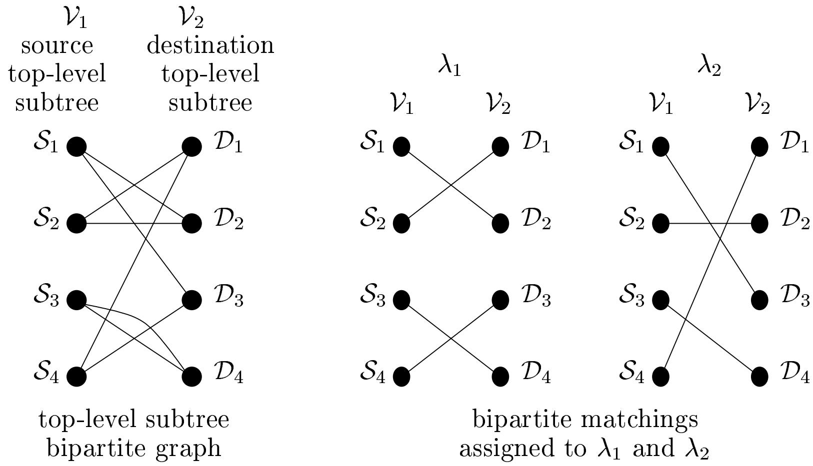


Figure 4-8: Bipartite matchings of specific wavelengths.

We now argue that w^* wavelengths are sufficient to support any traffic matrix in the $\mathbf{k}$-allowable set. From statement 1 of lemma 8, each top-level subtree can transmit at most w^* wavelengths and receive at most w^* wavelengths. Thus, for a given a traffic matrix, each node in the corresponding top-level subtree bipartite graph has degree at most w^*. By lemma 6, the set of edges can be partitioned into w^* disjoint bipartite matchings. The sessions in each matching can be supported on a single wavelength. Thus, w^* wavelengths are sufficient to support any $\mathbf{k}$-allowable traffic matrix. Notice that, by finding w^* disjoint bipartite matchings, we provide the WA for both local and non-local sessions simultaneously.

The main idea of our on-line WA algorithm involves keeping w^* disjoint bipartite matchings of w^* wavelengths such that each traffic session corresponds to an edge in one bipartite matching. When a session departs, we simply remove its corresponding lightpath from the network. When a new (local or non-local) session arrives, we update the WA by finding up to two wavelengths whose bipartite matchings can be reassigned to include the new session.

The following is our on-line WA algorithm for an arbitrary tree topology with $\mathbf{k}$-allowable traffic which uses w^* wavelengths in each fiber, is rearrangeably nonblocking, and requires at most $d^* - 1$ lightpath rearrangements per new session request. We shall refer to this algorithm as the *on-line tree WA algorithm.*

Algorithm 3 (On-Line Tree WA Algorithm) (Use w^* wavelengths in each fiber.)

Session termination: When a session terminates, simply remove its associated lightpath from the network without any further lightpath rearrangement.

Session arrival: When a new session arrives and the resultant traffic matrix is still $\mathbf{k}$-allowable, proceed as follows. Assume that the new session is from a source in top-level subtree i to a destination in top-level subtree j. When $i = j$, the new session is local. Otherwise, it is non-local. In either case, follow the same procedures below.

Step 1: If there is a wavelength, denoted by λ_0, which is used by neither a source in top-level subtree i nor a destination in top-level subtree j, i.e. its matching in the traffic bipartite graph $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$ is incident on neither $\mathcal{S}_i$ nor $\mathcal{D}_j$, then assign the new session to λ_0. In this case, no lightpath rearrangement is required. Otherwise, proceed to step 2.

Step 2: Find a wavelength, denoted by λ_1, which is not used by any source in top-level subtree i, i.e. its bipartite matching is not incident on $\mathcal{S}_i$, and another wavelength, denoted by λ_2, which is not used by any destination in top-level subtree j, i.e. its bipartite matching is not incident on $\mathcal{D}_j$. Since the new session is allowable, there are at most $w^* - 1$ sessions from top-level subtree i. Since there are w^* available wavelengths, it follows that λ_1 exists. By the same argument, λ_2 always exists.

Modify the WA of only the sessions on λ_1 and λ_2. Construct the top-level subtree bipartite graph $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E}')$ in which the set of edges $\mathcal{E}'$ contains the bipartite matchings of λ_i and λ_j as well as the new edge $(\mathcal{S}_i, \mathcal{D}_j)$. From lemma 7, we can partition the set $\mathcal{E}'$ into two disjoint bipartite matchings. In addition, since $|\mathcal{V}_1| = |\mathcal{V}_2| = d^*$, lemma 7 tell us that the two matchings can be assigned to λ_1 and λ_2 such that at most $d^* - 1$ existing lightpaths need to be rearranged.

The construction of the on-line tree WA algorithm implies the following theorem.

Theorem 6 *For an arbitrary tree topology with* $\mathbf{k}$*-allowable traffic among* N *leaf nodes and the bottleneck node* v^* *with degree* d^**,* $W_{d,\mathbf{k}}$ *is given by*

$$W_{d,\mathbf{k}} = L_{d,\mathbf{k}} = w^* = \max_{e \in \mathcal{T}} \min \left(\sum_{i \in \mathcal{N}_{e,1}} k_i, \sum_{i \in \mathcal{N}_{e,2}} k_i \right).$$

In addition, there exists, by construction, an on-line WA algorithm which uses w^* wavelengths in each fiber and requires at most $d^* - 1$ lightpath rearrangements per new session request.

Theorem 6 tells us that wavelength conversion cannot decrease the wavelength requirement for $\mathbf{k}$-allowable traffic in an arbitrary tree topology. In addition, if we scale the traffic vector $\mathbf{k}$ by an integer factor, then the location of the bottleneck node v^* remains fixed, and the upper bound on the number of lightpath rearrangements per new session request does not increase. Finally, from statement 2 of lemma 8, among the tree topologies with N leaf nodes, the minimum value of the worst-case number of wavelengths w^* is at least $(\sum_{1 \leq i \leq N} k_i)/d^*$. The tree topologies with w^* close to this lower bound are the ones in which each top-level subtree has the sum of k_i's approximately equal to $(\sum_{1 \leq i \leq N} k_i)/d^*$. Roughly speaking, it is desirable to have all the top-level subtrees support an equal amount of traffic.

The following example illustrates the operations of the on-line tree WA algorithm.

Example 6 Consider the tree network with the traffic matrix given in figure 4-7. Note that $W_{d,\mathbf{k}} = 2$. Assume that the corresponding WA is given by the two bipartite matchings of wavelengths λ_1 and λ_2 as shown in figure 4-8. Now assume the following changes in the traffic matrix.

1. The existing session from source 3 in top-level subtree 2 to destination 2 in top-level subtree 1 on λ_1 terminates.
2. The existing session from source 1 in top-level subtree 1 to destination 5 in top-level subtree 3 on λ_2 terminates.
3. A new session from source 1 to destination 2 in top-level subtree 1 arrives.

After the second session termination, the bipartite matchings of λ_1 and λ_2 are shown in figure 4-9a. To support the new session, the tree WA algorithm performs step 2. In particular, it creates a top-level subtree bipartite graph whose edges are the bipartite matchings of λ_1 and λ_2 as well as the new edge $(\mathcal{S}_1, \mathcal{D}_1)$. The algorithm then partitions the set of edges into two disjoint bipartite matchings and assign them to λ_1 and λ_2, as shown in figure 4-9b. In particular, the session from top-level subtree 4 to top-level subtree 3 on λ_1 is reassigned to λ_2. In addition, the session from top-level subtree 4 to top-level subtree 1 on λ_2 is reassigned to λ_1. The new session is then assigned to λ_2. In this example, two rearrangements of existing lightpaths are made to support the new session.

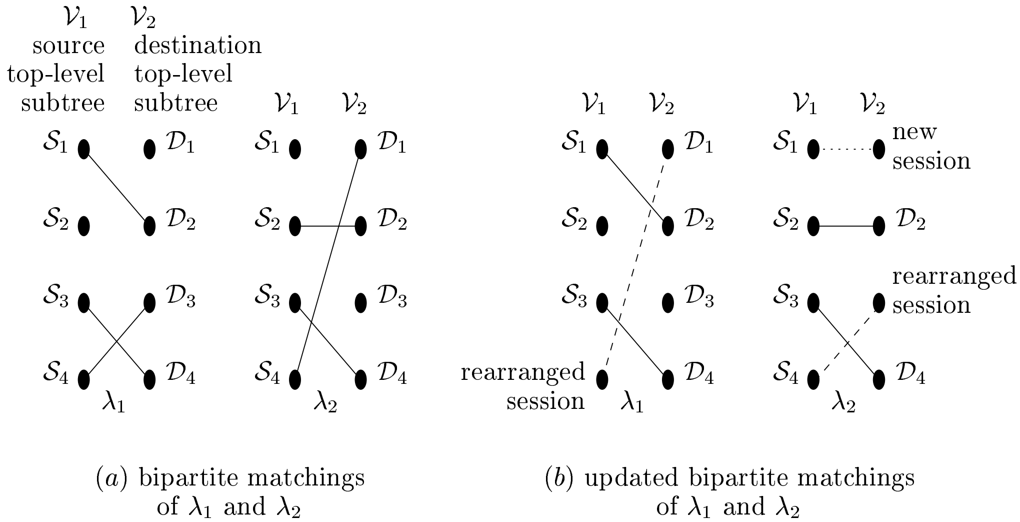


Figure 4-9: Example operations of the on-line tree WA algorithm.

Finally, an example similar to the one based on the WA given in figure 4-4 for a star topology can be constructed to show that the on-line tree WA algorithm may perform up to $d^* - 1$ lightpath rearrangements to support a new session request. We shall not repeat the details here.

4.3 Bidirectional Ring Topologies

In this section, we study the RWA problem for **k**-allowable traffic for an N-node bidirectional ring topology, where $N > 2$.[4] We first consider symmetric k-allowable traffic, i.e. **k**-allowable traffic in which all the k_i's are equal to k. Let $W_{d,k}$ denote the minimum number of wavelengths which, if provided in each fiber, can support symmetric k-allowable traffic. In [NLM02], it was shown that, $W_{d,k} = \lceil Nk/3 \rceil$ for $N \geq 7$. In addition, an off-line RWA algorithm that uses at most $\lceil Nk/3 \rceil$ wavelengths in each fiber (or equivalently in each ring direction) was developed.

In appendix B, we derive $W_{d,k}$ for the other values of N, i.e. $N < 7$, to obtain the closed-form expression

$$W_{d,k} = \begin{cases} \lceil 3k/4 \rceil, & N = 3, \\ k, & N = 4, \\ \lceil 5k/3 \rceil, & N = 5, 6, \\ \lceil Nk/3 \rceil, & N \geq 7. \end{cases} \tag{4.2}$$

[4]The RWA problem for a ring with two end nodes is trivial.

We first show that $W_{d,k} \geq \lceil Nk/3 \rceil$ for $N \geq 7$. Consider the symmetric k-allowable traffic in which each node sends k wavelengths to the node $(N-1)/2$ hops away in the clockwise (CW) ring direction for N odd, and $N/2-1$ hops away for N even. Figure 4-10 illustrates this traffic for $k=1$ in the 7-node ring and the 8-node ring.

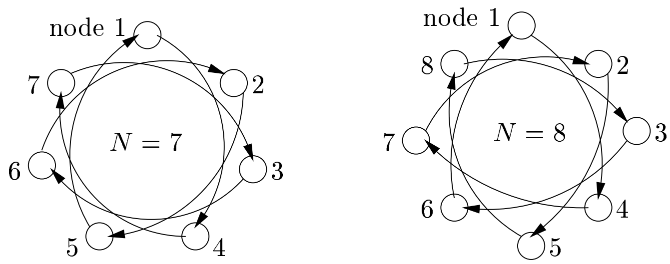


Figure 4-10: Symmetric k-allowable traffic for the lower bound of $W_{d,k}$ for $N \geq 7$.

Define a *directed wavelength* as a wavelength in either the clockwise (CW) or the counter-clockwise (CCW) ring direction. Given w wavelengths in each fiber, there are w CW directed wavelengths, and w CCW directed wavelengths. Note that any traffic session can be supported on a directed wavelength in either ring direction.

We next show that, for the traffic described above, a CW directed wavelength can support at most two sessions, while a CCW directed wavelength can support at most one session. Consider N odd. Any set of three sessions has the sum of path lengths in the CW direction equal to $3(N-1)/2$, which is greater than N, the number of links in each directed wavelength, for $N \geq 5$. (More explicitly, $3(N-1)/2 - N = (N-3)/2 > 0$ for $N \geq 5$.) In addition, any set of two sessions has the sum of path lengths in the CCW direction equal to $2(N+1)/2$, which is greater than N. (More explicitly, $2(N+1)/2 - N = 1 > 0$.)

Consider N even. Any set of three sessions has the sum of path lengths in the CW direction equal to $3(N/2-1)$, which is greater than N for $N \geq 8$. (More explicitly, $3(N/2-1) - N = N/2 - 3 > 0$ for $N \geq 8$.) In addition, any set of two sessions has the sum of path lengths in the CCW direction equal to $2(N/2+1)$, which is greater than N. (More explicitly, $2(N/2+1) - N = 2 > 0$.)

We conclude that, for $N \geq 7$, a CW directed wavelength can support at most two sessions, while a CCW directed wavelength can support at most one session. Thus, each wavelength can support at most three sessions. Since there are in total Nk sessions in the traffic described above, it follows that $W_{d,k} \geq \lceil Nk/3 \rceil$.

We shall present an on-line RWA algorithm that uses $\lceil (\sum_{i=1}^{N} k_i)/3 \rceil$ wavelengths in each fiber to support $\mathbf{k}$-allowable traffic. Note that, for $N \geq 7$, our algorithm can be used to support symmetric k-allowable traffic using the minimum number of wavelengths, i.e. $\lceil Nk/3 \rceil$ wavelengths in a fiber. In all the other cases, the algorithm yields an upper bound on $W_{d,\mathbf{k}}$, the minimum number of wavelengths which, if provided in each fiber, can support $\mathbf{k}$-allowable traffic with no wavelength conversion, i.e. $W_{d,\mathbf{k}} \leq \lceil (\sum_{i=1}^{N} k_i)/3 \rceil$.

We now describe the main idea behind our algorithm. Two sessions are said to be *adjacent* if the destination node of one session is the source node of the other session. The main idea behind our algorithm involves sharing a directed wavelength between two adjacent sessions, as suggested by the following known lemma in [NLM02].

Lemma 9 [NLM02] *In a bidirectional ring, any pair of adjacent sessions can either be supported on one CW directed wavelength or one CCW directed wavelength.*

The proof of lemma 9 is immediate from figure 4-11, where if the two corresponding lightpaths overlap in one ring direction, they do not overlap in the other direction. For example, lightpaths corresponding to a pair of adjacent sessions (1,4) and (4,2) collide in the CW direction, but do not collide in the CCW direction. In what follows, when an adjacent session pair is supported on one directed wavelength, we say that they share a directed wavelength.

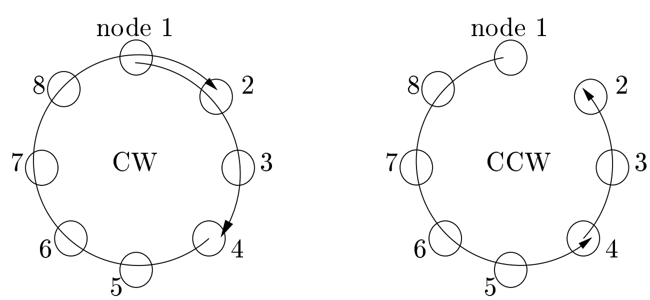


Figure 4-11: Adjacent sessions share a directed wavelength.

The main idea of our algorithm is to maintain the following two RWA conditions at all times: (i) only adjacent sessions share a directed wavelength, and (ii) at most two adjacent sessions share a directed wavelength.

To give some intuition about the main idea of our algorithm, consider the special case with all the k_i's equal to 1, i.e. symmetric 1-allowable traffic. In this case, our algorithm uses $\lceil N/3 \rceil$ wavelengths. We next describe informally how to use $\lceil N/3 \rceil$ wavelengths to support the traffic. We ignore integer rounding in the informal discussion below.

Given a traffic matrix, form as many adjacent session pairs as possible, up to $N/3$ pairs, in a greedy fashion, i.e. it does not matter if we end up with less than the maximum possible number of pairs. Let p denote the number of adjacent session pairs formed. Consider two cases.

- Case 1: $p = N/3$. In this case, we support $N/3$ adjacent session pairs containing $2N/3$ sessions on $N/3$ directed wavelengths in the required ring directions. This is always possible since there are $N/3$ directed wavelengths available in each ring direction. Having done so, there are at most $N - 2N/3 = N/3$ remaining sessions each of which we support on one directed wavelength in any ring direction. Thus, the total number of directed wavelengths required is at most $N/3 + N/3 = 2N/3$. It follows that $N/3$ wavelengths are sufficient.

- Case 2: $p < N/3$. In this case, we support p adjacent session pairs containing $2p$ sessions on p directed wavelengths in the required ring directions. This is always possible since there are $N/3$ directed wavelengths available in each ring direction. Note that we cannot form any new adjacent session pair in this case.

  Observe that each adjacent session pair has at least one common node. Figure 4-12 shows two adjacent session pairs, i.e. (7,4) and (4,3) together with (1,8) and (8,7), whose common nodes are nodes 4 and 8 respectively. In general, given p adjacent session pairs, there are at least p common nodes.

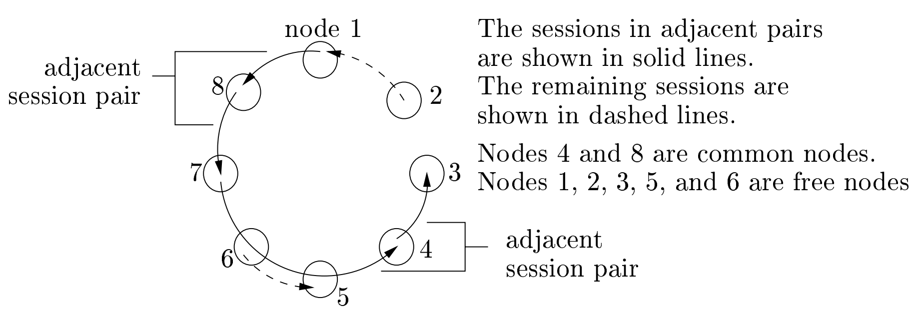


Figure 4-12: Adjacent session pairs, common nodes, and free nodes.

For convenience, we shall refer to all nodes other than the common nodes which can still transmit and/or receive a wavelength as *free nodes*. For example, in figure 4-12, after forming the above two adjacent session pairs, nodes 1, 2, 3, 5, and 6 are free nodes. Since there are at least p common nodes, there are at most $N - p$ free nodes.

Observe that each free node terminates, i.e. either transmits or receives, at most one remaining session. To see this, note that each free node cannot transmit (receive) more than one remaining session since it only has one transmitter (receiver). Moreover, each free node cannot transmit a remaining session and receive a remaining session simultaneously, or else we could form another new adjacent session pair, i.e. have more than p pairs. Thus, each remaining session is terminated at two distinct free nodes. For example, in figure 4-12, the remaining session (2,1) is terminated at free nodes 1 and 2. No other remaining session is terminated at either node 1 or node 2. Since there are at most $N - p$ free nodes, there are at most $(N - p)/2$ remaining sessions. We support each remaining session on one directed wavelength in any ring direction. Thus, the total number of directed wavelengths required is $p + (N - p)/2 = N/2 + p/2 < N/2 + N/6 = 2N/3$. It follows that $N/3$ wavelengths are sufficient.

We shall later prove by similar arguments that $\lceil(\sum_{i=1}^{N} k_i)/3\rceil$ wavelengths are sufficient to support **k**-allowable traffic. We now describe our on-line RWA algorithm which is rearrangeably nonblocking, uses $\lceil(\sum_{i=1}^{N} k_i)/3\rceil$ wavelengths in each fiber, and requires at most three lightpath rearrangements per new session request. We shall refer to this algorithm as the *on-line ring RWA algorithm.*

Algorithm 4 (On-Line Ring RWA Algorithm) (Use $\lceil(\sum_{i=1}^{N} k_i)/3\rceil$ wavelengths in each fiber.)

Session termination: When a session terminates, simply remove its associated lightpath from the ring without any further lightpath rearrangement.

Session arrival: When a session arrives and the resultant traffic matrix is still **k**-allowable, proceed as follows.

Step 1: If there is a nonsharing session, i.e. a session which does not share its directed wavelength with any session, and it is adjacent to and can share its directed wavelength with the new session, assign the two sessions to share that directed wavelength. In this case, no lightpath rearrangement is required. Otherwise, proceed to step 2.

Step 2: If there is a free directed wavelength in either ring direction, assign a free directed wavelength to the new session. In this case, no lightpath rearrangement is required. Otherwise, proceed to step 3.

Step 3: Among the nonsharing sessions and the new session, we claim and shall prove shortly that there must exist a pair of adjacent sessions. Form such an adjacent session pair by searching through all pairs of sessions in some order, e.g. from sessions terminating at node 1 to sessions terminating at node N. Once an adjacent session pair is found, there are two possibilities.

(3a) If the adjacent session pair can share the directed wavelength of one session in the pair, assign the adjacent session pair to share that directed wavelength. In this case, the adjacent session pair does not include the new session since step 1 would have otherwise applied. Therefore, one existing lightpath must be rearranged. Sharing of the directed wavelength by the adjacent session pair will free one directed wavelength on which the new session can be supported with only one lightpath rearrangement. Figure 4-13 illustrates this scenario. In particular, existing sessions (1,5) and (5,2) form an adjacent session pair which can be supported on the directed wavelength of session (5,2). After the lightpath of session (1,5) is rearranged, the new session (1,4) is supported on the directed wavelength previously used by session (1,5).

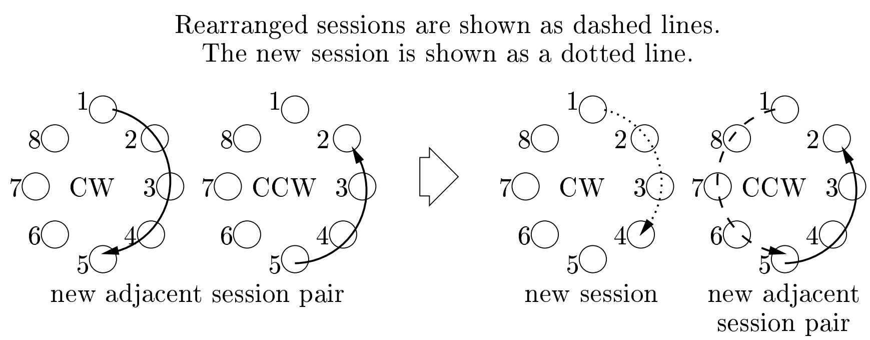


Figure 4-13: Step 3a of the on-line ring RWA algorithm.

(3b) If the adjacent session pair cannot share the directed wavelength of either session in the pair, we claim and shall prove shortly that there must exist a directed wavelength with a nonsharing session in the opposite ring direction, i.e. the ring direction in which the adjacent session pair can share a directed wavelength. Remove the lightpath of that nonsharing session from its directed wavelength, and assign the adjacent session pair to share that directed wavelength. When the adjacent session pair includes the new session, the new session will now be supported, and sharing of the directed wavelength by the adjacent session pair will free one directed wavelength on which the removed nonsharing session can be supported. In this case, a total of two lightpath rearrangements are made. Figure 4-14 illustrates this scenario. In particular, existing session (1,5) and the new session (5,2) form an adjacent session pair which can be supported on the directed wavelength of existing session (3,8). After the lightpaths of sessions (1,5) and (3,8) are rearranged, the new session (5,2) shares a directed wavelength with session (1,5) on the directed wavelength previously used by session (3,8), while session (3,8) is supported on the directed wavelength previously used by session (1,5).

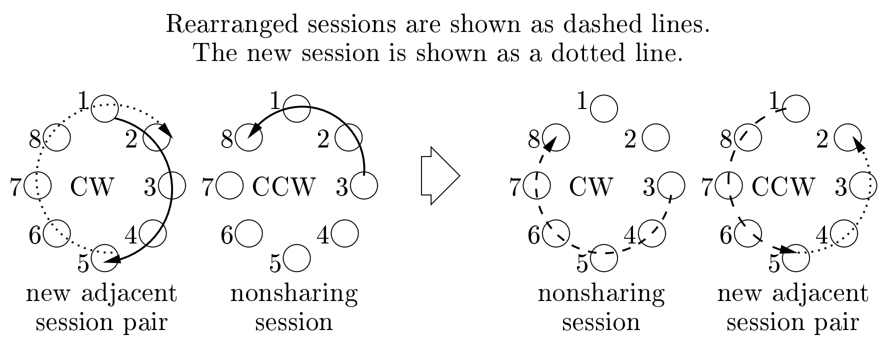


Figure 4-14: Step 3b case 1 of the on-line ring RWA algorithm.

When the adjacent session pair does not include the new session, sharing of the directed wavelength by the adjacent session pair will free two directed wavelengths on which the removed nonsharing session and the new session can be supported. In this case, a total of three lightpath rearrangements are made. Figure 4-15 illustrates this scenario. In particular, existing sessions (1,5) and (5,2) form an adjacent session pair which can be supported on the directed wavelength of existing session (3,8). After the lightpaths of sessions (1,5), (5,2), and (3,8) are rearranged, the adjacent session pair (1,5) and (5,2) are supported on the directed wavelength previously used by session (3,8), session (3,8) is supported on the directed wavelength previously used by session (1,5), and the new session (1,4) is supported on the directed wavelength previously used by session (5,2).

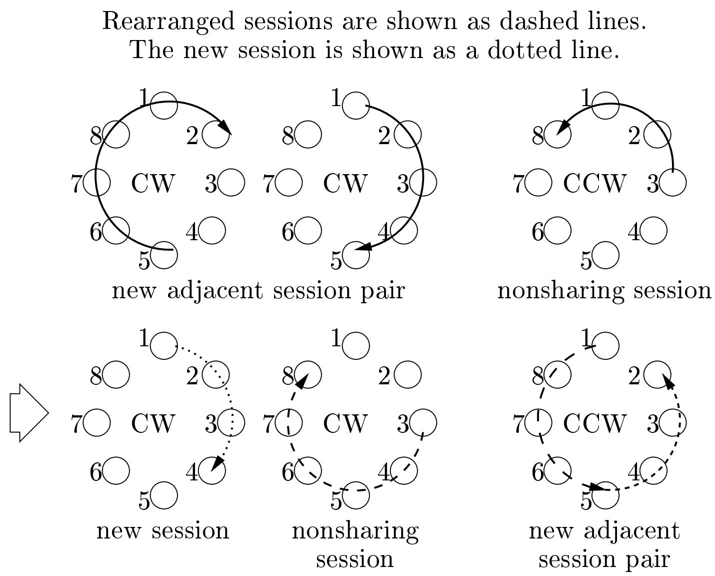


Figure 4-15: Step 3b case 2 of the on-line ring RWA algorithm.

Before proving the correctness of the on-line ring RWA algorithm, we establish two useful lemmas related to step 3 of the algorithm. The first lemma gives an upper bound on the number of adjacent session pairs which share a directed wavelength in step 3 before the new session request. The second lemma gives an upper bound on the number of nonsharing sessions in step 3 before the new session request. In what follows, let p be the number of adjacent session pairs which share a directed wavelength before the new session request. Let q be the number of nonsharing sessions before the new session request. Let w be the number of wavelengths in use before the new session request. Note that $w = p + q$. For convenience, define $K = \sum_{i=1}^{N} k_i$.

Lemma 10 *In step 3 of the on-line ring RWA algorithm, $p < \lfloor K/3 \rfloor$.*

Proof: Since the total number of sessions is at most K in **k**-allowable traffic, it follows that $2p + q < K$ before the new session request. Thus, w is bounded by

$$w \;=\; p + q \;<\; p + (K - 2p) \;=\; K - p.$$

In step 3, since there is no free directed wavelength for the new session, it follows that the number of wavelengths in use w is equal to the total number of directed wavelengths $2\lceil K/3 \rceil$. Therefore, $K - p > w = 2\lceil K/3 \rceil$, yielding the desired relation

$$p \;<\; K - 2\lceil K/3 \rceil \;\leq\; \lfloor K/3 \rfloor. \qquad \square$$

Lemma 11 *In step 3 of the on-line ring RWA algorithm, if no adjacent session pair can be formed among the nonsharing sessions and the new session, then* $q \leq \lfloor (K-p)/2 \rfloor$.

Proof: Note that node i, $1 \leq i \leq N$, is equipped with k_i tunable transmitter/receiver pairs. Overall, we have a total of K transmitter/receiver pairs. Each pair of adjacent sessions which shares a directed wavelength utilizes one transmitter/receiver pair at some node, one transmitter at another node, and one receiver at yet another node.

Let p_i be the number of adjacent session pairs which share a directed wavelength and have node i as a common node. Since an adjacent session pair may have more than one common node, $\sum_{i=1}^{N} p_i \geq p$. Let $k_i' = k_i - p_i$ denote the number of transmitter/receiver pairs which are not used by those p_i adjacent session pairs at node i. Note that $k_i' + p_i = k_i$. In addition, let k_i^t and k_i^r denote the number of nonsharing sessions transmitted and received at node i respectively. It is clear that $k_i^t \leq k_i'$ and $k_i^r \leq k_i'$.

Since no new adjacent session pair can be formed among the nonsharing sessions, it follows that, at each node i, either $k_i^t = 0$ or $k_i^r = 0$. Thus, $k_i^t + k_i^r \leq k_i'$. Because each nonsharing session uses one transmitter and one receiver, it follows that

$$2q \;=\; \sum_{i=1}^{N} (k_i^t + k_i^r) \;\leq\; \sum_{i=1}^{N} k_i' \;=\; K - \sum_{i=1}^{N} p_i \;\leq\; K - p.$$

Since q is an integer, it follows that $q \leq \lfloor (K-p)/2 \rfloor$. □

Proof of algorithm correctness: From the algorithm description, it is clear that we always keep the two desired RWA conditions, i.e. (i) only adjacent sessions share a directed wavelength, and (ii) at most two adjacent sessions share a directed wavelength. In addition, it is clear that at most three lightpath rearrangements are made to support each new session request.

It remains to prove the two claims in step 3. The first claim states that there always exists a new adjacent session pair. We proceed by contradiction. Suppose that no new adjacent session pair can be formed among the nonsharing sessions and the new session. From lemma 11, $q \leq \lfloor (K-p)/2 \rfloor$. Since there is no free directed wavelength for the new session in step 3, it follows that the number of wavelengths in use w is equal to the total number of directed wavelengths $2\lceil K/3 \rceil$. Therefore,

$$p + \lfloor (K-p)/2 \rfloor \;\geq\; p + q \;=\; w \;=\; 2\lceil K/3 \rceil.$$

It follows that

$$p \;\geq\; 2\lceil K/3\rceil - \lfloor (K-p)/2\rfloor \;\geq\; 2K/3 - (K-p)/2,$$

or equivalently, $p \geq K/3$, which contradicts the fact that $p < \lfloor K/3\rfloor$ in step 3 from lemma 10. Hence, a new adjacent session pair always exists in step 3.

We now prove the second claim in step 3 that if we need to find a nonsharing session in the opposite ring direction, i.e. the ring direction in which the new adjacent session pair can share a directed wavelength, one always exists. The claim is a direct consequence of lemma 10, i.e. $p < \lfloor K/3\rfloor$ in step 3. In other words, the number of sharing session pairs is less than the number of directed wavelengths in each ring direction. Since step 2 was not taken, all the other $2\lceil K/3\rceil - p$ directed wavelengths are taken by nonsharing paths. Therefore, in either ring direction, a directed wavelength with a nonsharing session exists. □

The construction of the on-line ring RWA algorithm implies the following theorem.

Theorem 7 *For a bidirectional ring with N nodes and $\mathbf{k}$-allowable traffic, $W_{d,\mathbf{k}}$ is upper bounded by*

$$W_{d,\mathbf{k}} \quad\leq\quad \left\lceil \frac{\sum_{i=1}^{N} k_i}{3} \right\rceil.$$

In addition, there exists, by construction, an on-line RWA algorithm which uses $\lceil(\sum_{i=1}^{N} k_i)/3\rceil$ wavelengths in each fiber and requires at most three lightpath rearrangements per new session request.

When $N \geq 7$ and all the k_i's are equal to k (i.e. symmetric k-allowable traffic), it was shown in [NLM02] that $W_{d,\mathbf{k}} = \lceil Nk/3\rceil$. In this case, the above upper bound is tight. Otherwise, the above upper bound is not necessarily tight and our algorithm may use more than the minimum number of wavelengths. An interesting example is an N-node bidirectional ring which contains one hub node, say node 1, with $k_1 = N-1$, and the other $N-1$ nodes each with $k_i = 1$. We shall show in the next section that, in this case, $W_{d,\mathbf{k}} = \lceil (N-1)/2\rceil$, which is less than the upper bound $\lceil 2(N-1)/3\rceil$ from theorem 7. To do so, we develop an on-line RWA algorithm which uses $\lceil (N-1)/2\rceil$ wavelengths and requires at most four lightpath rearrangements per new session request.

The following example illustrates the operations of the on-line ring RWA algorithm.

Example 7 Consider symmetric 1-allowable traffic in the 6-node bidirectional ring. The on-line ring RWA algorithm uses two wavelengths, or equivalently two CW directed wavelengths and two CCW directed wavelengths. Assume the traffic matrix in which each node transmits a wavelength to the node two hops away in the CCW ring direction. In addition, assume that the current RWA on the four directed wavelengths is as given in figure 4-16a.

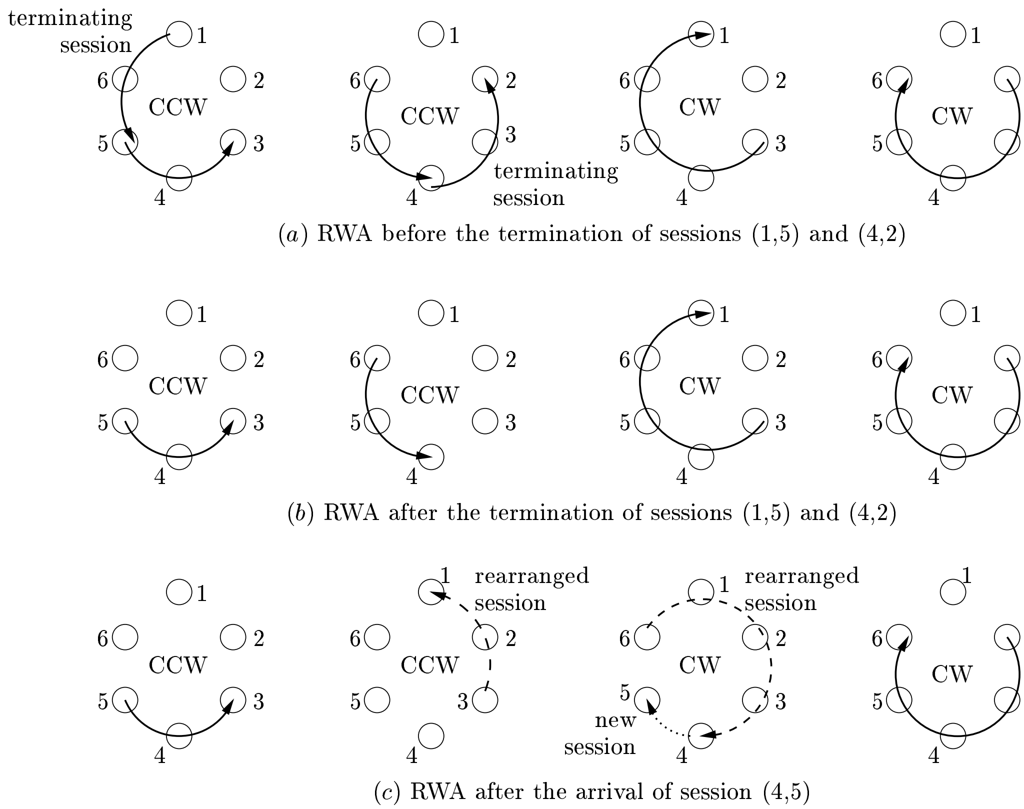


Figure 4-16: Example operations of the on-line ring RWA algorithm.

Now assume the following changes in the traffic matrix.

1. Existing session (1,5) terminates.
2. Existing session (4,2) terminates.
3. A new session (4,5) arrives.

After the termination of sessions (1,5) and (4,2), the RWA is shown in figure 4-16b. When the new session (4,5) arrives, it forms an adjacent session pair with either session (5,3) or session (6,4). In either case, the new session cannot share the directed wavelength of the existing session in the pair. Thus, the algorithm cannot perform step 1. Since there is no free directed wavelength, the algorithm does not perform step 2. In this example, the algorithm performs step 3a case 1. There are multiple possible RWA updates in this step. In one possible RWA update, existing session (3,1) is rearranged from its CW directed wavelength to the CCW directed wavelength previously used by session (6,4). The adjacent session pair (6,4) and (4,5) is then supported on the freed CW directed wavelength. There are two lightpath rearrangements made (corresponding to sessions (3,1) and (6,4)), as shown in figure 4-16c.

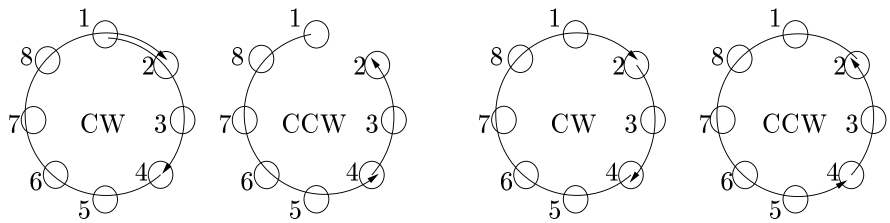
4.3.1 RWA for a Single-Hub Bidirectional Ring

In this subsection, we give an example scenario for $\mathbf{k}$-allowable traffic in which the on-line ring RWA algorithm does not use the minimum number of wavelengths. Consider a bidirectional ring with N nodes. In particular, node 1 acts as a hub node with $k_1 = N-1$. In addition, for $2 \leq i \leq N$, $k_i = 1$. Note that the non-hub nodes can directly transmit and/or receive wavelengths among themselves.

We first derive a lower bound on the minimum number of wavelengths $W_{d,\mathbf{k}}$. Consider a cut set corresponding to the two links adjacent to the hub node. The maximum traffic across the two fibers leaving from the hub occurs when the hub node transmits $N-1$ wavelengths. Since there are $N-1$ wavelengths traveling on two fibers, one fiber must support at least $\lceil (N-1)/2 \rceil$ wavelengths. Thus, $W_{d,\mathbf{k}} \geq \lceil (N-1)/2 \rceil$.

We prove informally below that $W_{d,\mathbf{k}} \leq \lceil (N-1)/2 \rceil$, yielding $W_{d,\mathbf{k}} = \lceil (N-1)/2 \rceil$. Our formal proof is based on a new on-line RWA algorithm for a single-hub ring which uses $\lceil (N-1)/2 \rceil$ wavelengths and is given in appendix C. Note that the general on-line ring RWA algorithm given earlier uses $\lceil 2(N-1)/3 \rceil$ wavelengths, which is greater than the minimum number of wavelengths.

As in the general on-line ring RWA algorithm, the main idea of our new RWA algorithm involves sharing of a directed wavelength by an adjacent session pair. In addition, we define a special kind of adjacent session pairs as described next. Two sessions form a *mutual adjacent* session pair if they have two common nodes, i.e. the source node of one session is the destination node of the other session and vice versa. For convenience, we refer to an adjacent session pair which is not mutually adjacent as a *nonmutual adjacent* session pair. While a nonmutual adjacent session pair can share a directed wavelength in only one ring direction, a mutual adjacent session pair can share a directed wavelength in any ring direction, as shown in figure 4-17. In particular, the nonmutual adjacent session pair (1,4) and (4,2) can share a CCW directed wavelength, but not a CW directed wavelength. On the other hand, the mutual adjacent session pair (2,4) and (4,2) can share a directed wavelength in any ring direction.



The nonmutual adjacent session pair (1,4) and (4,2) can share a directed wavelength in only one ring direction.

The mutual adjacent session pair (2,4) and (4,2) can share a directed wavelength in any ring direction.

Figure 4-17: Supporting a mutual adjacent session pair on a directed wavelength.

We shall refer to an adjacent session pair in which the hub node is one common node as an adjacent session pair *at the hub*. Our RWA is based on the following two RWA conditions: (i) only adjacent session pairs at the hub share a directed wavelength, and (ii) all mutual adjacent session pairs at the hub share a directed wavelength.

Below is our informal proof that $\lceil (N-1)/2 \rceil$ wavelengths are sufficient to support the traffic. We ignore integer rounding in the informal discussion below.

Given a traffic matrix, form all the mutual adjacent session pairs at the hub, but do not assign directed wavelengths for them at this point. Then form all the nonmutual adjacent session pairs at the hub. Let r and s denote the number of mutual and nonmutual adjacent session pairs at the hub respectively. Let t be the number of the remaining sessions. Note that we cannot form any new adjacent session pair at the hub among these t sessions.

We first support the s nonmutual adjacent session pairs at the hub on s directed wavelength in the required ring directions. We now show this is always possible. Observe that each non-hub node terminates, i.e. transmits or receives, at most one session in these s adjacent pairs. To see this, note that each non-hub node cannot transmit (receive) more than one session since it only has one transmitter (receiver). Moreover, each non-hub node cannot transmit a session and receive a session in these s adjacent pairs simultaneously, or else we can form another mutual adjacent session pair at the hub. It follows that each nonmutual adjacent session pair at the hub is terminated at two non-hub nodes, and no other nonmutual adjacent session pair at the hub is terminated at any of these two nodes. Since there are $N-1$ non-hub nodes, it follows that $s \leq (N-1)/2$. Since there are $(N-1)/2$ directed wavelengths available in each ring direction, there are enough wavelengths to support the s session pairs.

We next support the r mutual adjacent session pairs at the hub on any r unused directed wavelengths. We now show this is always possible. Note that each mutual adjacent session pair at the hub is terminated at one distinct non-hub node. From the above discussion, each nonmutual adjacent session pair at the hub is terminated at two distinct non-hub nodes. Since there are $N-1$ non-hub nodes, it follows that $r + 2s \leq N - 1$, or equivalently $r \leq (N-1) - 2s$. Since there are $(N-1) - s$ unused directed wavelengths left for this step, the inequality $r \leq (N-1) - 2s$ implies that there are enough directed wavelengths to support the r session pairs.

In the final step, we support the t remaining sessions on any t unused directed wavelengths. We now show this is always possible. Since we cannot form any adjacent session pair at the hub from these t sessions, the hub node can either transmit or receive some or all of these t sessions but not both. Without loss of generality, assume that the hub node transmits none of these t sessions. Consider the transmitters at the non-hub nodes. Each of the r mutual adjacent session pairs at the hub uses one transmitter at some non-hub node. Similarly, each of the s nonmutual adjacent session pairs at the hub uses one transmitter at some non-hub node. Since the hub node does not transmit any of the t remaining sessions, each of the t sessions uses one transmitter at some non-hub node. Since there are $N-1$ non-hub nodes, it follows that $r + s + t \leq N - 1$, or equivalently $t \leq (N-1) - r - s$. Since there are $(N-1) - r - s$ unused directed wavelengths left for this step, there are enough directed wavelengths to support the remaining t sessions.

Based on the above main idea of our RWA, we can construct an on-line RWA algorithm which uses $\lceil (N-1)/2 \rceil$ wavelengths in each fiber, is rearrangeably nonblocking, and requires at most four lightpath rearrangements per new session request. We shall refer to this algorithm as the *on-line single-hub ring RWA algorithm.* We present the algorithm and its correctness proof in appendix C.

4.3.2 Bidirectional Ring with Wavelength Converters

In this subsection, we give an example in which wavelength converters can reduce the number of wavelengths required to support $\mathbf{k}$-allowable traffic in a bidirectional ring. Let $L_{d,\mathbf{k}}$ denote the minimum number of wavelengths which, if provided in each fiber, can support $\mathbf{k}$-allowable traffic given full wavelength conversion at all nodes. This example implies that, for a bidirectional ring, it is possible that $L_{d,\mathbf{k}} < W_{d,\mathbf{k}}$.

Consider the 7-node ring with symmetric 1-allowable traffic, i.e. the $\mathbf{k}$-allowable traffic in which all the k_i's are equal to 1. We know that $W_{d,\mathbf{k}} = \lceil 7/3 \rceil = 3$. We shall show below that $L_{d,\mathbf{k}} = 2$.

To derive the lower bound $L_{d,\mathbf{k}} \geq 2$, consider the cut set which separates the ring into two connected subnetworks with three and four nodes respectively. It is easy to see that the maximum traffic across the two fibers from the 3-node subnetwork to the 4-node subnetwork is three wavelengths. Since there are three wavelengths travelling on two fibers across the cut, one fiber must support at least $\lceil 3/2 \rceil = 2$ wavelengths. Thus, $L_{d,\mathbf{k}} \geq 2$.

We now show that $L_{d,\mathbf{k}} \leq 2$. Without loss of generality, we shall assume that any given symmetric 1-allowable traffic matrix is *maximal* in the sense that we cannot add an extra session to the traffic matrix (except perhaps for self-traffic which we do not consider). When the traffic matrix is not maximal, we can add extra sessions to make it maximal, solve the RWA problem, and then remove the extra sessions. It is easy to see that, in any maximal traffic matrix, the sessions form a set of cycles.

For symmetric 1-allowable traffic in the 7-node ring, there are eight possible scenarios for the set of cycles, as listed below.

1. Three 2-cycles.[5]
2. Two 2-cycles and one 3-cycle.
3. One 2-cycle and one 4-cycle.
4. One 2-cycle and one 5-cycle.
5. Two 3-cycles.
6. One 3-cycle and one 4-cycle.
7. One 6-cycle.
8. One 7-cycle.

In scenarios 1, 3, 5, and 7, we can ignore the node which neither transmits nor receives traffic and view the network as the 6-node ring. Equation (4.2) tells us that $W_{d,\mathbf{k}} = \lceil 6/3 \rceil = 2$ wavelengths are sufficient. We consider scenarios 2, 4, 6, and 8 separately below. Notice that only scenario 8 requires wavelength converters.

[5] An m-cycle is a cycle which contains m sessions.

Two 2-cycles and one 3-cycle: Viewing the 3-cycle as being in the 3-node ring, equation (4.2) tells us that $W_{d,\mathbf{k}} = \lceil 3/4 \rceil = 1$ wavelength can support the 3-cycle. Since each 2-cycle is a mutual adjacent session pair, it can be supported on one directed wavelength in any ring direction. It follows that one wavelength (two directed wavelengths) can support the two 2-cycles. Thus, two wavelengths are sufficient to support the traffic.

One 2-cycle and one 5-cycle: From the 5-cycle, form two adjacent session pairs and support them on two directed wavelengths in the required ring directions. The remaining session from the 5-cycle can be supported on one directed wavelength in any ring direction. Similarly, one directed wavelength in any ring direction can support the 2-cycle. Thus, two wavelengths (four directed wavelengths) are sufficient to support the traffic.

One 3-cycle and one 4-cycle: Viewing the 4-cycle as being in the 4-node ring, equation (4.2) tells us that $W_{d,\mathbf{k}} = 1$ wavelength can support the 4-cycle. From the above discussion, one wavelength can support the 3-cycle. Thus, two wavelengths are sufficient to support the traffic.

One 7-cycle: In this scenario, we use the argument from [CM02]. Let H_{CW} and H_{CCW} be the total number of hops traversed by all the sessions in the CW and CWW ring directions respectively. Since a session which traverses x hops in the CW direction traverses $7 - x$ hops in the CCW direction, $H_{CCW} = 49 - H_{CW}$. Since all the sessions form a cycle in the 7-node ring, the possible values of H_{CW} are 7, 14, 21, 28, 35, and 42. If H_{CW} is equal to 7 or 14, routing all the sessions in the CW direction incurs the maximum fiber load of two wavelengths, and thus two wavelengths are sufficient to support the traffic. If H_{CW} is equal to 35 or 42, then H_{CCW} is equal to 14 or 7, and thus routing all the sessions in the CCW direction requires at most two wavelengths.

Let us now consider the case with $H_{CW} = 21$. We claim that, when $H_{CW} = 21$, there must exist a set of four adjacent sessions which traverse at most 12 hops in total in the CW direction. To justify the claim, let $x_0, x_1, x_2, x_3, x_4, x_5$, and x_6 denote the numbers of hops traversed in the CW direction by all the sessions in the adjacent order. We prove the claim by contradiction. Assume that every set of four adjacent sessions traverse more than 12 hops in the CW direction. Then we have the inequalities $x_i + x_{i+1 \bmod 7} + x_{i+2 \bmod 7} + x_{i+3 \bmod 7} > 12$ for all $0 \leq i \leq 6$, e.g. the inequality $x_0 + x_2 + x_3 + x_4 > 12$ corresponds to $i = 0$. By summing all the inequalities over all

$0 \leq i \leq 6$, we have that $4(x_0 + x_1 + ... + x_6) > 84$, yielding $H_{CW} = x_0 + x_1 + ... + x_6 > 21$, contradicting the fact that $H_{CW} = 21$.

Given a set of four adjacent sessions which traverse at most 12 hops in total in the CW direction, we can support them on two CW directed wavelengths. Since $H_{CW} = 21$, the remaining three adjacent sessions traverse at least 9 hops in total in the CW direction, or equivalently at most 12 hops in the CCW direction. It follows that we can support them on two CCW directed wavelengths. In conclusion, two wavelengths are sufficient to support the traffic.

Finally, when $H_{CW} = 28$, we have $H_{CCW} = 21$. By exchanging the roles of the CW and CCW ring directions, the same arguments as for the case with $H_{CW} = 21$ can be applied to argue that two wavelengths are sufficient to support the traffic.

In conclusion, we have shown that, for symmetric 1-allowable traffic in the 7-node bidirectional ring, $L_{d,\mathbf{k}} = 2 < W_{d,\mathbf{k}} = 3$. More generally, it is shown in [CM02] that, in the N-node bidirectional ring, for symmetric k-allowable traffic, $L_{d,\mathbf{k}}$ is bounded by

$$\begin{array}{rcccll} \lceil Nk/4 \rceil & \leq & L_{d,\mathbf{k}} & \leq & \lceil Nk/4 \rceil + 1, & N \text{ even}, \\ \lceil (N-1)k/4 \rceil & \leq & L_{d,\mathbf{k}} & \leq & \lceil Nk/4 \rceil + 1, & N \text{ odd}. \end{array}$$

Since $W_{d,\mathbf{k}} = \lceil Nk/3 \rceil$ for $N \geq 7$, it is clear that wavelength converters can reduce the wavelength requirement for symmetric k-allowable traffic for a sufficiently large value of N.

4.4 2D Torus Topologies

In this section, we study the RWA problem for $\mathbf{k}$-allowable traffic in a two-dimensional (2D) torus topology. We shall consider only symmetric k-allowable traffic, i.e. $\mathbf{k}$-allowable traffic in which all the k_i's are equal to k. The RWA problem for general $\mathbf{k}$-allowable traffic remains to be investigated in the future.

Consider an $R \times C$ torus topology with N nodes, where $N = RC$ and $R \geq C$. Let $L_{d,\mathbf{k}}$ denote the minimum number of wavelengths which, if provided in each fiber, can support $\mathbf{k}$-allowable traffic given full wavelength conversion at all nodes. We first derive a lower bound on $L_{d,k}$.

Lemma 12 *For an $R \times C$ torus topology with $R \geq C$, $L_{d,k} \geq \lceil k(R-1)/4 \rceil$.*

Proof: For R even, consider a cut set which separates $R/2$ consecutive rows of nodes from the other $R/2$ consecutive rows. Assume a traffic matrix in which each node transmits k wavelengths to a node in the other set. In this traffic, a total of $kRC/2$ sessions travel from one set of nodes to the other set of nodes on $2C$ fibers. It follows that one fiber connecting the two sets of nodes must support at least $\left\lceil \frac{kRC/2}{2C} \right\rceil = \lceil kR/4 \rceil$ wavelengths. Thus, $L_{d,k} \geq \lceil kR/4 \rceil$.

For R odd, consider a cut set which separates $(R-1)/2$ consecutive rows of nodes from the other $(R+1)/2$ consecutive rows. Assume a traffic matrix in which each node in the smaller set transmits k wavelengths to a node in the other set. In this traffic, a total of $kC(R-1)/2$ sessions travel from one set of nodes to the other set of nodes on $2C$ fibers. It follows that one fiber connecting the two sets of nodes must support at least $\left\lceil \frac{kC(R-1)/2}{2C} \right\rceil = \lceil k(R-1)/4 \rceil$ wavelengths. Thus, $L_{d,k} \geq \lceil k(R-1)/4 \rceil$.

In conclusion, for a general (odd or even) positive integer R, $L_{d,k} \geq \lceil k(R-1)/4 \rceil$. □

We shall construct an RWA algorithm which uses $\lceil kR/2 \rceil$ wavelengths in each fiber. Let $W_{d,\mathbf{k}}$ denote the minimum number of wavelengths which, if provided in each fiber, can support **k**-allowable traffic with no wavelength conversion. The algorithm yields the upper bound $W_{d,k} \leq \lceil kR/2 \rceil$. This upper bound on $W_{d,k}$ is about twice the value of our lower bound on $L_{d,k}$.

Define a *directed wavelength* in a 2D torus topology as follows. Each wavelength consists of an upward directed wavelength and a downward directed wavelength as described next. An upward directed wavelength is directed upwards along any column and to the right along any row, as as illustrated in figure 4-18a. On the other hand, a downward directed wavelength is directed downwards along any column and to the left along any row, as illustrated in figure 4-18b.

We shall apply *column-first routing* where each lightpath travels along the source column and then along the destination row. In addition, each lightpath is supported by no more than one directed wavelength, i.e. if it travels upwards along the source column, then it must travel to the right along the destination row according to the definition of a directed wavelength. The main idea of our RWA algorithm is based on the following observation.

Lemma 13 *For an $R \times C$ torus topology, under column-first routing, a set of sessions from distinct source columns to distinct destination rows can all be supported on a single directed wavelength, which can be either upward or downward directed.*

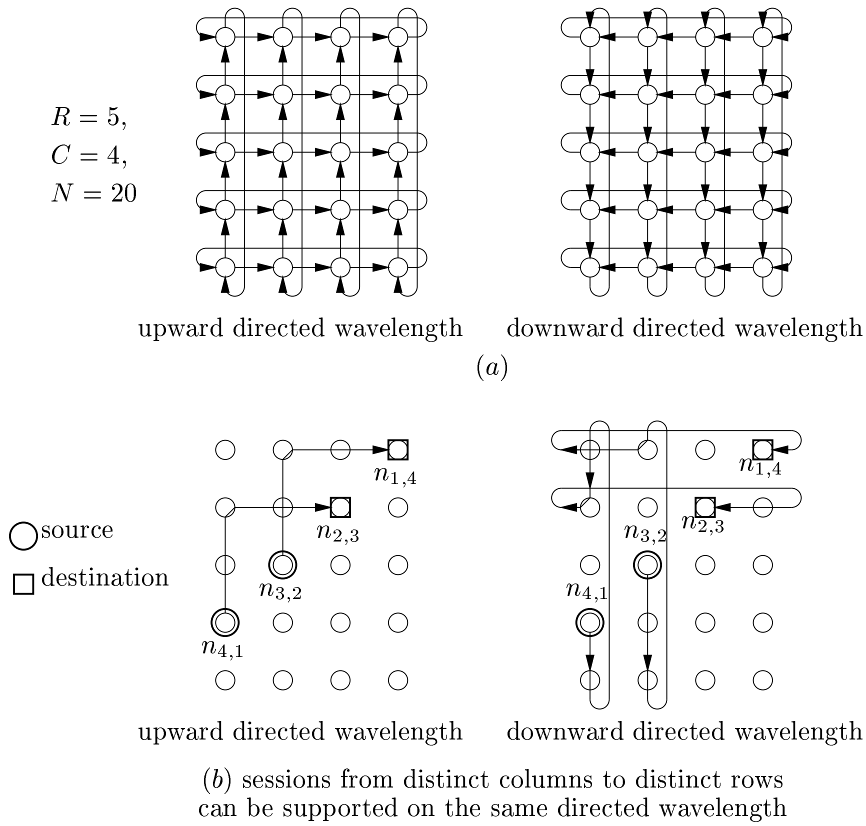


Figure 4-18: Directed wavelength and its supported sessions.

Proof: Since the sessions come from distinct source columns, at most one session utilizes the fibers in a given column. Similarly, since the sessions go to distinct destination rows, at most one session utilizes the fibers in a given row. It follows that there is no wavelength collision on any fiber in the network. □

Let $n_{i,j}$ denote the node in row i and column j. Let $(n_{i,j}, n_{k,l})$ denote a session from $n_{i,j}$ to $n_{k,l}$. Figure 4-18b illustrates the statement of lemma 13. In particular, there are two sessions $(n_{4,1}, n_{2,3})$ and $(n_{3,2}, n_{1,4})$ which are transmitted from two distinct source columns to two distinct destination rows. The two sessions can be supported on either an upward or a downward directed wavelength.

We can view the set of sessions from distinct source columns to distinct destination rows as a matching in a bipartite graph. For a given traffic matrix, we can construct the *column-to-row* bipartite graph, denoted by $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$, as follows. The set of abstract nodes $\mathcal{V}_1$ contains C nodes corresponding to the C source columns. The set of abstract nodes $\mathcal{V}_2$ contains R nodes corresponding to the R destination rows. In the set of edges $\mathcal{E}$, an edge between node i in $\mathcal{V}_1$ and node j in $\mathcal{V}_2$ corresponds to a session from a source in column i to a destination in row j. Figure 4-19(a-b) shows an example of the column-to-row bipartite graph and its traffic matrix. Note that there may be multiple edges between the same pair of nodes. For example, since there are two sessions from $\mathcal{C}_3$ to $\mathcal{R}_4$, i.e. $(n_{2,3}, n_{4,2})$ and $(n_{4,3}, n_{4,1})$, there are two parallel edges between $\mathcal{C}_3$ and $\mathcal{R}_4$.

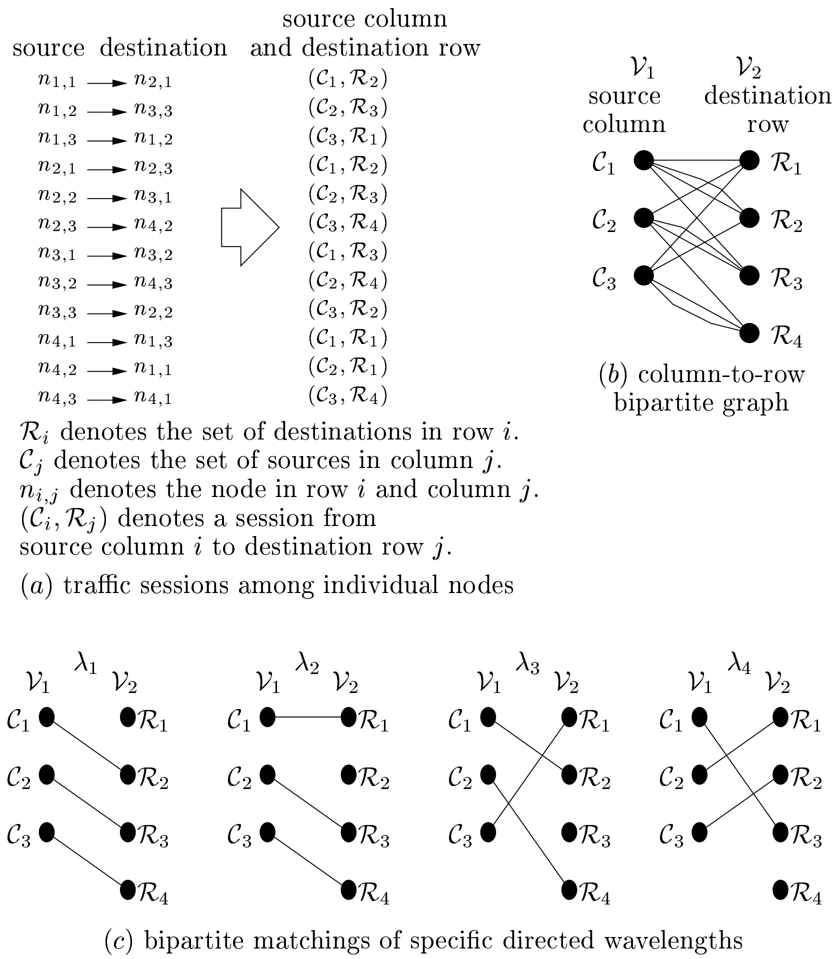


Figure 4-19: Column-to-row bipartite graph.

Figure 4-19c shows one possible partition of the set of edges $\mathcal{E}$ into four disjoint bipartite matchings. Observe that the sessions belonging to a matching in the column-to-row bipartite graph are transmitted from distinct source columns to distinct destination rows. From lemma 13, these sessions can be supported on one directed wavelength using column-first routing. Our algorithm will assign a single bipartite matching to a single directed wavelength. In what follows, we shall refer to the bipartite matching assigned to directed wavelength λ_1 simply as the bipartite matching of λ_1. Figure 4-19c shows example bipartite matchings of specific directed wavelengths. Note that there are at most C sessions in each matching.

Before presenting our on-line RWA algorithm, we derive a few useful lemmas related to bipartite matchings. The following lemma is a slightly more general version of lemma 6 in section 4.1. The difference is that we assume $|\mathcal{V}_1| \leq |\mathcal{V}_2|$ instead of $|\mathcal{V}_1| = |\mathcal{V}_2|$.

Lemma 14 *In a bipartite graph* $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$ *with* $|\mathcal{V}_1| = C \leq |\mathcal{V}_2|$*, if each node has degree at most* m*, the set* $\mathcal{E}$ *can be partitioned into* m *disjoint bipartite matchings.*

Proof: If $|\mathcal{V}_1| = |\mathcal{V}_2| = C$, then lemma 6 can be applied. It remains to consider the cases with with $|\mathcal{V}_1| = C < |\mathcal{V}_2|$.

From $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$, we can construct a new bipartite graph, denoted by $(\mathcal{V}_1', \mathcal{V}_2, \mathcal{E})$, as follows. Add $|\mathcal{V}_2| - C$ dummy nodes to the set $\mathcal{V}_1$ to create the modified set of nodes $\mathcal{V}_1'$. The set of nodes $\mathcal{V}_2$ and the set of edges $\mathcal{E}$ are the same as before. In the resultant bipartite graph $(\mathcal{V}_1', \mathcal{V}_2, \mathcal{E})$, $|\mathcal{V}_1'| = |\mathcal{V}_2|$ and each node has degree at most m. From lemma 6, we can obtain m disjoint bipartite matchings, denoted by $\mathcal{M}_1$, $\mathcal{M}_2$, ..., and $\mathcal{M}_m$. Since each bipartite matching $\mathcal{M}_i$, $1 \leq i \leq m$, is also a matching in the original bipartite graph $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$, it follows that $\mathcal{M}_1$, $\mathcal{M}_2$, ..., and $\mathcal{M}_m$ obtained in this fashion are the desired m disjoint bipartite matchings. □

Lemma 14 can be used to argue that kR directed wavelengths are sufficient to support any traffic matrix in the symmetric k-allowable set. Given a traffic matrix, we can write down the corresponding column-to-row bipartite graph in which each node has degree at most kR. By lemma 14, the set of edges can be partitioned into kR disjoint bipartite matchings. The sessions on each bipartite matching can be supported on a single directed wavelength. Therefore, kR directed wavelengths are sufficient to support any symmetric k-allowable traffic matrix.

Our on-line RWA algorithm for a 2D torus topology is constructed in a similar fashion to the on-line star WA algorithm in section 4.1. Both algorithms involve finding matchings in a bipartite graph. The main difference has to do with what a node in a bipartite graph represents. In the on-line star WA algorithm, a node represents a single source or a single destination. In this section, a node represents a source column or a destination row.

The main idea of our on-line RWA algorithm involves keeping kR disjoint bipartite matchings of kR directed wavelengths such that each traffic session corresponds to an edge in one bipartite matching. When a session departs, we simply remove its corresponding lightpath from the network. When a new session, say $(\mathcal{C}_i, \mathcal{R}_j)$, arrives, we find one directed wavelength which is not used by any source in column i, and one directed wavelength which is not used by any destination in row j. If the two directed wavelengths are the same, we can support the new session without any lightpath rearrangement. Otherwise, we rearrange some existing lightpaths on the two directed wavelengths to support the new session. The following lemma makes the above discussion concrete and states an upper bound on the number of lightpath rearrangements. Note that the lemma is slightly more general than lemma 7 in section 4.1 since we assume $|\mathcal{V}_1| \leq |\mathcal{V}_2|$ instead of $|\mathcal{V}_1| = |\mathcal{V}_2|$.

Lemma 15 *In a bipartite graph $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$ with $|\mathcal{V}_1| = C \leq |\mathcal{V}_2|$, given a new edge $(\mathcal{C}_i, \mathcal{R}_j)$, $\mathcal{C}_i \in \mathcal{V}_1$, $\mathcal{R}_j \in \mathcal{V}_2$, a matching $\mathcal{M}_1$ of directed wavelength λ_1 which is not incident on $\mathcal{C}_i$, and a matching $\mathcal{M}_2$ of directed wavelength λ_2 which is not incident on $\mathcal{R}_j$, there exist two disjoint bipartite matchings which contain all the edges in $\mathcal{M}_1$ and $\mathcal{M}_2$ as well as the new edge $(\mathcal{C}_i, \mathcal{R}_j)$.*

In addition, these two disjoint bipartite matchings can be assigned to λ_1 and λ_2 so that the number of lightpath rearrangements is at most $C - 1$.

Proof: The proof is identical to the proof of lemma 7. The only difference is that, in this proof, we use lemma 14 instead of lemma 6 to argue the existence of the two disjoint bipartite matchings which contain all the edges in $\mathcal{M}_1$ and $\mathcal{M}_2$ as well as the new edge $(\mathcal{C}_i, \mathcal{R}_j)$. We shall not repeat the details here. □

The following is our on-line RWA algorithm for a 2D torus topology with symmetric k-allowable traffic. The algorithm uses $\lceil kR/2 \rceil$ wavelengths in each fiber, is rearrangeably nonblocking, and requires at most $C - 1$ lightpath rearrangements per new session request. We shall refer to this algorithm as the *on-line torus RWA algorithm*

Algorithm 5 (On-Line Torus RWA Algorithm) (Use $\lceil kR/2 \rceil$ wavelengths in each fiber.)

Session termination: When a session terminates, simply remove its associated lightpath from the network without any further lightpath rearrangement.

Session arrival: When a session arrives and it is allowable, proceed as follows. Let i and j denote the source column and the destination row of the new session.

Step 1: If there is a directed wavelength, denoted by λ_0, which is used by neither a source in column i nor a destination in row j, then assign the new session to λ_0, and use column-first routing. In this case, no lightpath rearrangement is required. Otherwise, proceed to step 2.

Step 2: Find a directed wavelength, denoted by λ_1, which is not used by any source in column i, i.e. its bipartite matching is not incident on $\mathcal{C}_i$, and another directed wavelength, denoted by λ_2, which is not used by any destination in row j, i.e. its bipartite matching is not incident on $\mathcal{R}_j$. We claim and shall prove shortly that λ_1 and λ_2 exist.

Modify the RWA of only the sessions on λ_1 and λ_2. Construct the column-to-row bipartite graph $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E}')$ in which the set of edges $\mathcal{E}'$ contains the bipartite matchings of λ_1 and λ_2 as well as the new edge $(\mathcal{C}_i, \mathcal{R}_j)$. Notice that $|\mathcal{V}_1| = C \leq R = |\mathcal{V}_2|$ and each abstract node has degree at most 2. From lemma 15, the set $\mathcal{E}'$ can be partitioned into two disjoint bipartite matchings. In addition, lemma 15 tells us that the two matchings can be assigned to λ_1 and λ_2 such that at most $C-1$ existing lightpaths need to be rearranged.

Proof of algorithm correctness: It remains to prove the claim in step 2, which states that the directed wavelengths λ_1 and λ_2 as defined in step 2 must exist. We shall prove the existence of λ_1. Similar arguments can be used to prove the existence of λ_2. Since the new session is allowable, there are at most $kR-1$ sessions transmitted from source column i. Since there are $2\lceil kR/2 \rceil$ directed wavelengths, the number of directed wavelengths available for a session transmitted from source column i is at least

$$2\lceil kR/2 \rceil - (kR-1) \quad \geq \quad kR - (kR-1) \quad \geq \quad 1.$$

Therefore, λ_1 always exists. $\square$

Although we concentrate on an $R \times C$ torus topology with $R \geq C$, similar results can be obtained for an $R \times C$ torus topology with $R \leq C$ by reversing the roles of rows and columns. We summarize the results in this section in the following theorem.

Theorem 8 *For an $R \times C$ torus network with symmetric k-allowable traffic, $W_{d,k}$ is bounded by*

$$\left\lceil \frac{k(\max(R,C)-1)}{4} \right\rceil \quad \leq \quad L_{d,k} \quad \leq W_{d,k} \quad \leq \quad \left\lceil \frac{k\max(R,C)}{2} \right\rceil .$$

In addition, there exists, by construction, an on-line RWA algorithm which uses $\lceil k\max(R,C)/2 \rceil$ wavelengths in each fiber and requires at most $\min(R,C)-1$ lightpath rearrangements per new session request.

As a comparison, when $\min(R,C) = 1$, we have a bidirectional ring with N nodes, where $N = RC$. The torus RWA algorithm in this section uses $\lceil kN/2 \rceil$ wavelengths in each fiber while the ring RWA algorithm specialized for the ring topology uses $\lceil kN/3 \rceil$ wavelengths. Hence, while the torus RWA algorithm is more general, it uses more wavelengths for the ring topology.

The following example illustrates the operations of the on-line torus RWA algorithm.

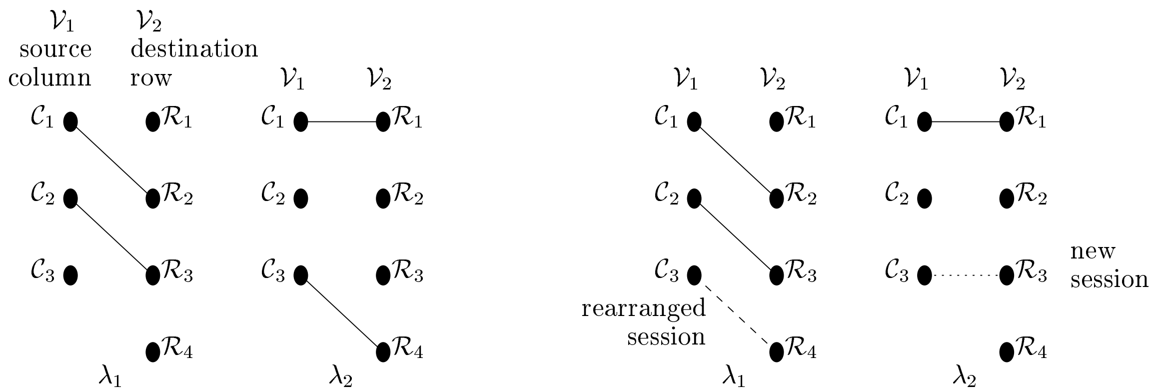
Example 8 Consider the 4×3 torus network with the symmetric 1-allowable traffic given in figure 4-19. Theorem 8 tells us that $W_{d,k} \leq 2$, i.e. four directed wavelengths are sufficient. Assume that the current RWA is given by the bipartite matchings of directed wavelengths λ_1, λ_2, λ_3, and λ_4 in figure 4-19. Now assume the following changes in the traffic.

1. The existing session from $n_{2,3}$ in column 3 to $n_{4,2}$ in row 4 on λ_1 terminates.

2. The existing session from $n_{1,2}$ in column 2 to $n_{3,3}$ in row 3 on λ_2 terminates.

3. A new session from $n_{2,3}$ in column 3 to $n_{3,3}$ in row 3 arrives.

After the second session termination, the bipartite matchings of λ_1 and λ_2 are shown in figure 4-20a. To support the new session, the on-line torus RWA algorithm performs step 2. In particular, it creates a column-to-row bipartite graph whose edges are the bipartite matchings of λ_1 and λ_2 as well as the new edge $(\mathcal{C}_3, \mathcal{R}_3)$. The algorithm then partitions the set of edges into two disjoint bipartite matchings and assigns them to λ_1 and λ_2, as shown in figure 4-20b. In particular, the existing session from $n_{4,3}$ in column 3 to $n_{4,1}$ in row 4 on λ_2 is reassigned to λ_1, and the new session is then assigned to λ_2. In this example, one rearrangement of an existing lightpath is made to support the new session.



(a) bipartite matchings of λ_1 and λ_2

(b) updated bipartite matchings of λ_1 and λ_2

Figure 4-20: Example operations of the on-line torus RWA algorithm.

4.5 Binary Hypercube Topologies

In this section, we briefly mention a known result on the RWA problem for **k**-allowable traffic in a binary hypercube topology with $N = 2^n$ end nodes, where n is a positive integer. We shall concentrate on symmetric 1-allowable traffic, i.e. **k**-allowable traffic in which all the k_i's are equal to 1.

We first derive a lower bound on $L_{d,1}$, the minimum number of wavelengths which, if provided in each fiber, can support 1-allowable traffic given full wavelength conversion at all nodes. Consider symmetric 1-allowable traffic in which each node sends one wavelength to the node which is n hops away. More specifically, if we label the nodes using n-bit binary strings as in section 3.4, then each node transmits a wavelength to the node whose label is the bit-by-bit binary complement of its label, e.g. node 001 transmits to node 110, node 101 transmits to node 010. Given this traffic, all the sessions traverse an aggregate of Nn hops under shortest path routing. Since there are Nn fibers in the N-node binary hypercube, one fiber must support at least $\lceil Nn/Nn \rceil = 1$ wavelength, yielding the trivial lower bound $L_{d,1} \geq 1$.

Since the bound $L_{d,1} \geq 1$ is trivial, the above derivation may seem pointless. However, note that, since each session in the above traffic is between a pair of nodes which are the furthest apart, Nn is the maximum possible total number of hops under shortest path routing. The above discussion suggests that, for any symmetric 1-allowable traffic, one wavelength may be sufficient.

Let $W_{d,1}$ denote the minimum number of wavelengths which, if provided in each fiber, can support 1-allowable traffic with no wavelength conversion. In [AR95], it was shown that $W_{d,1} \leq 8$. The proof of this bound given in [AR95] is based on the construction of an RWA algorithm and is rather involved. We shall not discuss it here. To our knowledge, there is no known example scenario in which one wavelength is not sufficient. Thus, it remains to be investigated whether or not $W_{d,1} = L_{d,1} = 1$. So far, we know that $1 \leq L_{1,d} \leq W_{1,d} \leq 8$.

4.6 Arbitrary Topologies

In this section, we discuss the RWA problem for $\mathbf{k}$-allowable traffic in an arbitrary topology. Let $L_{d,\mathbf{k}}$ and $W_{d,\mathbf{k}}$ denote the minimum number of wavelengths which, if provided in each fiber, can support $\mathbf{k}$-allowable traffic with full wavelength conversion at all nodes and without wavelength conversion respectively.

We shall describe two lower bounds on $L_{d,\mathbf{k}}$ and two upper bounds on $W_{d,\mathbf{k}}$. Since $L_{d,\mathbf{k}} \leq W_{d,\mathbf{k}}$, given a lower bound on $L_{d,\mathbf{k}}$ and an upper bound on $W_{d,\mathbf{k}}$, the actual value of $W_{d,\mathbf{k}}$ lies between the two bounds.

4.6.1 Lower Bound on $L_{d,\mathbf{k}}$: the Link Counting Bound

In section 3.5.1, we used the link counting bound argument from [Pan92] to derive a lower bound on the required number of wavelengths for l-uniform traffic with full wavelength conversion at all nodes. For $\mathbf{k}$-allowable traffic, we can also use the link counting bound argument to derive a lower bound on $L_{d,\mathbf{k}}$. More specifically, given the traffic, let H be the sum of the number of hops traversed by each of the sessions under shortest path routing, and F be the number of fibers in the network. Then some fiber must support at least $\lceil H/F \rceil$ wavelengths, and thus $L_{d,\mathbf{k}} \geq \lceil H/F \rceil$.

However, the difficulty in applying the link counting bound for $\mathbf{k}$-allowable traffic has to do with finding the traffic matrix which yields the tightest bound. For static l-uniform traffic in section 3.5, this difficulty does not exist. In what follows, we shall refer to a traffic matrix which yields the tightest bound on $L_{d,\mathbf{k}}$ as a *limiting traffic matrix.*

The link counting bound is reasonably tight when, given a limiting traffic matrix, there exists a routing scheme which distributes traffic evenly on all the fibers. For example, consider supporting symmetric 1-allowable traffic, i.e. $\mathbf{k}$-allowable traffic in which all the k_i's are equal to 1, in the N-node bidirectional ring. One limiting traffic matrix is such that each node transmits a wavelength to the node $N/2$ hops away in the clockwise (CW) ring direction for N even, and $(N-1)/2$ hops away for N odd. In this example, the link counting bound is

$$L_{d,\mathbf{k}} \;\geq\; \begin{cases} \left\lceil \frac{N(N/2)}{2N} \right\rceil = \left\lceil \frac{N}{4} \right\rceil, & N \text{ even}, \\ \left\lceil \frac{N(N-1)/2}{2N} \right\rceil = \left\lceil \frac{N-1}{4} \right\rceil, & N \text{ odd}. \end{cases}$$

As mentioned in section 4.3.2, we know from [CM02] that $L_{d,\mathbf{k}} \leq \lceil N/4 \rceil + 1$ for any N (even or odd). Thus, the link counting bound is quite tight in this case.

As an example in which the link counting bound is not tight, consider symmetric 1-allowable traffic in the N-node binary tree topology in example 3, where $N = 2^n$ for some positive integer n. From theorem 6, we know that $L_{d,\mathbf{k}} = N/2$. For the link counting bound, one limiting traffic matrix is such that each node sends a wavelength to another node on the opposite side of the binary tree. Under this traffic, it is straightforward to show that $H = 2Nn$. Since $F = 4(N-1)$, it follows that $L_{d,\mathbf{k}} \geq \lceil Nn/(2(N-1)) \rceil$, which is approximately $n/2$ for large N. Since $L_{d,\mathbf{k}} = N/2$, the link counting bound is not tight. Notice that there is a bottleneck link in the N-node binary tree topology. Therefore, it is not possible to distribute traffic evenly on all the fibers.

4.6.2 Lower Bound on $L_{d,\mathbf{k}}$: the Cut Set Bound

In section 3.5.2, we used the cut set bound argument from [BB97] to derive a lower bound on the required number of wavelengths for l-uniform traffic with full wavelength conversion at all nodes. For $\mathbf{k}$-allowable traffic, we can also use the cut set bound argument to derive a lower bound on $L_{d,\mathbf{k}}$ in the similar fashion as described next.

Consider a cut set $\mathcal{C}$ which separates the end nodes into two sets $\mathcal{N}_{\mathcal{C},1}$ and $\mathcal{N}_{\mathcal{C},2}$. The amount of traffic (in wavelengths) across this cut from $\mathcal{N}_{\mathcal{C},1}$ to $\mathcal{N}_{\mathcal{C},2}$ can be up to $\min(|\mathcal{N}_{\mathcal{C},1}|, |\mathcal{N}_{\mathcal{C},2}|)$. Since there are $|\mathcal{C}|$ fibers from $\mathcal{N}_{\mathcal{C},1}$ to $\mathcal{N}_{\mathcal{C},2}$, one fiber across this cut must support up to $\lceil \min(|\mathcal{N}_{\mathcal{C},1}|, |\mathcal{N}_{\mathcal{C},2}|)/|\mathcal{C}| \rceil$ wavelengths, i.e. $L_{d,\mathbf{k}} \geq \lceil \min(|\mathcal{N}_{\mathcal{C},1}|, |\mathcal{N}_{\mathcal{C},2}|)/|\mathcal{C}| \rceil$. To tighten the bound, we search for the cut which yields the maximum lower bound, i.e.

$$L_{d,\mathbf{k}} \;\geq\; \max_{\mathcal{C}} \left\lceil \frac{\min(|\mathcal{N}_{\mathcal{C},1}|, |\mathcal{N}_{\mathcal{C},2}|)}{|\mathcal{C}|} \right\rceil .$$

Notice that, in section 4.3, we used the cut set bound to define the value of w^* in (4.1). In a tree topology, the cut set is a single link, and the bottleneck link yields the cut set bound. From corollary 1, we know that the cut set bound is tight for a tree topology.

We now give an example in which the cut set bound is not tight. Consider symmetric 1-allowable traffic in the 5-node bidirectional ring. Any cut set which separates the 5-node ring into two connected subnetworks with two and three nodes respectively yields the cut set bound $L_{d,\mathbf{k}} \geq \lceil \min(2,3)/2 \rceil = 1$. However, we argue below that $L_{d,\mathbf{k}} \geq 2$.

Consider the symmetric 1-allowable traffic matrix in which each node sends one wavelength to the node two hops away in the CW ring direction. It is easy to see that one CW directed wavelength can support at most two sessions, whereas one counterclockwise (CCW) directed wavelength can support at most one session. Since there are in total five sessions, we need more than one wavelength to support the given traffic. Thus, $L_{d,\mathbf{k}} \geq 2$.

Although the cut-set bound is not tight in the above example, we have not found an example in which the cut-set bound is not tight in the asymptotic sense, i.e. the difference between the bound and the actual value of $L_{d,\mathbf{k}}$ grows larger as the network size increases. The tightness of the cut-set bound remains to be investigated further.

4.6.3 Upper Bound on $W_{d,\mathbf{k}}$: the Embedded Tree Bound

In this subsection, we shall return to the wavelength assignment (WA) problem for $\mathbf{k}$-allowable traffic in an arbitrary tree topology considered in section 4.2 and relax the assumption that only leaf nodes are end nodes. This relaxation allows us to embed a tree topology in an arbitrary connected topology, as we have done for l-uniform traffic in section 3.5.3. The on-line tree WA algorithm can then be used to derive an upper bound on $W_{d,\mathbf{k}}$. As a specific example, figure 4-21a shows an arbitrary topology. One possible embedded tree is shown in figure 4-21b. Nodes 2, 4, and 5 are non-leaf end nodes.

Given an embedded tree topology with non-leaf end nodes, we can create the associated generic tree topology with no non-leaf end node as follows. For each non-leaf end node, create a new leaf node attached to it with the same value of k_i. The new leaf node is an end node, while the existing non-leaf node is no longer an end node. For example, figure 4-21c shows the generic tree topology associated with the embedded tree topology in figure 4-21b. In particular, there are three new leaf

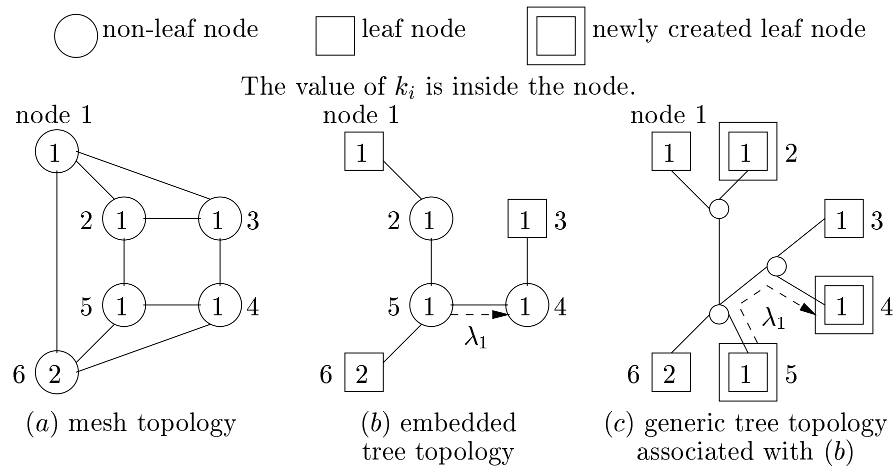


Figure 4-21: Embedded tree topology and its associated generic tree topology.

nodes in figure 4-21c created from the three non-leaf end nodes in figure 4-21b.

We now argue that the minimum number of wavelengths for **k**-allowable traffic for the generic tree, denoted by $W_{d,\mathbf{k},g}$, is at least the minimum number of wavelengths for the embedded tree, denoted by $W_{d,\mathbf{k},e}$. To see this, observe that, for the same traffic matrix, the WA for the generic tree can be used for the embedded tree as described next. Each lightpath in the generic tree can be mapped to an identical lightpath in the embedded tree except for all the newly created links in the generic tree. For example, the three-hop lightpath on wavelength λ_1 from leaf node 5 to leaf node 4 in figure 4-21c is mapped to the one-hop lightpath on λ_1 from node 5 to node 4 in figure 4-21b. It follows that $W_{d,\mathbf{k},e} \leq W_{d,\mathbf{k},g}$. We state this relationship formally as a lemma below.

Lemma 16 *For* **k***-allowable traffic, the wavelength requirements for an embedded tree and for its associated generic tree are related by* $W_{d,\mathbf{k},e} \leq W_{d,\mathbf{k},g}$.

Figure 4-22 shows an example scenario in which $W_{d,\mathbf{k},e} < W_{d,\mathbf{k},g}$. In the embedded tree shown in figure 4-22a, there are two leaf nodes with $k_1 = k_2 = 1$ and one non-leaf end node with $k_3 = 2$. By inspection, we see that at most one wavelength is used in each fiber. Thus, $W_{d,\mathbf{k},e} = 1$. In the associated generic tree shown in figure 4-22b, there are three leaf nodes with $k_1 = k_2 = 1$ and $k_3 = 2$. From theorem 6, we know that $W_{d,\mathbf{k},g} = 2$. Thus, $W_{d,\mathbf{k},e} < W_{d,\mathbf{k},g}$.

The following theorem indicates the scenarios in which $W_{d,\mathbf{k},e} = W_{d,\mathbf{k},g}$.

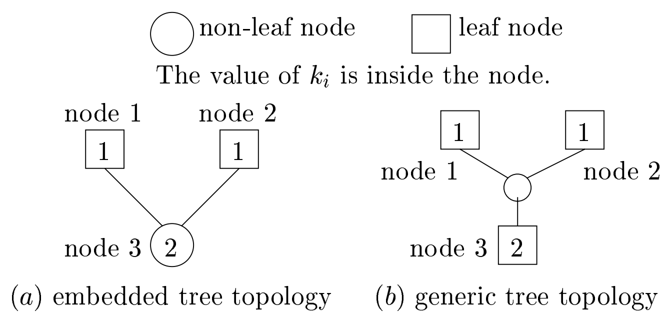


Figure 4-22: Example scenario in which $W_{d,\mathbf{k},e} < W_{d,\mathbf{k},g}$.

Theorem 9 *Let $W_{d,\mathbf{k},g} = w^*$. If the value of k_i for each non-leaf end node in the embedded tree topology is less than w^*, then $W_{d,\mathbf{k},e} = W_{d,\mathbf{k},g} = w^*$.*

Proof: Suppose that the value of k_i for each non-leaf end node in the embedded tree is less than w^*. From the definition of w^* for a generic tree given in (4.1), we see that the bottleneck link e^* in the generic tree is not one of the newly created links as compared with the embedded tree, or else w^* would be smaller than what it is. Thus, this bottleneck link e^* exists in the embedded tree and up to w^* wavelengths of traffic can traverse across it in one direction. It follows that $W_{d,\mathbf{k},e} \geq w^*$. Since $W_{d,\mathbf{k},e} \leq W_{d,\mathbf{k},g}$ (from lemma 16) and $W_{d,\mathbf{k},g} = w^*$, $W_{d,\mathbf{k},e} = W_{d,\mathbf{k},g} = w^*$. □

In a sufficiently large tree in which no end node has a significantly large value of k_i, we expect k_i for each non-leaf node i to be less than w^*, and thus $W_{d,\mathbf{k},e} = W_{d,\mathbf{k},g} = w^*$. Consequently, in most arbitrary topologies of interest, we expect to be able to embed a tree topology whose generic tree has the same wavelength requirement as the embedded tree. In these scenarios, the on-line tree WA algorithm can be used to obtain the WA for the generic tree which is then mapped to the WA for the embedded tree using the same number of wavelengths. The value of $W_{d,\mathbf{k},e}$ can then be used as an upper bound on $W_{d,\mathbf{k}}$. We summarize the discussion below as a corollary to theorem 9.

Corollary 4 *Let $W_{d,\mathbf{k},g} = w^*$. If the value of k_i for each non-leaf end node in the embedded tree topology is less than w^*, then the generic tree can be used to obtain the embedded tree bound in place of the embedded tree, i.e. $W_{d,\mathbf{k}} \leq W_{d,\mathbf{k},g} = W_{d,\mathbf{k},e}$.*

For example, in figure 4-21c, the value of w^* is equal to 2. Since the k_i's for all the non-leaf node are less than 2, corollary 4 tells us that, for the topology given in figure 4-21a, the bound from the embedded tree in figure 4-21b is $W_{d,\mathbf{k}} \leq W_{d,\mathbf{k},g} = W_{d,\mathbf{k},e} = 2$.

The embedded tree bound is a reasonable estimate on $W_{d,\mathbf{k}}$ when the network nodes are sparsely connected. However, for a densely connected network, it can perform poorly. For example, consider symmetric 1-allowable traffic in an $R \times R$ torus topology. We know from theorem 8 that $W_{d,\mathbf{k}} \leq \lceil R/2 \rceil$. From statement 2 of lemma 8, the generic tree associated with any embedded tree with R^2 end nodes has $w^* \geq R^2/d^*$, where d^* is the degree of the bottleneck node. Since d^* in the generic tree is at most 5, it follows that $W_{d,\mathbf{k},g} \geq R^2/5$. From theorem 9, $W_{d,\mathbf{k},e} = W_{d,\mathbf{k},g}$ in this example. Thus, the embedded tree bound $W_{d,\mathbf{k},e}$ is at least $R^2/5$. Since $W_{d,\mathbf{k}} = \lceil R/2 \rceil$, the embedded tree bound is not tight in this example.

4.6.4 Upper Bound on $W_{d,\mathbf{k}}$ in term of $L_{d,\mathbf{k}}$: the Graph Coloring Bound

In section 3.5.4, we used the graph coloring bound argument from [Agg+96] to derive an upper bound on the required number of wavelengths for l-uniform traffic with no wavelength conversion. For $\mathbf{k}$-allowable traffic, we can also use the graph coloring bound argument to derive an upper bound on $W_{d,\mathbf{k}}$ in a similar fashion as described next.

Given a routing assignment for all the sessions, i.e. the routes of all the lightpaths, for all $\mathbf{k}$-allowable traffic matrices such that the maximum load in a fiber is $L_{d,\mathbf{k}}$ wavelengths, we derive an upper bound on $W_{d,\mathbf{k}}$ by keeping the same routing assignment and performing wavelength assignment (WA).

As mentioned in section 3.5.4, the WA assignment problem can be reduced to the graph coloring problem in the path graph. For a given traffic and the routes of all the lightpaths, we create the corresponding path graph as follows. Each lightpath is mapped one-to-one to a node in the path graph. Two nodes in the path graph are connected if and only if the two corresponding lightpaths share a fiber. Let h be the length of the longest lightpath over all traffic matrices. Then, for any given traffic, each lightpath shares a fiber with at most $h(L_{d,\mathbf{k}} - 1)$ other lightpaths. It follows that the maximum node degree in the path graph is $h(L_{d,\mathbf{k}} - 1)$. Therefore, $h(L_{d,\mathbf{k}} - 1) + 1$ wavelengths are sufficient to color the path graph associated with any given traffic matrix, i.e.

$$W_{d,\mathbf{k}} \quad \leq \quad h(L_{d,\mathbf{k}} - 1) + 1.$$

Unfortunately, the graph coloring bound can be quite pessimistic. For example, consider symmetric 1-allowable traffic in the N-node bidirectional ring with N even. From section 4.3, we know that $W_{d,\mathbf{k}} = \lceil N/3 \rceil$ for $N \geq 7$. We also know from [CM02] that $\lceil N/4 \rceil \leq L_{d,\mathbf{k}} \leq \lceil N/4 \rceil + 1$. For the length (in hops) of the longest lightpath, it is clear that $h = N/2$. Therefore, the graph coloring bound is approximately $\frac{N}{2} \left\lceil \frac{N}{4} \right\rceil \approx N^2/8$. Since $W_{d,\mathbf{k}} = \lceil N/3 \rceil$, the graph coloring bound is not tight in this example.

Chapter 5

Band/Wavelength RWA Problem

We now motivate studying the band/wavelength RWA problem. The goal is to understand when and how individual wavelengths should be aggregated into bands of wavelengths for optical switching in order to reduce the cost of optical switches. We shall present some preliminary results and point out directions for future research.

Recall that an optical switch is subjected to a crossbar constraint, which dictates that no more than one input (output) can be connected to a single output (input). An optical switch can direct traffic sessions from each input fiber to its designated output fiber. For current optical switches in practice, there is no significant difference whether each input fiber carries traffic sessions on one wavelength or multiple wavelengths. In the previous chapters, each optical switch in a switching node acts as a wavelength selective switch, i.e. each input fiber contains one wavelength of traffic. In this chapter, we allow each input fiber to an optical switch to carry a band of multiple wavelengths. Accordingly, we shall refer to an optical switch used in this fashion as a *band switch*. In addition, we shall refer to switching of traffic optically in band of wavelengths (instead of in wavelengths) simply as *band switching*.

For convenience, throughout the chapter, an optical switch refers to a reconfigurable optical switch. When we discuss a fixed optical switch, we shall specify explicitly.

5.1 Reduction in Optical Switches through Band Switching

In this section, we use a simple scenario to show how band switching can reduce the number of optical switches required in the network.

Consider the N-node star topology with symmetric k-allowable traffic, i.e. $\mathbf{k}$-allowable traffic in which all the k_i's are equal to k. Assume that each end node is connected to optical switches at the hub node. In addition, we assume that k is significantly greater than N. Under this assumption, each node is likely to send several wavelengths to each of its destinations, and band switching of traffic among the N nodes is attractive. On the other hand, if k is smaller than N, each node is likely to send only a small number of wavelengths to each of its destinations, and would utilize only a small fraction in each band under band switching. In this case, band switching is not attractive.

More specifically, consider symmetric 6-allowable traffic in the 3-node star topology as shown in figure 5-1. For clarity, we consider each end node as one distinct source node and one distinct destination node. Without band switching, theorem 5 in section 4.1 tells us that six wavelengths are required to support the traffic. Consequently, we need six units of 3×3 optical switches at the hub node, as shown in figure 5-1a.

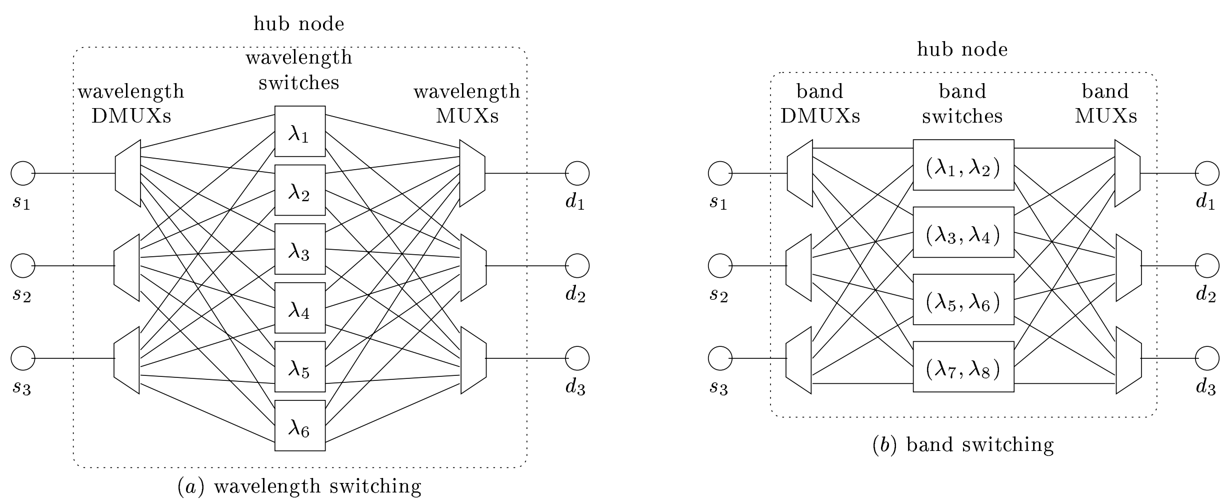


Figure 5-1: The 3-node star topology in which each end node is connected to optical switches at the hub.

Consider band switching with the band size of two wavelengths. As will be shown shortly, we can support symmetric 6-allowable traffic using eight wavelengths and four units of 3×3 optical switches, as illustrated in figure 5-1b. We shall derive general expressions for the required number of band switches and the required number of wavelengths below. With current technology, there is no significant cost difference between a 3×3 optical switch for band switching and a 3×3 optical switch for wavelength switching. Roughly speaking, in this example, band switching saves two optical switches at the expense of two additional wavelengths.

Let b denote the band size in wavelengths. For symmetric k-allowable traffic in the N-node star network in which each end node is connected to optical switches at the hub, let $B(N,k,b)$ denote the required number of optical switches at the hub, and $W(N,k,b)$ denote the required number of wavelengths in a fiber with no wavelength conversion.

If each node pair communicates in units of bands (instead of wavelengths), the traffic can be viewed as symmetric k'-allowable traffic with a band as a traffic unit and $k' \approx k/b$. From theorem 5 in section 4.1, we know that $B(N,k,b) \approx k/b$. Since $W(N,k,b) = bB(N,k,b)$, it follows that $W(N,k,b) \approx k$.

However, when the end nodes transmit in units of wavelengths, there may be some transmission bands which are underutilized. In this case, we need more than k/b transmission bands, and thus more than k/b optical switches at the hub and more than k wavelengths.

For example, consider a scenario in which node 1 transmits one wavelength to nodes 2, ..., $N-1$ and $k-(N-2)$ wavelengths to node N. The number of optical switches required at the hub to support the traffic to nodes 2, ..., $N-1$ is $N-2$. In addition, the number of optical switches required at the hub to support the traffic to node N is $\approx (k-(N-2))/b$. Thus, in total, we need at least $\approx (N-2)+(k-(N-2))/b = k/b + (1-1/b)(N-2)$ switches.

The following theorem provides exact expressions for $B(N,k,b)$ and $W(N,k,b)$ for $k \geq N-1$.

Theorem 10 *For symmetric k-allowable traffic in the N-node star network in which each end node is connected to optical switches at the hub, if $k \geq N-1$, then $B(N,k,b)$ and $W(N,k,b)$ are given by*

$$\begin{aligned} B(N,k,b) &= (N-1) + \left\lfloor \frac{k-(N-1)}{b} \right\rfloor, \\ W(N,k,b) &= b(N-1) + b\left\lfloor \frac{k-(N-1)}{b} \right\rfloor. \end{aligned}$$

Proof: Consider a particular end node, say node 1. Let B_1 be the minimum number of transmit bands from node 1 required to support symmetric k-allowable traffic. We first show that $B_1 \geq (N-1) + \left\lfloor \frac{k-(N-1)}{b} \right\rfloor$. Consider the traffic in which node 1 sends one wavelength to nodes 2, 3, ..., $N-1$ and $k-(N-2)$ wavelengths to node N. In this case, from node 1, $N-2$ bands are required to support the traffic to nodes 2, 3, ..., $N-1$, and $\left\lceil \frac{k-(N-2)}{b} \right\rceil$ bands are required to support the traffic to node N. The total number of bands is the lower bound on B_1 given below.

$$B_1 \quad \geq \quad (N-2) + \left\lceil \frac{k-(N-2)}{b} \right\rceil \quad = \quad (N-1) + \left\lfloor \frac{k-(N-1)}{b} \right\rfloor$$

The last equality can be justified as follows. If $\frac{k-(N-2)}{b}$ is an integer, then $\left\lceil \frac{k-(N-2)}{b} \right\rceil = \frac{k-(N-2)}{b} = \left\lfloor \frac{k-(N-1)}{b} \right\rfloor + 1$. If $\frac{k-(N-1)}{b}$ is an integer, then $\left\lceil \frac{k-(N-2)}{b} \right\rceil = \frac{k-(N-1)}{b} + 1 = \left\lfloor \frac{k-(N-1)}{b} \right\rfloor + 1$. In all the other cases, we have $\left\lceil \frac{k-(N-2)}{b} \right\rceil = \left\lceil \frac{k-(N-1)}{b} \right\rceil = \left\lfloor \frac{k-(N-1)}{b} \right\rfloor + 1$.

We next show that $B_1 \leq (N-1) + \left\lfloor \frac{k-(N-1)}{b} \right\rfloor$. Let $\mathcal{N}'$ denote the set of destination nodes to which node 1 transmits partially utilized bands, and $N' = |\mathcal{N}'|$. For each node i in $\mathcal{N}'$, let q_i' denote the number of utilized wavelengths in the partially utilized bands from node 1.

To support the traffic from node 1, we can use N' bands for $\sum_{i \in \mathcal{N}'} q_i'$ wavelengths and additional $(k - \sum_{i \in \mathcal{N}'} q_i')/b$ fully utilized bands. Thus we have that $B_1 \leq N' + (k - \sum_{i \in \mathcal{N}'} q_i')/b$. It remains to show that this upper bound is at most $(N-1) + \left\lfloor \frac{k-(N-1)}{b} \right\rfloor$, i.e.

$$N' + \frac{k - \sum_{i \in \mathcal{N}'} q_i'}{b} \quad \leq \quad (N-1) + \left\lfloor \frac{k-(N-1)}{b} \right\rfloor .$$

If $N' = N-1$, i.e. there are partially utilized bands to all the $N-1$ destinations from node 1, then $q_i' \geq 1$ for every node i in $\mathcal{N}'$, and thus $\sum_{i \in \mathcal{N}'} q_i' \geq N' = N-1$. Since $(k - \sum_{i \in \mathcal{N}'} q_i')/b$ is an integer, $(k - \sum_{i \in \mathcal{N}'} q_i')/b = \lfloor (k - \sum_{i \in \mathcal{N}'} q_i')/b \rfloor$. It follows that

$$\begin{aligned} N' + \frac{k - \sum_{i \in \mathcal{N}'} q_i'}{b} \quad &= \quad N' + \left\lfloor \frac{k - \sum_{i \in \mathcal{N}'} q_i'}{b} \right\rfloor \\ &= \quad (N-1) + \left\lfloor \frac{k - \sum_{i \in \mathcal{N}'} q_i'}{b} \right\rfloor \quad \leq \quad (N-1) + \left\lfloor \frac{k-(N-1)}{b} \right\rfloor . \end{aligned}$$

On the other hand, if $N' < N-1$, using the fact that $\sum_{i \in \mathcal{N}'} q_i' \geq N'$ and $N' \leq N-2$, we can bound $N' + (k - \sum_{i \in \mathcal{N}'} q_i')/b$ as follows.

$$N' + \frac{k - \sum_{i \in \mathcal{N}'} q_i'}{b} \quad \leq \quad N' + \frac{k - N'}{b} \quad \leq \quad \left(1 - \frac{1}{b}\right)(N-2) + \frac{k}{b}$$

$$= \quad (N-1) + \frac{k}{b} - \frac{N-1}{b} - 1 + \frac{1}{b} \quad \leq \quad (N-1) + \left\lfloor \frac{k-(N-1)}{b} \right\rfloor + \frac{1}{b}$$

Since $b > 1$ and $N' + (k - \sum_{i \in \mathcal{N}'} q'_i)/b$ is an integer, it follows that

$$N' + \frac{k - \sum_{i \in \mathcal{N}'} q'_i}{b} \quad \leq \quad (N-1) + \left\lfloor \frac{k-(N-1)}{b} \right\rfloor.$$

Therefore, we have shown that $B_1 = (N-1) + \left\lfloor \frac{k-(N-1)}{b} \right\rfloor$. Since the above choice of node 1 is arbitrary, we conclude that the minimum number of transmit bands required at each node is $(N-1) + \left\lfloor \frac{k-(N-1)}{b} \right\rfloor$. By similar arguments, the minimum number of receive bands required at each node is $(N-1) + \left\lfloor \frac{k-(N-1)}{b} \right\rfloor$.

What we have here is a traffic scenario in which each node transmits up to $(N-1) + \left\lfloor \frac{k-(N-1)}{b} \right\rfloor$ bands and receives up to $(N-1) + \left\lfloor \frac{k-(N-1)}{b} \right\rfloor$ bands. This traffic is similar to symmetric k'-allowable traffic with $k' = (N-1) + \left\lfloor \frac{k-(N-1)}{b} \right\rfloor$. The difference is that, in this case, the traffic unit is a band instead of a wavelength. It follows from theorem 5 in section 4.1 that $(N-1) + \left\lfloor \frac{k-(N-1)}{b} \right\rfloor$ is the minimum number of bands required to support this traffic. Consequently, $B(N,k,b) = (N-1) + \left\lfloor \frac{k-(N-1)}{b} \right\rfloor$.

Since each band contains b wavelengths, it follows that $W(N,k,b) = bB(N,k,b) = b(N-1) + b\left\lfloor \frac{k-(N-1)}{b} \right\rfloor$. □

Consider again the example in figure 5-1 where $N = 3$, $k = 6$, and $b = 2$. Theorem 10 tells us that the minimum number of switches required at the hub is $B(3,6,2) = 2 + \lfloor (6-2)/2 \rfloor = 4$. In addition, the required number of wavelengths is $W(3,6,2) = 4 + 2\lfloor (6-2)/2 \rfloor = 8$.

It is worth noting that $B(N,k,b) \leq k$, i.e. band switching never requires more optical switches than wavelength switching. To see this, we use the assumption that $k \geq N-1$ to obtain the last inequality below.

$$\begin{aligned} B(N,k,b) \quad &= \quad (N-1) + \left\lfloor \frac{k-(N-1)}{b} \right\rfloor \quad \leq \quad \left(1 - \frac{1}{b}\right)(N-1) + \frac{k}{b} \\ &\leq \quad \left(1 - \frac{1}{b}\right) k + \frac{k}{b} \quad = \quad k. \end{aligned}$$

For fixed values of N and k, $B(N,k,b)$ as given in theorem 10 is a decreasing function of b. As we increase the band size (in wavelengths), we expect to use fewer optical switches in the network. However, the price to pay is the increase in the required number of wavelengths $W(N,k,b)$. With an appropriate choice of b, we expect the decrease in optical switches to outweigh the increase in wavelengths. We shall investigate this trade-off in more details in the next section.

5.2 Trade-Off between Optical Switches and Wavelengths

In this section, we study the trade-off between the decrease in optical switches and the increase in the number of wavelengths as a result of band switching. Consider again the N-node star topology in which each end node transmits (receives) traffic to (from) optical switches at the hub. Under symmetric k-allowable traffic, where $k > N-1$, theorem 10 indicates the required number of optical switches $B(N,k,b)$ and the required number of wavelengths $W(N,k,b)$.

In what follows, we shall make an approximation by ignoring integer rounding in the expressions of $B(N,k,b)$ and $W(N,k,b)$. This approximation allows us to see the trade-off between $B(N,k,b)$ and $W(N,k,b)$ more clearly. In particular, ignoring integer rounding, $B(N,k,b)$ and $W(N,k,b)$ are given by

$$\begin{aligned} B(N,k,b) &\approx (N-1) + \frac{k-(N-1)}{b}, \\ W(N,k,b) &\approx b(N-1) + k - (N-1). \end{aligned}$$

From the above expressions, it is clear that, for fixed values of N and k, $B(N,k,b)$ decreases with b, whereas $W(N,k,b)$ increases with b. Roughly speaking, the larger band size decreases the number of optical switches at the expense of more wavelengths.

At this point, it is natural to ask what band size b minimizes the system cost due to optical switches and transmission wavelengths. The cost structure of optical equipment is rapidly changing and is beyond the scope of this thesis. However, to illustrate the cost trade-off, we use a simple linear cost structure below.

Let c_1 and c_2 be the linear cost coefficients for optical switches and transmission wavelengths respectively. Assume that the system cost due to optical switches and transmission wavelengths, denoted by $c(b)$, can be expressed as

$$\begin{aligned} c(b) &= c_1 B(N,k,b) + c_2 W(N,k,b) \\ &\approx c_1 \left[(N-1) + \frac{k-(N-1)}{b} \right] + c_2 \left[b(N-1) + k - (N-1) \right]. \end{aligned}$$

Figure 5-2 shows the graphs of optical switching cost and transmission cost as a function of the band size b. To minimize the cost $c(b)$, we solve for the optimal band size, denoted by b^*, which is the solution to the equation

$$0 = \frac{df_C}{db}(b) = -\frac{c_1}{b^2}(k-(N-1)) + c_2(N-1).$$

The corresponding solution is $b^* = \sqrt{\frac{c_1(k-(N-1))}{c_2(N-1)}}$.

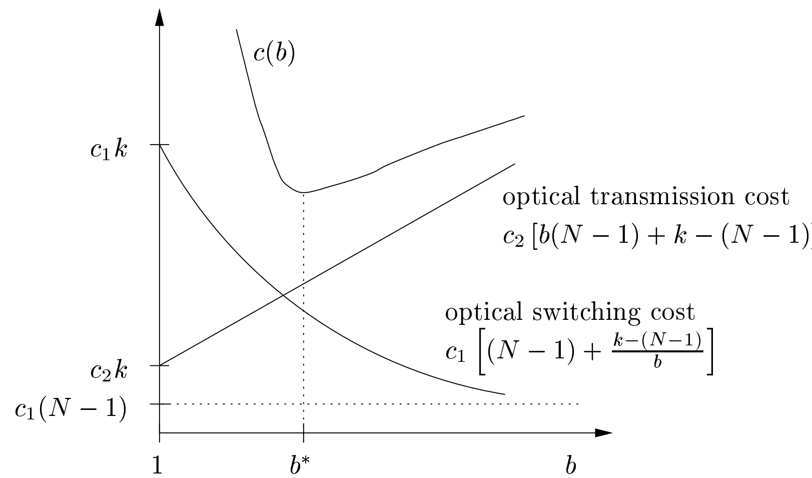


Figure 5-2: Cost trade-off between optical switches and transmission wavelengths.

As expected, when the cost coefficient c_1 for optical switches is low compared to the cost coefficient c_2 for transmission wavelengths, the expression for b^* suggests us to use a small band size. On the other hand, when c_1 is high compared to c_2, a large band size is more attractive.

5.3 Alternative Network Architecture for Band Switching

In this section, we present an alternative network architecture which can further reduce the number of optical switches while using approximately the same number of wavelengths under band switching. As in sections 5.1 and 5.2, consider the N-node star topology with symmetric k-allowable traffic, where $k > N-1$. However, in this section, we allow the use of fixed optical switches at the hub. Given the band size of b wavelengths, let $B(N,k,b)$ and $W(N,k,b)$ denote the minimum numbers of band switches and wavelengths required to support the traffic.

We begin with an informal discussion in which we ignore integer rounding. As previously mentioned in section 5.1, if each node pair communicates in units of bands (instead of wavelengths), then $B(N,k,b) \approx k/b$ and $W(N,k,b) \approx k$. However, when the end nodes communicate in units of wavelengths, there may be some transmission bands which are underutilized. In this case, theorem 10 tells us that $B(N,k,b) \approx (N-1) + (k-(N-1))/b = k/b + (1-1/b)(N-1)$ and $W(N,k,b) \approx k + (b-1)(N-1)$. The excess amount of resources, i.e. $(1-1/b)(N-1)$ optical switches and $(b-1)(N-1)$ wavelengths, can be viewed as the cost penalty due to underutilization of the bands.

We now present an alternative network architecture which no longer needs the excess number of optical switches while using approximately the same number of wavelengths. The main idea is based on the observation that we need up to $k/b + (1-1/b)(N-1)$ bands to handle the scenarios in which each node distributes its traffic to all the other nodes. This observation motivates us to provide a dedicated band connection between each node pair. Notice that this scheme is similar to providing dedicated resources for 1-uniform traffic, but with a band (instead of a wavelength) as a traffic unit. On top of dedicated provision of resources, we provide some optical switches at the hub node as before. We shall support the traffic by first using the dedicated bands, and then the shared bands (through optical switches) if necessary. In what follows, we shall refer to this alternative architecture as the *semi-reconfigurable architecture*.

As an example, consider the same scenario given in figure 5-1, i.e. $N = 3$, $k = 6$, and $b = 2$. Figure 5-3 shows the corresponding semi-reconfigurable architecture. Note that the semi-reconfigurable architecture uses two optical switches instead of four, and still use eight wavelengths as before. We ignore the cost of fixed optical switches which are usually much less expensive than reconfigurable optical switches. The following theorem provides general expressions for the required number of optical switches $B(N,k,b)$ and the required number of wavelengths $W(N,k,b)$ in the semi-reconfigurable architecture.

Theorem 11 *For symmetric k-allowable traffic in the N-node star network with the semi-reconfigurable architecture, if $k \geq N-1$, then $B(N,k,b)$ and $W(N,k,b)$ are given by*

$$\begin{aligned} B(N,k,b) &= \left\lceil \frac{k}{b} \right\rceil - 1, \\ W(N,k,b) &= b\left[(N-1) + \left\lceil \frac{k}{b} \right\rceil - 1\right]. \end{aligned}$$

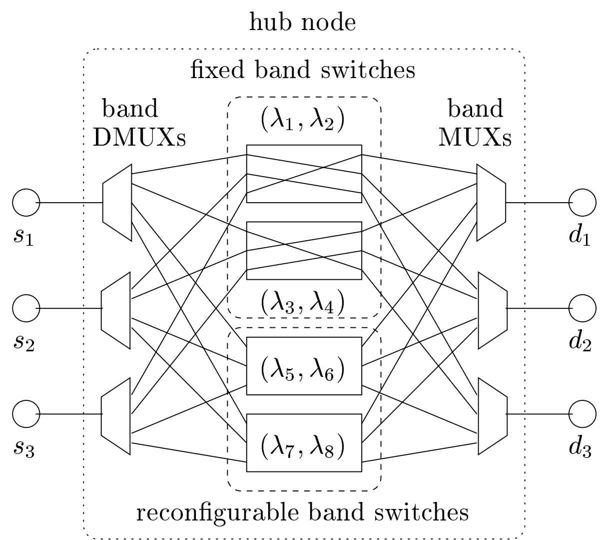


Figure 5-3: The semi-reconfigurable architecture for the 3-node star topology with symmetric 6-allowable traffic.

Proof: To show that $B(N,k,b) \geq \lceil k/b \rceil - 1$, consider the case in which node 1 transmits k wavelengths to node 2. Given a band as a traffic unit, node 1 needs to transmit $\lceil k/b \rceil$ bands to node 2. Since there is one dedicated band connection from node 1 to node 2, the number of bands required to go through optical switches is $\lceil k/b \rceil - 1$. Thus, $B(N,k,b) \geq \lceil k/b \rceil - 1$.

To show that $B(N,k,b) \leq \lceil k/b \rceil - 1$, consider the traffic transmitted from node 1. Let $\tilde{k}$ be the total number of wavelengths utilized in the dedicated bands. If a shared band through an optical switch is needed, then it is necessarily true that $\tilde{k} > b$. Thus, from node 1, a sufficient number of bands is

$$\frac{k-\tilde{k}}{b} \quad \leq \quad \frac{k-b}{b} \quad \leq \quad \left\lceil \frac{k}{b} \right\rceil - 1.$$

By the same argument, a sufficient number of bands to node 1 is $\lceil k/b \rceil - 1$. Since the choice of node 1 is arbitrary, it follows that the traffic through optical switches can be viewed as symmetric k'-allowable traffic with a band as a basic traffic unit and $k' = \lceil k/b \rceil - 1$. From theorem 5 in section 4.1, $\lceil k/b \rceil - 1$ bands are sufficient, i.e. $B(N,k,b) \leq \lceil k/b \rceil - 1$.

Thus, we have shown that $B(N,k,b) = \lceil k/b \rceil - 1$. Apart from $B(N,k,b)$ bands which go through optical switches, each node has $N-1$ dedicated band connections with all the other nodes. Therefore, $W(N,k,b) = b[(N-1) + B(N,k,b)] = b[(N-1) + \lceil k/b \rceil - 1]$. $\square$

As in section 5.2, we now study the cost trade-off between the decrease in optical switches and the increase in the number of wavelengths. We shall make an approximation by ignoring integer rounding in the expressions of $B(N,k,b)$ and $W(N,k,b)$, as shown below.

$$\begin{aligned} B(N,k,b) &\approx k/b - 1 \\ W(N,k,b) &\approx b(N-1) + k - b \end{aligned}$$

In comparison to the basic architecture in sections 5.1 and 5.2 where $B(N,k,b) \approx k/b + (1 - 1/b)(N-1)$ and $W(N,k,b) \approx b(N-1)+k-(N-1)$, we see that $B(N,k,b)$ for the semi-reconfigurable architecture is smaller while $W(N,k,b)$ is approximately the same provided that the product Nb dominates the terms N and b.

To find an appropriate value of the band size b (in wavelengths), let c_1 and c_2 be the linear cost coefficients for optical switches and transmission wavelengths respectively. Assume that the system cost due to optical switches and transmission wavelengths, denoted by $c(b)$, can be expressed as

$$\begin{aligned} c(b) &= c_1 B(N,k,b) + c_2 W(N,k,b) \\ &\approx c_1 \left[\frac{k}{b} - 1\right] + c_2 \left[b(N-1) + k - b\right]. \end{aligned}$$

Figure 5-4 shows the graphs of optical switching cost and transmission cost as a function of the band size b. To minimize the cost $c(b)$, we solve for the optimal band size, denoted by b^*, which is the solution to the equation

$$0 = \frac{df_C}{db}(b) = -\frac{c_1}{b^2}k + c_2(N-2).$$

The corresponding solution is $b^* = \sqrt{\frac{c_1 k}{c_2(N-2)}}$.

As in section 5.2, when the cost coefficient c_1 for optical switches is low compared to the cost coefficient c_2 for transmission wavelengths, a small band size is attractive. On the other hand, when c_1 is high compared to c_2, a large band size is attractive.

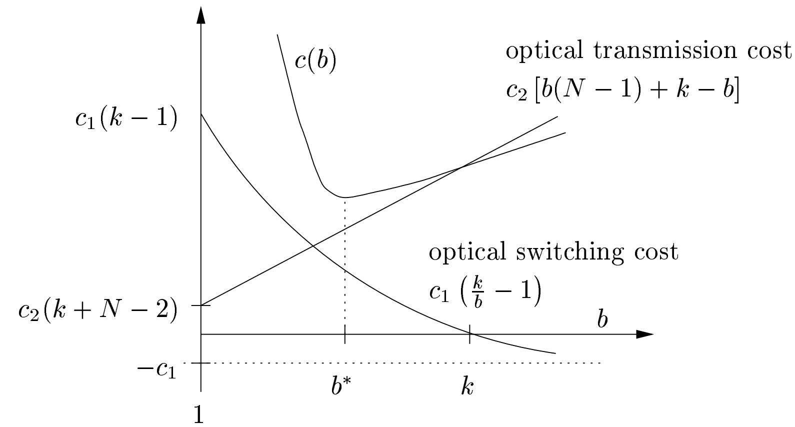


Figure 5-4: Cost trade-off between optical switches and transmission wavelengths for the semi-reconfigurable architecture.

5.4 Traffic Aggregation for Band Switching

In this section, we consider the use of band switching in a network with N end nodes and symmetric k-allowable traffic with $k < N-1$. When $k < N-1$, using a band instead of a wavelength as a traffic unit is not attractive since each node may transmit to k different destinations and would utilize only one wavelength in each band it transmits.

To overcome low utilization of transmission bands, one possible strategy is to use a two-level hierarchical architecture as described next. Some nodes in the network serve as aggregation nodes. Each end node is connected to one aggregation node or more. The aggregation nodes switch traffic to and from its connected end nodes in wavelength units. However, the aggregation nodes switch traffic among themselves using only band switching. As an example, figure 5-5 shows a two-level 9-node star network with three aggregation nodes. The aggregation nodes switch traffic in wavelengths, while the central hub node switches traffic in bands.

Consider symmetric 2-allowable traffic among the end nodes. The traffic among the three aggregation nodes can be viewed as symmetric 6-allowable traffic. Notice that, if we view the aggregation nodes as end nodes, we have the same scenario as the example in section 5.3, i.e. symmetric 6-allowable traffic among three end nodes. Thus, the semi-reconfigurable architecture in figure 5-3 can be used at the central hub. Figure 5-6 shows the detailed architecture for the two-level 9-node star. For simplicity, we route all the traffic through the central hub, even though

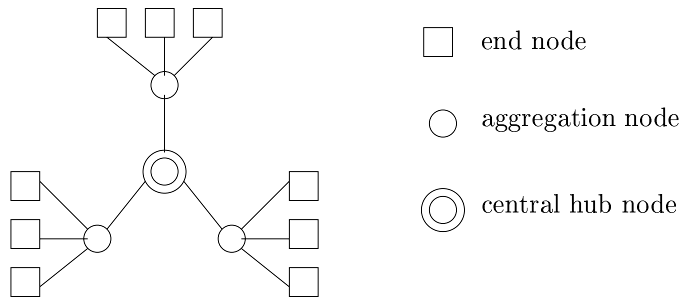


Figure 5-5: Two-level architecture with traffic aggregation for band-switching.

some sessions are among the end nodes connected to the same aggregation node.

In the two-level architecture shown in figure 5-6, we use in total 24 units of 3×1 optical switches, 24 units of 1×3 optical switches, 2 units of 3×3 optical switches, and 8 wavelengths. If we do not use band switching, the architecture similar to figure 5-1a in section 5.1 requires 2 units of 9×9 optical switches and 2 wavelengths. In this example, it may look inefficient to use band switching. However, as the number of end nodes N gets large, the cost of an $N \times N$ switch may get prohibitively high. In that case, the two-level architecture may become attractive since it requires optical switches with smaller numbers of ports.

The cost comparison between different architectures is beyond the scope of this thesis. However, investigation in this area should play an important role for the choice of network architecture. We leave this topic for future research.

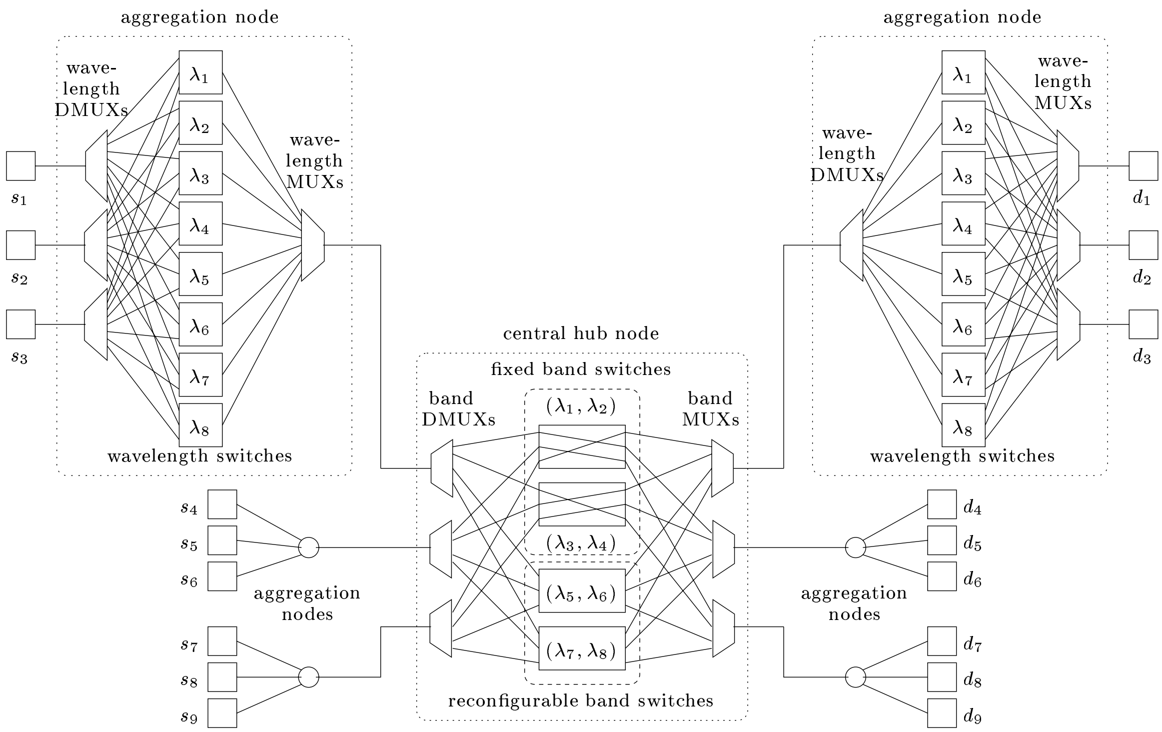


Figure 5-6: Detailed architecture for the network in figure 5-5.

Chapter 6

Conclusion and Directions for Future Research

We considered the design of an all-optical wavelength division multiplexing (WDM) core network connecting multiple local networks through electronic switches at the access nodes. In the core network, we expect that traffic can be separated into two components. In the first component, each session (between a pair of access nodes) is an aggregate of a large number of small-rate end-to-end sessions. Each individual end-to-end session is not necessarily static but, through statistical averaging, an aggregate of individual sessions is approximately static. We support traffic sessions of this type by static provisioning of routes and wavelengths. In the second traffic component, each session (between a pair of access nodes) cannot be well approximated as static due to insufficient aggregation. Thus, we support traffic sessions of this type by dynamic provisioning of routes and wavelengths.

The wavelengths used for dynamic provisioning need to be equipped with reconfigurable components including reconfigurable optical switches and tunable transmitters/receivers. To reduce the network costs, the wavelengths used for static provisioning can be equipped with non-reconfigurable components such as fixed optical switches and non-tunable transmitters/receivers.

We studied routing and wavelength assignment (RWA) problems for both static and dynamic traffic with no wavelength conversion. More specifically, for static traffic, we studied how to provide l dedicated wavelength paths between each pair of access nodes, i.e. l-uniform traffic, for basic all-to-all connectivity. Our goal is to develop off-line RWA algorithms which use the minimum number of wavelengths to support l-uniform traffic. We described the existing literature for the bidirectional ring and two-dimensional (2D) torus topologies, and developed off-line RWA algorithms for arbitrary tree and binary hypercube topologies. We observed that, as the network topology gets more densely connected, i.e. the number of fibers per node increases, the required number of wavelengths decreases.

Let $L_{s,l}$ and $W_{s,l}$ denote the minimum number of wavelengths which, if provided in each fiber, can support l-uniform traffic with full wavelength conversion at all nodes and without wavelength conversion respectively. Interestingly, in all the topologies for which we were able to obtain closed form expressions for $W_{s,l}$ and $L_{s,l}$, we found that $W_{s,l} = L_{s,l}$, i.e. wavelength converters cannot decrease the wavelength requirement for l-uniform traffic.

For arbitrary network topologies, we discussed several known bounds on $W_{s,l}$ and $L_{s,l}$, and introduced an upper bound on $W_{s,l}$ based on embedding a tree topology in any given arbitrary topology. We observed that the cut set bound yields the exact value of $L_{s,l}$ in several arbitrary topologies. Whether or not the cut set bound yields the exact value of $L_{s,l}$ in any arbitrary topology remains an open problem. In addition, we suspect that the equation $W_{s,l} = L_{s,l}$ is also valid for any arbitrary topology. Whether or not $W_{s,l} = L_{s,l}$ for any arbitrary topology is another open problem for static RWA.

To study dynamic RWA, we adopted the nonblocking formulation. We assume that the basic traffic unit is a wavelength, and the traffic matrix changes from time to time but always belongs to the **k**-allowable traffic set defined based on the numbers of fully tunable transmitters and fully tunable receivers at the access nodes. In addition, we assume that a transition from one traffic matrix to another is a result of either a single session arrival or a single session departure. Our goal is to design on-line RWA algorithms which can support all the **k**-allowable traffic matrices in a rearrangeably nonblocking fashion while using the minimum number of wavelengths and incurring few rearrangements of existing lightpaths, if any, for each new session request.

We provided on-line RWA algorithms for arbitrary tree, bidirectional ring, and 2D torus topologies, and described the existing literature on binary hypercube topologies. We observed from our on-line RWA algorithms that the number of lightpath rearrangements per new session request is closely related to the number of lightpaths supported on a single wavelength. Roughly speaking, a higher amount of wavelength reuse incurs a greater number of lightpath rearrangements. In all cases, we observed that the number of lightpath rearrangements depends on the topological properties, e.g. network size, but not on the actual size of the traffic $\mathbf{k}$ as we increase $\mathbf{k}$ by some integer factor.

Let $L_{d,\mathbf{k}}$ and $W_{d,\mathbf{k}}$ denote the minimum number of wavelengths which, if provided in each fiber, can support $\mathbf{k}$-allowable traffic with full wavelength conversion at all nodes and without wavelength conversion respectively. For arbitrary network topologies, we discussed several known bounds on $W_{d,\mathbf{k}}$ and $L_{d,\mathbf{k}}$, and introduced an upper bound on $W_{d,\mathbf{k}}$ based on embedding a tree topology in any given arbitrary topology. There exist examples which show that none of those bounds are tight. Developing good bounds on $W_{d,\mathbf{k}}$ is an interesting topic for future research. Also interesting is the design of on-line RWA algorithms for arbitrary topologies in which we can derive bounds on the number of lightpath rearrangements per new session request.

Unlike the case of static l-uniform traffic, the use of wavelength converters can reduce the wavelength requirement for dynamic traffic. For example, it is known that, for symmetric k-allowable traffic in an N-node bidirectional ring topology, $L_{d,\mathbf{k}} < W_{d,\mathbf{k}}$ for a sufficiently large N. Therefore, the investigation of how wavelength converters can be used efficiently is another interesting topic for future research in dynamic RWA.

Having developed off-line and on-line RWA algorithms for several specific network topologies, we hope that our analytical approaches and techniques can be used in the development of similar RWA algorithms for a wider class of network topologies in the future.

Finally, we began exploring the band/wavelength RWA problem in which we switch traffic in bands instead of individual wavelengths. Our goal is to understand when and how individual wavelengths should be aggregated into bands of wavelengths to reduce the cost of optical switching. We considered symmetric k-allowable traffic in the N-node star topology. For k significantly greater than N, we argued that band switching is attractive, and demonstrated the trade-off between the number of optical switches and the number of wavelengths as a function of the band size (in wavelengths). For k smaller than $N-1$, we presented a two-level architecture. In the lower level, the aggregation nodes switch traffic to and from the end nodes using wavelength switches. In the higher level, the aggregation nodes exchange traffic among themselves using only band switches at the central hub. The cost comparisons among different choices of network architecture remain to be investigated in the future.

Appendix A

Efficient Bipartite Matchings with Maximum Node Degree 2

In this section, we provide an efficient algorithm for partitioning the edges in a bipartite graph $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$ with $|\mathcal{V}_1| = |\mathcal{V}_2| = V$ and maximum node degree 2 into two disjoint matchings. As pointed out in section 4.1, the general algorithm for bipartite matching in [CLR90] can be used for our task with the running time $O(V^2)$.[1] Our algorithm performs the same task with the running time $O(V)$.

Assume for now that each node in $\mathcal{V}_1$ has degree 2. The assumption implies that each node in $\mathcal{V}_2$ has degree 2. To see this, assume some node in $\mathcal{V}_2$ has degree less than 2. Since there are $2V$ edges incident on V nodes in $\mathcal{V}_2$, there must exist a node in $\mathcal{V}_2$ with degree greater than 2, contradicting the assumption of maximum node degree 2.

The main idea of our algorithm is as follows. In a bipartite graph with node degree 2, the edges in $\mathcal{E}$ form a set of disjoint cycles each of which contains an even number of edges. For example, figure A-1 shows three disjoint cycles in a bipartite graph with node degree 2.

For each cycle, we move along the edges of the cycle and alternately assign them to two matchings, denoted by $\mathcal{M}_1$ and $\mathcal{M}_2$, such that no two adjacent edges belong to the same matching. Note that this is possible since there are even number of edges in each cycle. Finally, we collect the edges in all disjoint cycles to form two matchings $\mathcal{M}_1$ and $\mathcal{M}_2$, as illustrated in figure A-2. We describe

[1] By running time $O(g(n))$, we mean the running time can be expressed as a function $f(n)$ of the problem size n such that there exist a positive real constant c and a positive integer n_0 satisfying $0 \leq f(n) \leq cg(n)$ for all $n \geq n_0$.

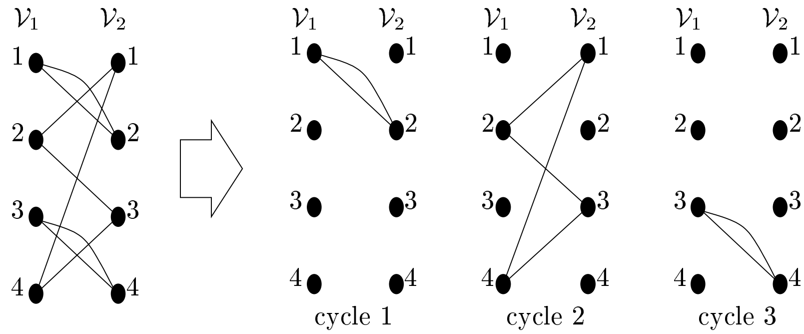
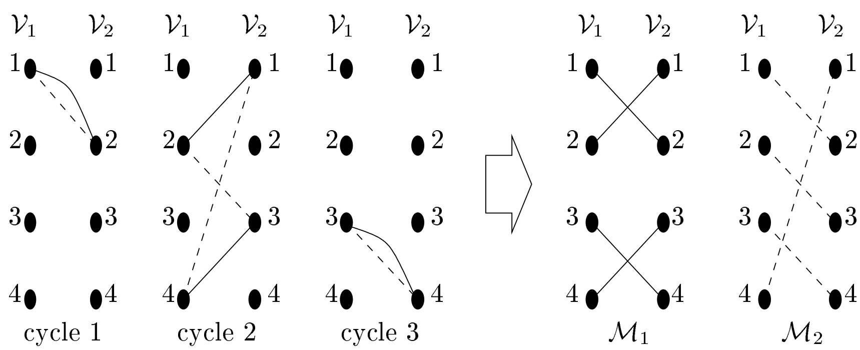


Figure A-1: Cycles in a bipartite graph with node degree 2.



Solid (dashed) edges belong to $\mathcal{M}_1$ ($\mathcal{M}_2$).

Figure A-2: Assignment of edges to matchings $\mathcal{M}_1$ and $\mathcal{M}_2$.

the algorithm formally below. We shall refer to this algorithm as the *degree-2 bipartite matching algorithm*.

Algorithm 6 (Degree-2 Bipartite Matching Algorithm) Given a bipartite graph $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E})$ with $|\mathcal{V}_1| = |\mathcal{V}_2| = V$ and node degree 2, form two matchings $\mathcal{M}_1$ and $\mathcal{M}_2$ as follows.

Step 1: Form a new cycle disjoint from all the previous cycles starting from the lowest-index node in $\mathcal{V}_1$ with an incident edge not yet assigned to either $\mathcal{M}_1$ or $\mathcal{M}_2$. Assign the edges in this cycle alternately to $\mathcal{M}_1$ and $\mathcal{M}_2$ such that no two adjacent edges in the cycle are assigned to the same matching.

Step 2: Look for the next lowest-index node in $\mathcal{V}_1$ with an incident edge not yet assigned to either $\mathcal{M}_1$ or $\mathcal{M}_2$. If such a node exists, proceed to step 1. If such a node does not exist, terminate the algorithm.

Proof of algorithm correctness: We first argue that each iteration of step 1 terminates with a new cycle. Note that, except for the starting node in $\mathcal{V}_1$, step 1 arrives at any other node on one of its incident edges and leaves on the other. Thus, when it terminates, step 1 must terminate at the starting node and form a new cycle. Since the number of nodes is finite, step 1 cannot keep visiting new nodes forever and has to terminate.

We next show that each cycle has an even number of edges. To see this, choose a node in $\mathcal{V}_1$ as the starting node for the cycle. If we move along the cycle by an odd number of edges, we end up in $\mathcal{V}_2$. On the other hand, if we move along the cycle by an even number of edges, we end up in $\mathcal{V}_1$. Thus, when we end up at the starting node in $\mathcal{V}_1$, we must have traversed an even number of edges.

We now show that each node belongs to exactly one cycle. Consider a given node v. Since each cycle containing v has two edges incident on v, node v which has degree 2 cannot belong to two or more disjoint cycles. To argue that v must belong to some cycle, we proceed by contradiction. Assume that v does not belong to any cycle. Since step 2 cannot terminate if v is in $\mathcal{V}_1$, it follows that node v must be in $\mathcal{V}_2$. However, the existence of such a node v in $\mathcal{V}_2$ implies that there is a node in $\mathcal{V}_1$ connected to v by an edge not yet assigned to either $\mathcal{M}_1$ or $\mathcal{M}_2$. This contradicts the terminating condition in step 2.

Since each node belongs to exactly one cycle, all the edges in the bipartite graph are assigned to $\mathcal{M}_1$ and $\mathcal{M}_2$. In addition, since there are even number of edges in each cycle, the algorithm successfully assigns adjacent edges in the same cycle to two different matchings. It follows that no two edges in $\mathcal{M}_1$ are incident on the same node. We conclude that $\mathcal{M}_1$ is indeed a matching. Similar arguments show that $\mathcal{M}_2$ is indeed a matching. $\square$

Since the degree-2 bipartite matching algorithm visits each node in the bipartite graph exactly once, it follows that the running time of the algorithm is $O(V)$.

We now relax the assumption that each node in $\mathcal{V}_1$ has degree 2. If there is a node in $\mathcal{V}_1$ with degree less than 2, we can add extra edges to the bipartite graph to make all the nodes in $\mathcal{V}_1$ and $\mathcal{V}_2$ have degree 2 as follows. Label the nodes in $\mathcal{V}_1$ and in $\mathcal{V}_2$ from 1 to V. Find the lowest-index node in $\mathcal{V}_1$ with degree less than 2. Add an edge from this node to the lowest-index node in $\mathcal{V}_2$ with degree less than 2. Repeat the process until all the nodes in $\mathcal{V}_1$ have degree 2. When all the nodes in $\mathcal{V}_1$ have degree 2, all the nodes in $\mathcal{V}_2$ also have degree 2. After using the degree-2 bipartite matching algorithm to partition the edges into two disjoint matchings, we can remove the extra edges to get the two desired matchings. Since adding and removing the extra edges can be done with the running time $O(V)$, the overall algorithm has the running time $O(V)$.

Appendix B

$W_{d,k}$ for Bidirectional Rings

In this section, we derive the general expression for $W_{d,k}$, the minimum number of wavelengths in a fiber required to support symmetric k-allowable traffic ($\mathbf{k}$-allowable traffic in which all the k_i's are equal to k) without wavelength conversion, for the bidirectional ring topology with $N \geq 3$ nodes.[1] More specifically, we shall prove that

$$W_{d,k} = \begin{cases} \lceil 3k/4 \rceil, & N = 3, \\ k, & N = 4, \\ \lceil 5k/3 \rceil, & N = 5, 6, \\ \lceil Nk/3 \rceil, & N \geq 7. \end{cases}$$

In section 4.3, we prove that $W_{d,k} = \lceil kN/3 \rceil$ for $N \geq 7$. The same proof can be used to show that $\lceil kN/3 \rceil$ for $N = 5$. It remains to justify the above expression of $W_{d,k}$ for $N = 3, 4, 6$. We shall consider each value of N separately below. In each case, we make use of a known lemma in [Pan92] which is a direct consequence of lemma 6

Lemma 17 [Pan92] *A symmetric k-allowable traffic matrix can be partitioned into k symmetric 1-allowable traffic matrices.*

Proof: Given a symmetric k-allowable traffic matrix, we can construct the traffic bipartite graph as defined in section 4.1. Each node in the bipartite graph has degree at most k. From lemma 6, we

[1] The RWA problem for the 2-node ring is trivial. It is obvious that $W_{d,k} = \lceil k/2 \rceil$ for $N = 2$.

can partition the set of edges into k disjoint bipartite matchings. Since each matching corresponds to a symmetric 1-allowable traffic matrix, the lemma statement follows. □

Throughout this section, we shall assume that, whenever we partition a given symmetric k-allowable traffic matrix into k symmetric 1-allowable traffic matrices, each 1-allowable traffic matrix is *maximal* in the sense that we cannot add an extra session to the traffic matrix (except perhaps for self-traffic which we do not consider). When the assumption does not hold, we can add extra sessions to make the traffic matrix maximal, solve the RWA problem, and then remove the extra sessions. It is easy to see that, in any maximal traffic matrix, the sessions form a set of cycles.

B.1 Proof of $W_{d,k} = \lceil 3k/4 \rceil$ for $N = 3$

To derive a lower bound on $W_{d,k}$, consider the symmetric k-allowable traffic in which each node transmits k wavelengths to the node one hop away in the clockwise (CW) ring direction. A CW directed wavelength can support up to three sessions, while a counterclockwise (CCW) directed wavelength can support only one session. Thus, each wavelength can support up to four sessions. Since there are in total $3k$ sessions, it follows that $W_{d,k} \geq \lceil 3k/4 \rceil$.

It remains to prove that $W_{d,k} \leq \lceil 3k/4 \rceil$. Assume there are $\lceil 3k/4 \rceil$ wavelengths in a fiber. Partition any given symmetric k-allowable traffic matrix into k symmetric 1-allowable traffic matrices. In each 1-allowable traffic matrix (assumed to be maximal), the sessions form either a 2-cycle or a 3-cycle.[2] Each 2-cycle is a mutual adjacent session pair, i.e. the source (destination) of one session is the destination (source) of the other session, and can be supported on one directed wavelength in any ring direction. By inspection, it is easy to see that each 3-cycle can be supported on one directed wavelength in some ring direction, as illustrated in figure B-1. In particular, there are only two possible scenarios for a 3-cycle in the 3-node ring. The 3-cycle can be supported on one CW directed wavelength in one scenario, and on one CCW directed wavelength in the other. Therefore, each 1-allowable traffic matrix can be supported on one directed wavelength.

We can support $\lceil 3k/4 \rceil$ symmetric 1-allowable traffic matrices on $\lceil 3k/4 \rceil$ directed wavelengths in the required ring directions. This is possible since there are $\lceil 3k/4 \rceil$ wavelengths available. Having done so, there are $k - \lceil 3k/4 \rceil$ remaining 1-allowable traffic matrices. These matrices contain at most

[2] An m-cycle is a cycle which contains m sessions.

The two possible scenarios for a 3-cycle in the 3-node ring.

Figure B-1: Supporting a 3-cycle on one directed wavelength in the 3-node ring.

$3(k - \lceil 3k/4 \rceil)$ sessions each of which we support on one directed wavelength in any ring direction. The total number of directed wavelengths required is $\lceil 3k/4 \rceil + 3(k - \lceil 3k/4 \rceil) \leq 2\lceil 3k/4 \rceil$. It follows that $\lceil 3k/4 \rceil$ wavelengths are sufficient.

B.2 Proof of $W_{d,k} = k$ for $N = 4$

To derive a lower bound on $W_{d,k}$, consider the symmetric k-allowable traffic matrix in which each node transmits k wavelengths to the node two hops away in the CW direction. A directed wavelength in any ring direction can support up to two sessions. Thus, each wavelength can support up to four sessions. Since there are in total $4k$ sessions, it follows that $W_{d,k} \geq \lceil 4k/4 \rceil = k$.

It remains to prove that $W_{d,k} \leq k$. Assume there are k wavelengths in a fiber. Partition any given symmetric k-allowable traffic matrix into k symmetric 1-allowable traffic matrices. We claim and shall prove below that each 1-allowable traffic matrix can be supported on one wavelength. It follows that k wavelengths are sufficient to support k symmetric 1-allowable traffic matrices, and thus the original traffic matrix. Therefore, $W_{d,k} \leq k$.

We now prove the claim that a symmetric 1-allowable traffic matrix can be supported on one wavelength. We consider three possible scenarios for the set of cycles in a 1-allowable traffic matrix (assumed to be maximal).

1. Two 2-cycles: We can support one 2-cycle on a CW directed wavelength and the other on a CCW directed wavelength. Thus, one wavelength is sufficient.
2. One 3-cycle: Ignoring the node which neither transmits nor receives traffic, we see from figure B-1 that a 3-cycle can be supported on one directed wavelength in some ring direction. Thus, one wavelength is sufficient.
3. One 4-cycle: If there exists an adjacent session triplet which can be supported on one directed

wavelength, support the session triplet on one directed wavelength in the required ring direction.[3] We then support the remaining session on one directed wavelength in the opposite ring direction. Thus, one wavelength is sufficient.

Otherwise, i.e. no such session triplet exists, we form two adjacent session pairs from the 4-cycle. The two session pairs can be supported on two directed wavelengths in the required ring directions. We now show that the two required ring directions cannot be the same, and thus one wavelength is sufficient.

We proceed by contradiction. Let s_1, s_2, s_3, and s_4 denote the four contiguous sessions in the 4-cycle. Moreover, (s_1, s_2) and (s_3, s_4) are the two adjacent session pairs. Let x_1, x_2, x_3, and x_4 denote their path lengths (in hops) in the CW direction, and $X = x_1 + x_2 + x_3 + x_4$. Suppose each session pair can be supported on a CW directed wavelength, but not on a CCW directed wavelength. Since (s_1, s_2) can be supported on a CW directed wavelength but do not form a 2-cycle, $x_1 + x_2 < 4$. Similarly, $x_3 + x_4 < 4$. Thus, $X < 8$. Since (s_1, s_2, s_3) is not a session triplet which can be supported on one CW directed wavelength, $x_1 + x_2 + x_3 > 4$. Thus, $X > 4$. The inequalities $4 < X < 8$ contradict the fact that the sum of path lengths in any cycle in the 4-node ring must be an integer multiple of 4.

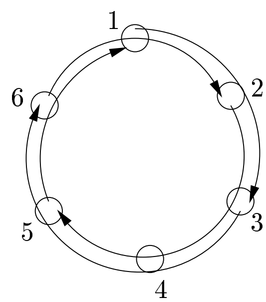
We conclude that the two adjacent session pairs cannot require two CW directed wavelengths. Reversing the roles of CW and CCW directions in the above arguments, we see that they cannot require two CCW directed wavelengths. It follows that one CW directed wavelength and one CCW directed wavelength, i.e. one wavelength, are sufficient.

B.3 Proof of $W_{d,k} = \lceil 5k/3 \rceil$ for $N = 6$

To derive a lower bound on $W_{d,k}$, consider the symmetric k-allowable traffic matrix which can be partitioned into k symmetric 1-allowable traffic matrices each of which contains the sessions shown in figure B-2. In particular, there are five sessions: (1,3), (3,6), (6,2), (2,5), and (5,1).

By inspection, a CW directed wavelength can support up to two sessions, while a CCW directed wavelength can support only one session. Thus, each wavelength can support up to three sessions.

[3] An adjacent session triplet is a set of three sessions s_1, s_2, and s_3 such that the destination of s_1 is the source of s_2, and the destination of s_2 is the source of s_3.



The sessions are (1,3), (3,6), (6,2), (2,5), and (5,1).

A CW directed wavelength can can support at most two sessions.

A CCW directed wavelength can can support only one session.

Figure B-2: Symmetric 1-allowable traffic for the lower bound of $W_{d,k}$ for $N = 6$.

Since there are in total $5k$ sessions, it follows that $W_{d,k} \geq \lceil 5k/3 \rceil$.

It remains to prove that $W_{d,k} \leq \lceil 5k/3 \rceil$. Assume that there are $\lceil 5k/3 \rceil$ wavelengths in a fiber. Partition any given symmetric k-allowable traffic matrix into k symmetric 1-allowable traffic matrices. We claim and shall prove later that each symmetric 1-allowable traffic matrix can either be supported on two CW directed wavelengths and one CCW directed wavelength, or on one CW directed wavelength and two CCW directed wavelengths. Let p be the number of 1-allowable traffic matrices which can be supported on two CW directed wavelengths and one CCW directed wavelength. These p matrices can be supported on $2p$ CW directed wavelengths and p CCW directed wavelengths. The other $k-p$ matrices can be supported on $k-p$ CW directed wavelengths and $2(k-p)$ CCW directed wavelengths. Thus, in total, we can support the given symmetric k-allowable traffic matrix on $k+p$ CW directed wavelengths and $2k-p$ CCW directed wavelengths. We consider three cases below.

- Case 1: $k+p \geq \lceil 5k/3 \rceil$. Support $\lceil 5k/3 \rceil$ session pairs in the CW direction. This is possible since there are $\lceil 5k/3 \rceil$ wavelengths available. Having done so, there are $k+p-\lceil 5k/3 \rceil$ remaining session pairs which can share a CW directed wavelength. However, we support these pairs without sharing using $2(k+p-\lceil 5k/3 \rceil)$ CCW directed wavelengths. In addition, there are $2k-p$ session pairs which can share a CCW directed wavelength. Thus, the number of CCW directed wavelengths required is

  $$2(k+p-\lceil 5k/3 \rceil)+2k-p \quad \leq \quad 2k/3+p \quad \leq \quad \lceil 5k/3 \rceil.$$

  Thus, $\lceil 5k/3 \rceil$ wavelengths are sufficient.

- Case 2: $2k-p \geq \lceil 5k/3 \rceil$. This case is similar to case 1. By reversing the roles of CW and

CCW directions, we obtain the same conclusion that $\lceil 5k/3 \rceil$ wavelengths are sufficient.

- Case 3: $k + p \leq \lceil 5k/3 \rceil$ and $2k - p \leq \lceil 5k/3 \rceil$. In this case, it is clear that $\lceil 5k/3 \rceil$ directed wavelength in each ring direction, i.e. $\lceil 5k/3 \rceil$ wavelengths, are sufficient.

We now prove the claim that each symmetric 1-allowable traffic matrix can either be supported on two CW directed wavelengths and one CCW directed wavelength, or on one CW directed wavelength and two CCW directed wavelengths. We consider six possible scenarios for the set of cycles in a 1-allowable traffic matrix (assumed to be maximal).

1. Three 2-cycles: We can support two 2-cycles on two CW directed wavelengths and the other on one CCW directed wavelength. Thus, the claim follows.

2. One 2-cycle and one 3-cycle: Consider the 3-cycle. By ignoring the nodes which neither transmit nor receive traffic, we see from figure B-1 that the 3-cycle can be supported on one directed wavelength in some ring direction. For the 2-cycle, we can support it on one directed wavelength in any ring direction. The claim then follows.

3. One 2-cycle and one 4-cycle: Consider the 4-cycle. By ignoring the nodes which neither transmit nor receive traffic, we see from the previous section that one wavelength is sufficient to support the 4-cycle. For the 2-cycle, we can support it on one directed wavelength in any ring direction. The claim then follows.

4. Two 3-cycles: From the above argument, each 3-cycle can be supported on one directed wavelength in some ring direction. Thus, the claim follows.

5. One 5-cycle: We can form two adjacent session pairs and support them on two directed wavelengths in the required ring directions. Since the remaining session can be supported on one directed wavelength in any ring direction, the claim follows.

6. One 6-cycle: If there exists an adjacent session triplet which can be supported on one directed wavelength, support the session triplet on one directed wavelength in the required ring direction. Then form another adjacent session pair and support it on one directed wavelength in the required ring direction. Since the remaining session can be supported on one directed wavelength in any ring direction, the claim follows.

   Otherwise, i.e. no such session triplet exists, we form three adjacent session pairs from the 6-cycle. The three session pairs can be supported on three directed wavelengths in the required ring directions. We show below that the three required ring directions cannot be the same, and thus the claim is valid.

We proceed by contradiction. Let s_1, s_2, s_3, s_4, s_5, and s_6 denote the six contiguous sessions in the 6-cycle. Moreover, (s_1, s_2), (s_3, s_4), and (s_5, s_6) are the three adjacent session pairs. Let x_1, x_2, x_3, x_4, x_5, and x_6 denote their path lengths (in hops) in the CW direction, and $X = x_1 + x_2 + x_3 + x_4 + x_5 + x_6$. Suppose each session pair can be supported on a CW directed wavelength, but not on a CCW directed wavelength. Since (s_1, s_2) can be supported on a CW directed wavelength but do not form a 2-cycle, $x_1 + x_2 < 6$. Similarly, $x_3 + x_4 < 6$ and $x_5 + x_6 < 6$. Thus, $X < 18$. Since (s_1, s_2, s_3) is not a session triplet which can be supported on one CW directed wavelength, $x_1 + x_2 + x_3 > 6$. Similarly, $x_4 + x_5 + x_6 > 6$. Thus, $X > 12$. The inequalities $12 < X < 18$ contradict the fact that the sum of path lengths in any cycle in the 6-node ring must be an integer multiple of 6.

We conclude that the three adjacent session pairs cannot require three CW directed wavelengths. Reversing the roles of CW and CCW directions in the above arguments, we see that they cannot require three CCW directed wavelengths.

Appendix C

On-Line Single-Hub Ring RWA Algorithm

In this section, we present the on-line single-hub ring RWA algorithm, as mentioned in section 4.3.1, as well as its correctness proof. In the algorithm below, we maintain two RWA conditions at all time: (i) only adjacent session pairs at the hub share a directed wavelength, and (ii) all mutual adjacent session pairs at the hub share a directed wavelength.

Algorithm 7 (On-Line Single-Hub Ring RWA Algorithm) (Use $\lceil (N-1)/2 \rceil$ wavelengths and perform at most four lightpath rearrangements per new session request.)

Session termination: When a session terminates, simply remove its associated lightpath from the ring without any further lightpath rearrangement.

Session arrival: When a session arrives and the resultant traffic matrix is still **k**-allowable, proceed as follows.

Step 1: If the new session, denoted by u, can form a mutual adjacent session pair at the hub with some existing session, denoted by x, there are two possibilities.

($1a$) If x is not sharing its directed wavelength, assign the mutual adjacent session pair u and x to share this directed wavelength. In this case, no lightpath rearrangement is required.

(1b) If x is sharing a directed wavelength with another existing session, denoted by y, then x and y are not mutually adjacent at the hub, or else u and x cannot be mutually adjacent at the hub. Remove y from its directed wavelength and assign the mutual adjacent session pair u and x to share the directed wavelength of y.

If there is a free directed wavelength, use it to support y. In this case, one lightpath rearrangement is made. Otherwise, we claim that y can form another adjacent session pair at the hub with some nonsharing session, denoted by z. Note that y and z cannot be mutually adjacent at the hub, or else they would have shared a directed wavelength.

If the directed wavelength of z can support y, assign y and z to share this directed wavelength. In this case, one lightpath rearrangement is made. Otherwise, we claim that there must exist either a nonsharing session or a mutual adjacent session pair in the opposite ring direction. In the case of a nonsharing session in the opposite ring direction, we remove that nonsharing session and support y and z on its directed wavelength. The removed nonsharing session can then be supported on the directed wavelength of z. In this case, a total of three lightpath rearrangements are made. In the case of a mutual adjacent session pair in the opposite ring direction, we remove that mutual adjacent session pair and support y and z on their directed wavelength. The removed mutual adjacent session pair can then be supported on the directed wavelength of z. In this case, a total of four lightpath rearrangements are made.

Step 2: If u cannot form a mutual adjacent session pair at the hub with any existing session and there is a free directed wavelength, use a free directed wavelength to support u. In this case, no lightpath rearrangement is made.

Step 3: If u cannot form a mutual adjacent session pair at the hub with any existing session and there is no free directed wavelength, we claim that, among nonsharing sessions and u, a nonmutual adjacent session pair at the hub can be formed. Denote this session pair by y and z. There are two possibilities.

(3a) If u is in the session pair, i.e. $y = u$ or $z = u$, assume without loss of generality that $y = u$. If the directed wavelength of z can support y, assign y and z to share this directed wavelength. In this case, no lightpath rearrangement is required. Otherwise, we claim there must exist either a nonsharing session or a mutual adjacent session pair in the opposite ring direction. In the case of a nonsharing session in the opposite ring direction, we remove that nonsharing session and support y and z on its directed wavelength. The removed nonsharing session can then be supported on the directed wavelength of z. In this case, a total of two lightpath rearrangements are made. In the case of a mutual adjacent session pair in the opposite ring direction, we remove that mutual adjacent session pair and support y and z on their directed wavelength. The removed mutual adjacent session pair can then be supported on the directed wavelength of z. In this case, a total of three lightpath rearrangements are made.

($3b$) If u is not in the session pair, then $y \neq u$ and $z \neq u$. If the directed wavelength of either y or z can support the session pair, assign y and z to share this directed wavelength. This sharing frees one directed wavelength on which u can be supported. In this case, one lightpath rearrangement is made. Otherwise, we claim that there must exist either a nonsharing session or a mutual adjacent session pair in the opposite ring direction. In the case of a nonsharing session in the opposite ring direction, we remove that nonsharing session and support y and z on its directed wavelength. The removed nonsharing session and the new session can then be supported on the directed wavelengths of y and z. In this case, a total of three lightpath rearrangements are made. In the case of a mutual adjacent session pair in the opposite ring direction, we remove that mutual adjacent session pair and support y and z on their directed wavelength. The removed mutual adjacent session pair and the new session can then be supported on the directed wavelengths of y and z. In this case, a total of four lightpath rearrangements are made.

Proof of algorithm correctness: From the algorithm description, it is clear that we always keep the two desired RWA conditions, i.e. (i) only adjacent sessions at the hub share a directed wavelength, and (ii) all mutual adjacent sessions at the hub share a directed wavelength. In addition, it is clear that at most four lightpath rearrangements are made to support each new session request. We shall prove the two claims in step 1, and the other three claims in step 3.

The first claim in step 1 and the first claim in step 3 are essentially the same. We shall prove the two claims at the same time. The claim states that if a session to be supported, denoted by w, is not mutually adjacent to any existing session at the hub and there is no free directed wavelength to support it, then there exists among nonsharing sessions and w an adjacent session pair at the hub, denoted by y and z.

We proceed by contradiction. Assume that an adjacent session pair at the hub cannot be found. Let p be the number of mutual adjacent session pairs at the hub. Let q be the number of nonmutual adjacent session pairs at the hub which share a directed wavelength. Let r be the number of nonsharing sessions including session w. We argue that $r \leq N-1-p-q$. To see this, define r_i^t and r_i^r, $1 \leq i \leq N$, to be the number of nonsharing sessions transmitted and received at node i respectively. Since there is no adjacent session pair at the hub (node 1) among these r sessions, we have that either $r_1^t = 0$ or $r_1^r = 0$. Without loss of generality, assume $r_1^t = 0$. Note that each of the $p+q$ sharing session pairs which are adjacent at the hub uses one transmitter at a nonhub node. There are in total $N-1$ transmitters at nonhub nodes. Thus, the number of transmitters used for nonsharing sessions at nonhub nodes are bounded by $\sum_{i=2}^{N} r_i^t \leq N-1-p-q$. It follows that

$$r \;=\; \sum_{i=1}^{N} r_i^t \;=\; r_1^t + \sum_{i=2}^{N} r_i^t \;\leq\; N-1-p-q.$$

Since we have a total of $2\lceil (N-1)/2 \rceil$ directed wavelengths, the number of directed wavelengths available to support nonsharing paths is $2\lceil (N-1)/2 \rceil - p - q$, which is at least the number of nonsharing paths $N-1-p-q$. This contradicts the assumption that there is no free directed wavelength to support w. Thus, we have shown that an adjacent session pair at the hub must exist.

The second claim in step 1 and the last two claims in step 3 are essentially the same. We shall prove them all at the same time. The claim states that if a nonmutual adjacent session pair at the hub, denoted by y and z, cannot fit on a directed wavelength of either y or z and there is no free directed wavelength in the opposite ring direction, then there exists either a nonsharing session or a mutual adjacent session pair on a directed wavelength in the opposite ring direction. As defined above, let p be the number of mutual adjacent session pairs at the hub. Let $\hat{q}$ be the number of nonmutual adjacent session pairs at the hub including sessions y and z. Note that each of these $\hat{q}$ session pairs may or may not share a directed wavelength. We first show that $\hat{q} \leq \lfloor (N-1)/2 \rfloor$. Define the following quantities for node i, $2 \leq i \leq N$. Let $\hat{q}_i^t$ and $\hat{q}_i^r$ denote the number of sessions in those $\hat{q}$ session pairs which are transmitted and received at node i respectively. It is clear that $\hat{q}_i^t \leq k_i$ and $\hat{q}_i^r \leq k_i$. By definition, each of these $\hat{q}$ session pairs is not a mutual adjacent session pair at the hub. Thus, at each nonhub node i, either $\hat{q}_i^t = 0$ or $\hat{q}_i^r = 0$. It follows that $\hat{q}_i^t + \hat{q}_i^r \leq k_i$. Because each of the $\hat{q}$ session pairs uses one transmitter and one receiver at nonhub nodes, it follows that

$$2\hat{q} \quad = \quad \sum_{i=2}^{N}(\hat{q}_i^t + \hat{q}_i^r) \quad \leq \quad \sum_{i=2}^{N} k_i \quad = \quad N - 1.$$

Since $\hat{q}$ is an integer, we have shown that $\hat{q} \leq \lfloor (N-1)/2 \rfloor$.

The claim is now apparent from the fact that $\hat{q} \leq \lfloor (N-1)/2 \rfloor$. In other words, the number of supported nonmutual adjacent session pairs at the hub $\hat{q} - 1$ is strictly less than the number of directed wavelengths in each ring direction $\lceil (N-1)/2 \rceil$. Given that there is no free directed wavelength, it follows that, in either ring direction, either a nonsharing session or a mutual adjacent session pair exists. $\square$

Bibliography

[Agg+96] A. Aggarwal *et al*, "Efficient routing in optical networks," *Journal of the ACM*, vol. 43, no. 6, pp. 973-1001, November 1996.

[AR95] Y. Aumann and Y. Rabani, "Improved bounds for all optical routing," *ACM-SIAM Symposium on Discrete Algorithms*, January 1995.

[BM96] D. Banerjee and B. Mukherjee, "A practical approach for routing and wavelength assignment in large wavelength-routed optical networks," *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 5, pp. 903-908, June 1996.

[BYC97] S. Banerjee, J. Yoo, and C. Chen, "Design of wavelength-routed optical networks for packet switched traffic," *Journal of Lightwave Technology*, vol. 15, no. 9, pp. 1636-1646, September 1997.

[BB97] S. Baroni and P. Bayvel, "Wavelength Requirements in Arbitrarily Connected Wavelength-Routed Optical Networks," *Journal of Lightwave Technology*, vol. 15, no. 2, pp. 242-251, February 1997.

[BH96] R.A. Barry and P.A. Humblet, "Models of blocking probability in all-optical networks with and without wavelength changers," *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 5, pp. 858-867, June 1996.

[Ber85] C. Berge, *Graphs.* North-Holland, 1985.

[BM00] R. Berry and E.H. Modiano, "Reducing electronic multiplexing costs in SONET/WDM rings with dynamically changing traffic," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 10, pp. 1961-1971, October 2000.

[Bir96] A. Birman, "Computing approximate blocking probabilities for a class of all-optical networks," *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 5, pp. 852-857, June 1996.

[CB96] C. Chen and S. Banerjee, "A new model for optimal routing and wavelength assignment in wavelength division multiplexed optical networks," in *Proceedings of IEEE INFOCOM*, pp. 164-171, April 1996.

[CM02] L. Chen and E.H. Modiano, "Efficient Routing and Wavelength Assignment for Reconfigurable WDM Networks with Wavelength Converters," to appear.

[CM00] A.L. Chiu and E.H. Modiano, "Traffic grooming algorithms for reducing electronic multiplexing costs in WDM ring networks," *Journal of Lightwave Technology*, vol. 18, no. 1, pp. 2-12, January 2000.

[CGK92] I. Chlamtac, A. Ganz, and G. Karmi, "Lightpath communications: an approach to high bandwidth optical WAN's," *IEEE Transactions on Communications*, vol. 40, no. 7, pp. 1171-1182, July 1992.

[CLR90] T.H. Cormen, C.E. Leiserson, and R.L. Rivest, *Introduction to Algorithms.* McGraw-Hill, 1990.

[Elr93] A. Elrefaie, "Multiwavelength survivable ring network architectures," in *Proceedings of IEEE ICC*, pp. 1245-1251, May 1993.

[GJ79] M.R. Garey and D.S. Johnson, *Computers and Intractability, A Guide to the Theory of NP-Completeness*, W.H. Freeman and Company, 1979.

[GRS00] O. Gerstel, R. Ramaswami, and G. Sasaki, "Cost-effective traffic grooming in WDM rings,"*IEEE/ACM Transactions on Networking*, vol. 8, no. 5, pp. 618-630, October 2000.

[Ger+99] O. Gerstel, G. Sasaki, S. Kutten, and R. Ramaswami, "Worst-case analysis of dynamic wavelength allocation in optical networks," *IEEE/ACM Transactions on Networking*, vol. 7, no. 6, pp. 833-845, December 1999.

[Gre93] P.E. Green, *Optical Networks.* Prentice Hall, 1993.

[LL96] K.C. Lee and V.O.K. Li, "A wavelength rerouting algorithm in wide-area all-optical networks," *Journal of Lightwave Technology*, vol. 14, no. 6, pp. 1218-1229, June 1996.

[LS99] L. Li and A.K. Somani, "Dynamic wavelength routing using congestion and neighborhood information," *IEEE/ACM Transactions on Networking*, vol. 7, no. 5, pp. 779-786, October 1999.

[Lin96] P.J. Lin, *Wide Area Optical Backbone Networks*, MIT Ph.D. Thesis, February 1996.

[Mar+93] M.A. Marsan, A. Bianco, E. Leonardi, and F. Neri, "Topologies for wavelength-routing all-optical networks," *IEEE/ACM Transactions on Networking*, vol. 1, no. 5, pp. 534-546, October 1993.

[Muk+96] B. Mukherjee, D. Banerjee, S. Ramamurthy, and A. Mukherjee, "Some principles for designing a wide-area WDM optical network,"*IEEE/ACM Transactions on Networking*, vol. 4, no. 5, pp. 684-696, October 1996.

[NLM02] A. Narula-Tam, P.J. Lin, and E. Modiano, "Efficient routing and wavelength assignment for reconfigurable WDM networks," *IEEE Journal of Selected Areas in Communications*, vol. 20, no. 1, pp. 75-88, January 2002.

[Pan92] R.K. Pankaj, *Architectures for Linear Lightwave Networks*, MIT Ph.D. Thesis, September 1992.

[RS95] R. Ramaswami and K.N. Sivarajan, "Routing and wavelength assignment in all-optical networks," *IEEE/ACM Transactions on Networking*, vol. 3, no. 5, pp. 489-500, October 1995.

[RS96] R. Ramaswami and K.N. Sivarajan, "Design of logical topologies for wavelength-routed optical networks," *IEEE Journal of Selected Areas in Communications*, vol. 14, no. 5, pp. 840-851, June 1996.

[RS01] R. Ramaswami and K.N. Sivarajan, *Optical Networks: A Practical Perspective.* Morgan-Kaufmann, 2001.

[SAS96] S. Subramaniam, M. Azizoglu, and A.K. Somani, "All-optical networks with sparse wavelength conversion," *IEEE/ACM Transactions on Networking*, vol. 4, no. 4, pp. 544-557, August 1996.

[Wil96] G. Wilfong, "Minimizing wavelengths in an all-optical ring network," in *Proceedings of International Symposium on Algorithms and Computation*, pp. 346-355, December 1996.

[YB97] J.Y. Yoo and S. Banerjee, "Design, Analysis, and Implementation of Wavelength-Routed All-Optical Networks," *IEEE Communications Surveys*, 1997.

[ZJM00] H. Zang, J.P. Jue, and B. Mukherjee, "A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks," *Optical Networks Magazine*, vol. 1, no. 1, pp. 47-60, January 2000.

[ZA95] Z. Zhang and A.S. Acampora, "A heuristic wavelength assignment algorithm for multihop WDM networks with wavelength routing and wavelength reuse," *IEEE/ACM Transactions on Networking*, vol. 3, no. 3, pp. 281-288, June 1995.

[ZRP00] Y. Zhu, G.N. Rouskas, and H.G. Perros, "A path decomposition algorithm for computing blocking probabilities in wavelength routing networks," *IEEE/ACM Transactions on Networking*, vol. 8, no. 6, pp. 747-762, December 2000.