Routing of Streams in WDM Reconfigurable Networks

Ciro A. Noronha Jr. and Fouad A. Tobagi

Technical Report No. CSL-TR-94-649

December 1994

This work was in part supported by NASA under grant NAGW-419, by NSF under grant NCR-9016032 and by Pacific Bell. Ciro Noronha was supported by a graduate scholarship from FAPESP from Sept/89 to Aug/93 under grant 89/1658.

Routing of Streams in WDM Reconfigurable Networks

Ciro A. Noronha Jr and Fouad A. Tobagi

Technical Report: CSL-TR-94-649

Computer Systems Laboratory
Departments of Electrical Engineering and Computer Science
Stanford University
Stanford, CA 94305

Abstract

Due to its low attenuation, fiber has become that medium of choice for point-to-point links. Using Wavelength-Division Multiplexing (WDM), many channels can be created in the same fiber. A network node equipped with a tunable optical transmitter can select any of these channels for sending data. An optical interconnection combines the signal from the various receivers in the network, and makes it available to the optical receivers, which may also be tunable. By properly tuning transmitters and/or receivers, point-to-point links can be dynamically created and destroyed. Therefore, in a WDM network, the routing algorithm has an additional degree of freedom compared to traditional networks: it can modify the network topology to create the routes. In this report, we consider the problem of routing audio/video streams in WDM networks. We present a general linear integer programming formulation for the problem. However, since this is a complex solution, we propose simpler heuristic algorithms, both for the unicast case and for the multicast case. The performance of these heuristics is evaluated in a number of scenarios, with a realistic traffic model, and from the evaluation we derive guidelines for usage of the heuristic algorithms.

Key Words and Phrases: WDM optical networks,multicast routing, multimedia, linear programming, simulated annealing, shortest path routing, minimum cost routing.

1 Introduction

Due to its low attenuation (less than 0.2 dB/km) and very high bandwidth, fiber has become the medium of choice for point-to-point links at high speeds, for any distance over ≈ 100 m and for data rates of 45 Mb/s and up. Using Wavelength-Division Multiplexing (WDM), many channels can be created in the same fiber. A network node equipped with a tunable transmitter can select any of these channels for sending data, and can dynamically change this selection. The range of wavelengths addressable depends on the technology used to implement the tunable transmitter [1]. The optical signal from the transmitters in the network is combined by an optical interconnection (e.g, a WDM star coupler), and made available to a subset of the optical receivers, determined by the optical interconnection. Optical receivers can also be tunable [1]. The interconnection pattern between nodes is defined by the tuning of transmitters and receivers to specific wavelengths, and can be dynamically changed. In a traditional (fixed-topology) network, given a traffic, the routing algorithm is responsible for finding routes for each of its components, satisfying the traffic requirements. In a WDM network with tunable transmitters and receivers, the routing algorithm has an additional degree of freedom: it can choose (or modify) the topology.

We assume that the applications generate traffic in *sessions*. A session is a group of video/audio streams that are logically related. For example, a video-conference with P participants where each of the conferees can see all the other participants represents a session with P multipoint streams, from each conferee to the other $P-1$ participants. A *stream* is defined as a continuous flow of information (i.e., video frames or audio samples) that has to be delivered in a timely fashion. Multimedia streams have the following new requirements (when compared to traditional data traffic), which must be taken into account when they are routed:

Bandwidth - multimedia streams use relatively high bandwidth on a continuous basis for long periods of time, while data traffic is bursty, but the average bandwidth used is low. For example, a high-quality compressed video stream can use anywhere from 1.5

to 8 Mb/s for extended periods of time, while the average bandwidth used by typical data applications can be well below 1 Mb/s.

Multipoint Communications - it is expected that a significant fraction of the multimedia traffic will be multipoint. Examples are videoconferencing, one-way video distribution and collaborative computing. Data applications, on the other hand, typically make only occasional use of multicasting.

Low Latency (on the order of 100-200 ms end-to-end), required for some applications (such as videoconferencing or collaborative computing) that provide interactive communications. Data applications typically do not have such strict latency constraints.

In this report, we consider the problem of routing multimedia streams in a WDM network. In section 2, we discuss the characteristics of the optical components used in building a network, and describe the previous work in optical WDM networks. In section 3, we give the problem formulation, and show that it can be solved exactly by linear programming. Since the problem is NP-complete, the optimum algorithm has worst-case exponential run-time; additionally, its implementation is complex. Therefore, in section 4 we give a number of heuristic algorithms for unicast and multicast routing; these heuristic algorithms find sub-optimal solutions. Evaluation of the heuristic algorithms is presented in sections 4.4.2, 6 (unicast traffic) and 7 (multicast traffic). For the unicast traffic case, we first derive an upper bound in the performance measure of interest, and show that the heuristic produces results that are close to the upper bound, thus obviating the need for pursuing the optimum solution. We also evaluate the performance of the WDM network in a dynamic environment, and compare it to that of a centralized switch. For the multicast traffic case, we compare the various heuristics proposed under a dynamic traffic environment. Our conclusions are presented in section 8.

2 Optical Network Components and Configurations

The optical WDM network has three basic "building blocks" [2]:

- optical interconnection;
- optical transmitters; and
- optical receivers.

In this section, we describe the characteristics of each of these components, and discuss the previous work in the area of WDM networks.

2.1 Optical Interconnection

The optical interconnection is responsible for mixing the light from the transmitters and splitting (dividing) it among the receivers, irrespective of the wavelength. The most common optical interconnection is the WDM star, shown in figure 1. The WDM star equally divides the optical power from each of the incoming ports among the output ports. The optical signal in each output port is a combination of the optical signals from each of the input ports, as shown in figure 1.

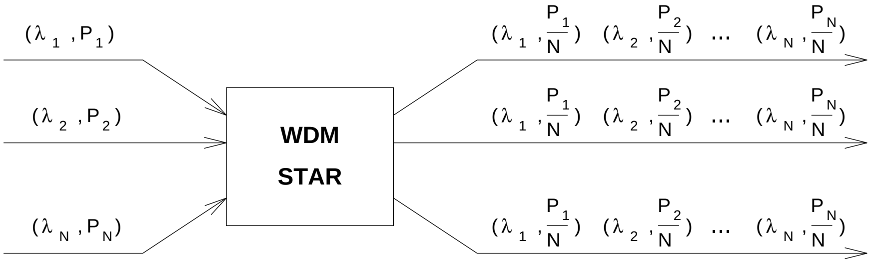


Figure 1: The WDM star coupler

Other optical interconnections, such as trees, multiple stars, etc., are possible; they do not necessarily divide the input power equally among all the outputs. In general, a given output will receive the signal from a subset of the inputs [3, 4].

2.2 Optical Transmitters

Optical transmitters are responsible for modulating the optical signal with the user data. There are two kinds of light sources for optical transmitters: LEDs and lasers. LEDs can only be amplitude-modulated because of their wide spectral widths, and cannot be modulated at very high data rates. They are unsuitable for use in a WDM network. Lasers can provide much larger output powers than LEDs, and can be modulated at much higher data rates. Moreover, due to the (relatively) narrow spectral width of the laser, it can be modulated not only in amplitude, but also in frequency or in phase.

There are several ways to build tunable lasers; the choice of methods usually represents a tradeoff between tuning speed on one side, and linewidth[1] and range of frequencies that can be addressed on the other side. Linewidth is important because, for a given modulation format, it determines the minimum channel spacing if WDM is used to combine several channels in the same fiber. Table 1 [5, 6] summarizes the characteristics of several tuning methods.

Table 1: Characteristics of Laser Tuning Methods

METHOD	RANGE	KIND	LINEWIDTH	SPEED
Electro-optical	7 nm	discrete	60 kHz	100 μs
Acusto-optical	70 nm	discrete	?	3 μs
2-section DFB	0.32 nm	continuous	?	< 5 ns
	3.3 nm	continuous	15 MHz	?
3-section DBR	8-10 nm	quasi-cont.	20-100 MHz	10 μs
	4.4 nm	continuous	1.9 MHz	10 μs
	2 nm	continuous	2.5-6.5 GHz	15 ns

---

[1] Due the the phase noise, the laser output spectrum is not an ideal line, but has a certain spectral width; the 3-dB spectral width is denoted by **linewidth.**

Another important transmitter parameter is the optical power. If the optical signal from a transmitter is split among multiple receivers by the optical interconnection, the optical power is one of the factors determining the number of receivers that can be reached.

2.3 Optical Receivers

The main component in an optical receiver is the photodiode, which converts the incoming light into an electrical signal. The photodiode responds to the optical power of the signal; it is largely independent of the wavelength. There are basically two kinds of optical receivers [2, 1]:

Direct Detection Receivers: The optical signal is applied directly to the photodiode. This kind of receiver can be used only with amplitude modulation (ASK). In a WDM system, the direct detection receiver must be preceded by an optical filter [7], which allows only a single wavelength to reach the photodiode. If the optical filter is tunable, the receiver will be tunable. Table 2 presents the characteristics of the various types of tunable filters [8].

Table 2: Tunable Filter Characteristics

TYPE	RANGE	BW (nm)	CHANNELS	LOSS	SPEED
Fabry-Perot	50 nm	< 0.01	100s	5 dB	ms
Acusto-Optics	400 nm	1	100s	5 dB	μs
Electro-Optics	10 nm	1	10	5 dB	ns
Active Semiconductors	1-4 nm	0.05	10	0 dB	ns

Coherent Receivers: Coherent receivers mix the light from a local laser with the incoming signal, prior to applying it to the photodiode. The local laser is kept synchronized to the transmit laser by means of a PLL. A coherent receiver responds only to a specific wavelength, defined by the wavelength of the local laser. The coherent receiver can be tuned by changing the wavelength of the local laser; see table 1 for the characteristics of tunable lasers. Coherent receivers are more complex than direct detection receivers.

An optical WDM receiver is characterized by the following parameters:

Sensitivity: Defined as the minimum power at the input of the receiver that still guarantees a bit error probability not higher than 10^{-9}. A more fundamental measure is the number of photons per bit required to achieve this error probability, but this quantity only makes sense on a limit situation where the dominant noise is the shot noise. In general, coherent receivers have higher sensitivity than direct detection receivers.

Minimum Channel Separation: Minimum separation, in wavelength, between the center frequencies of two distinct channels. Direct detection receivers use optical filters, which have large passbands, resulting on large channel separations. Coherent receivers do the filtering in the electrical domain, which allows for the use of much sharper filters; in this case, the channel separation is determined by the modulation format and the combined linewidth of the transmitter and local oscillator lasers.

Tuning Speed: Time for a receiver to switch from one wavelength to another. Coherent receivers can be tuned by adjusting their local oscillator lasers; the figures on table 1 are valid for this case too. However, if the local laser must be kept in phase with the transmitter laser (which is required for PSK and FSK), then the total tuning time includes also a component corresponding to the time necessary for achieving the phase lock, which will depend on the receiver structure. The tuning times for the various types of optical filters are given in table 2

2.4 Optical Network Configurations and Operation

The basic function of a network is to transport the data generated by the users and deliver it to the destination, with the appropriate quality of service. The physical implementation of the network depends on where the users are and what kind of service they expect. For example, a single shared channel is a reasonable implementation for a local area network, while in the wide-area it is more reasonable to implement a network with point-to-point links, operating in a store-and-forward fashion.

We classify the previous work into two categories: *Local Area Networks* (LANs) and *Wide Area Networks* (WANs). The work in Local Area Networks is characterized by the fact that a direct channel is established between the sender and the receiver, and communication is single-hop. The network interconnection is usually assumed to be a WDM star. The work in Wide Area Networks is characterized by the multi-hop communication aspect, and by a more general optical interconnection. The Metropolitan Area Network (MAN) is an intermediate case; some MAN schemes are multiple-hop, others are shared-channel. A comprehensive review of the work in the field of WDM networks can be found in [9] and [10]. Unless explicitly stated, all the work described in this section is theoretical. A survey of the work in experimental WDM networks can be found in [1]. Except for IBM's RAINBOW, which will be described in the next section, none of the experimental WDM networks has contributed to the field of routing: the bulk of the work done there was in the actual implementation, and since all these networks have a small number of nodes, routing is really not an issue for them.

WDM Local Area Networks/Single Hop Operation

In Local Area Networks, typically there is a direct channel between the sender and the receiver. In existing networks such as an Ethernet segment, the bandwidth of this channel is shared between all the nodes connected to it, and is an upper bound in the throughput of the network. As the user traffic increases, either the channel bandwidth must be increased, or more channels must be provided (and switching between these channels). WDM is a way of providing more channels, and the switching function can be implemented by having tunable transmitters and/or receivers. The following observations can be made:

- If the receivers are tunable, the network can provide physical multicasting, by tuning multiple receivers to the wavelength used by a given transmitter. However, there is a coordination problem, because the switching action (tuning) happens at the receiver, which must be somehow informed that the sender wishes to initiate the communication.
- If the transmitters are tunable, there is no sender-receiver coordination problem, be-

cause the switching action happens at the sender. However, multiple transmitters can potentially tune to the same wavelength; this represents a collision, and the resulting signal in general cannot be received. This problem can be dealt with by having some sort of coordination between senders (so it does not happen) or by providing some sort of multiple-access scheme, to recover from collisions.

Habbab et al [11] and later Mehravari [12] considered a WDM star network where the number of distinct wavelengths is much less than the number of stations. Each station has one tunable receiver and one tunable transmitter, and both are capable of addressing all the wavelengths in the network. Coordination between transmitters and receivers is achieved by reserving one wavelength for control; all idle nodes keep their receivers tuned to this wavelength. When a node decides to transmit, it chooses one of the data wavelengths at random and sends a packet in the control channel informing the destination of this choice. It then tunes its transmitter to the data wavelength chosen and sends the packet. Multiple-access schemes are used both in the control and in the data channels. The authors study the network throughput and delay as a function of the multiple-access schemes used in the control channel and in the data channels.

Chlamtac and Ganz [13] and later Ganz and Koren [14] considered a scenario where all the stations are synchronized, the transmitters are fixed and the receivers are tunable. All packets arrive aligned at the star coupler. Coordination between transmitters and receivers is achieved by having a common "tuning schedule", known by all nodes; the wavelengths are used in a TDM fashion. Control algorithms and approximate analysis based on Markov chains are presented.

The RAINBOW network [15] is a local/metropolitan area WDM network intended to cover a diameter of 25 km, designed and implemented by IBM. It connects up to 32 IBM PS/2's through a 32 × 32 passive star coupler and allows the computers to communicate circuit-switched data at a rate of 300 Mb/s/node, yielding an aggregate throughput of up to 9.6 Gb/s. The network's physical topology is the WDM star. Each computer is equipped with its own fixed frequency optical transmitter and tunable optical receiver. The optical transmitters utilize directly modulated distributed feedback (DFB) laser diodes. Wavelength selection at the receiver is accomplished with a tunable fiber Fabry-Perot filter whose cavity spacing is varied piezoelectrically. To open a circuit, a node tunes its receiver to the destination's wavelength, and starts transmitting a "request" pattern in its wavelength. Idle receivers are continuously polling the transmit wavelengths, looking for requests. Once a request is found, the node will keep its receiver tuned to the requestor's wavelength, and will acknowledge in is own wavelength. Communication now can start. The time for the receiver to identify and lock to a channel is 10 ms.

In summary, the work done in single-hop algorithms assumes that the tuning of transmitters and/or receivers can be very fast, and that the network either uses a multiple-access scheme (which is difficult to implement efficiently in optics) or is synchronized (which might be difficult to achieve at high speeds).

Wide Area Networks/Multi-Hop Operation

In a Wide-Area Network, due to its size and geographical distribution of nodes, it is not possible (or reasonable) to have channels shared by the nodes. For example, it is reasonable to connect all the nodes in a building in a star topology; all the fibers go to a closet where they connect to a WDM star coupler. However, it is not reasonable to connect all the major network nodes in a country to a single "central" star; the delay and loss in the fiber would be unacceptable. In this latter case, a mesh topology is more indicated; communication between neighboring nodes will happen with a minimum of delay. In a WAN, links are usually point-to-point, and communication happens in a store-and-forward manner. One of the first WDM networks proposed, the ShuffleNet [16], was a store-and-forward network. Transmitters and receivers were fixed, and the "links" were the WDM channels. It was no different than an interconnection of nodes using point-to-point links in a certain specific topology.

In [17, 18, 19, 20] it is assumed that the network reconfiguration process will be performed infrequently; during the reconfiguration, the network may even be non operational. The problem then becomes similar to a traditional topological design problem, where the traffic is Poisson and the traffic matrix is known, with additional constraints introduced by the fact that each node has a well-defined number of transmitters and receivers. They all consider that the combined optical signal from all transmitters is available to all receivers. The differences are in the following areas:

(i) The fiber plant: paper [17] also considers the design of the fiber plant (optical power budget, propagation delays). Papers [18, 19, 20] do not make any additional assumptions about the optical interconnection.

(ii) Objective function to be optimized: in [17], the objective function is to minimize the average delay; the authors assume a queue model for the nodes, which makes the delay a non-linear function of the flow in the links. They also take into account the propagation delays in the network. In [18], the network is assumed to operate under deflection routing, and the average delay is indirectly minimized by minimizing the length of the alternate paths between sources and destinations. In [19, 20], the objective function is to minimize the maximum flow over all links.

(iii) Additional constraints in the optimization: in [20] tunability restrictions are assumed, i.e., receivers can only be tuned to a subset of the available bandwidths. The other papers do not have additional constraints.

(iv) Solution method: in [17, 18] the objective function is non-linear, and the authors resort to the "simulated annealing" method to search for a sub-optimal solution. In [19, 20] the authors present an heuristic algorithm which divides the problem into two subproblems - the wavelength assignment subproblem and the routing subproblem, which are solved by linear programming.

Summary of Previous Work

In summary, the previous work in the field of WDM networks can be classified into single-hop routing (appropriate for LANs, and maybe MANs) and multi-hop routing (appropriate for MANs/WANs). For single-hop routing, one has to assume either a multiple-access scheme or synchronization between nodes; both are difficult to efficiently implement in practice. For multi-hop routing, it has been assumed that tuning of the transmitters and receivers happens over a very long time scale; the topology of the network does not change often, and when it does, it is in response to changes in the traffic matrix. The problem then becomes similar to the traditional topological design problem, with some additional constraints (i.e., the number of links leading to a node must be equal to the number of receivers in that node, and similarly for transmitters).

Restricting the WDM network to single-hop operation (tuning in a packet-by-packet basis) has practical implementation problems, and if the tuning is not fast enough, streams cannot be supported in this environment. The other extreme (reconfiguring the network only when the long-term traffic trends change) does not make use of the full switching potential of the WDM network. When dealing with streams, it is possible to reconfigure the network when requests arrive, and when streams terminate. Conceptually, this is similar to the long-term reconfiguration, but the change in traffic trends are actual stream arrivals and terminations.

3 Problem Formulation

In this section, we define the problem of routing streams in a WDM network. We start with the traffic model, which is the same as in as used in [21, 22], and then describe the assumptions we make about the optical network. We then give the problem formulation in precise mathematical terms, and show how it can be transformed into an integer linear programming problem. The approach taken is to present a sequence of linear programming formulations, starting from the simplest (unicast traffic, unit link labels, no latency constraints) and reaching the most general case.

3.1 The Traffic Model

For the traffic, we assume that user's requests come in *sessions.* A session is a group of streams that are logically related. We will denote by T the number of streams in the session. Stream i, $i = 1, \ldots, T$, is characterized by its source s_i, its n_i destinations $d_{i1}, d_{i2}, \ldots, d_{in_i}$, its bandwidth requirement r_i and its maximum latency constraint D_i. We assume that all the streams in the session arrive and depart simultaneously, the session arrival process is a Poisson process with rate λ, and the session duration is exponentially distributed with rate μ.

3.2 Network Assumptions

In this report, we assume that the network operates in a store-and-forward, multi-hop operation, but the reconfiguration of the network happens in a stream-by-stream basis, creating paths as the streams arrive, and removing them after they terminate. Bandwidth in the links can be shared in a TDM fashion.

We make the following assumptions about the network, which is depicted in Figure 2:

- There are N nodes in the network; node i, $i = 1, \ldots, N$ is equipped with S_i optical transmitters and P_i optical receivers.

- The optical interconnection is such that all receivers have access to the light signal from all transmitters. No other assumptions are made about it. This assumption simplifies the formulation of the problem, but limits the results to the LAN/MAN environment.

- The number of distinct wavelengths, denoted by W, is larger than the number of transmitters/receivers in the network. Due to the large available in the fiber, this is a reasonable assumption.

- At any time, only one transmitter can be tuned to a given wavelength. We do not consider multiple-access operation (i.e., many transmitters tuned to the same wavelength), because this is difficult to implement efficiently in optics.

- Any given transmitter can be connected to any given receiver - there are no tunability restrictions. Current technology allows the implementation of transmitters and receivers that are tunable over wide ranges, making this a reasonable assumption.

- Usually, S_i and P_i are much less than N. Therefore, each node will have direct connectivity to a (typically small) subset of nodes.

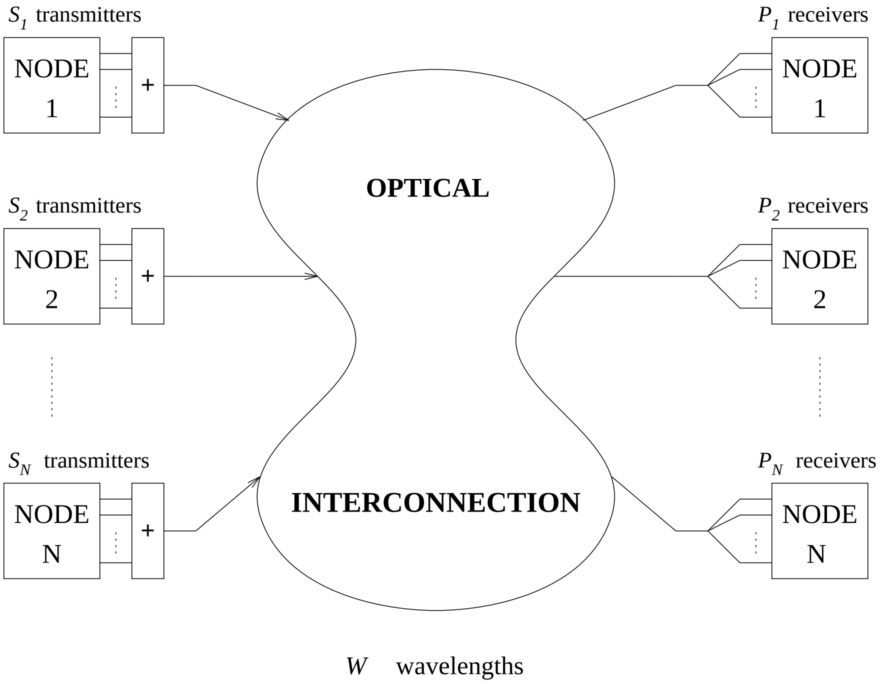


Figure 2: The WDM Network

3.3 Statement of the Problem

The problem under consideration can be stated as: "Given a session with T streams, each stream being characterized by its source, destinations, latency constraint and bandwidth requirement, find the logical network topology and routes that satisfy the stream requirements, while optimizing a given objective function."

A solution to the reconfiguration/routing process is composed of two parts: (i) the wavelength assignment, which defines which transmitters are connected (tuned) to which receivers (and thus defines the network topology), and (ii) the assignment of routes given the wavelength assignment. If multiple solutions exist for a reconfiguration/routing problem, the objective function is the criterion used to select the "best" one. As done in [21, 22], the objective function is chosen to be a linear combination of costs and delays.

3.4 First Formulation: The Unicast Routing Problem

In this section, we present the mathematical formulation for the optimum reconfiguration and routing problem in its simplest case. To write this formulation, we make the following additional assumptions:

- There are no latency constraints.
- All the streams are unicast.
- The total number of transmitters in the network ($\sum_{i=1}^{N} S_i$) is equal to the total number of receivers in the network ($\sum_{i=1}^{N} P_i$) and will be denoted by K. This is a reasonable assumption under unicast traffic because all communications is one-to-one; "extra" transmitters or receivers will remain unused and thus can be ignored.
- No physical multicast is allowed, i.e., there is at most one receiver and one transmitter per wavelength. Under this assumption, it does not matter which element (i.e., the transmitter or the receiver) is tunable.
- Link costs and delays are all unity.

All these assumptions will be relaxed in latter formulations. Defining:

T : Number of streams in the session
$\{r_i\}$: Required bandwidth for stream i, $i = 1, \ldots, T$
$\{s_i\}$: Source node for stream i, $i = 1, \ldots, T$
$\{d_i\}$: Destination node for stream i, $i = 1, \ldots, T$
K : Total number of transmitters/receivers in the network

N : Number of nodes in the network

V : Bandwidth of each individual link

$\boldsymbol{P}$: Receiver distribution vector ($N \times 1$); P_i is the number of receivers in node i

$\boldsymbol{L}$: Transmitter location matrix ($N \times K$); $L_{ij} = 1$ if transmitter j is located in node i, otherwise $L_{ij} = 0$.

$\boldsymbol{R}$: Wavelength allocation matrix ($N \times K$); $R_{ij} = 1$ if transmitter j is sending in a wavelength currently being received at node i, otherwise $R_{ij} = 0$.

$\boldsymbol{B}^i$: Destination vector ($N \times 1$) for stream i, $i = 1, \ldots, T$; $B^i_{s_i} = 1$, $B^i_{d_i} = -1$, and $B^i_j = 0$ for $j = 1, \ldots, N$; $j \neq s_i$; $j \neq d_i$

$\boldsymbol{X}^i$: Routing vector ($K \times 1$) for stream i; $X^i_j = 1$ if stream i is routed through transmitter j, $i = 1, \ldots, T$, $j = 1, \ldots, K$

The problem formulation can be expressed as:

GIVEN: $K, N, V, T, \boldsymbol{L}, \boldsymbol{P}; \{s_i\}, \{d_i\}, \{r_i\}, i = 1, \ldots, T$

MINIMIZE: Average path length

$$\sum_{i=1}^{T} r_i \sum_{j=0}^{K} X^i_j \tag{1}$$

WITH RESPECT TO: $\boldsymbol{R}$, $\boldsymbol{X}^i, i = 1, \ldots, T$

UNDER CONSTRAINTS:

1. Communication is one-to-one, i.e., there is only one transmitter and one receiver per wavelength.

$$\sum_{i=1}^{N} R_{ij} = 1, \quad j = 1, \ldots, K \tag{2}$$

2. Node i has only P_i receivers.

$$\sum_{j=1}^{K} R_{ij} = P_i, \quad i = 1, \ldots, N \tag{3}$$

3. There should be a path from every source to every destination. This is equivalent to writing a set of flow conservation equations, for routing one unit of flow from the source to the destination of each stream in the session.

$$(\boldsymbol{L} - \boldsymbol{R})\boldsymbol{X}^i = \boldsymbol{B}^i, \quad i = 1, \ldots, T \tag{4}$$

4. The total bandwidth of the streams routed through a link should not exceed the link bandwidth:

$$\sum_{i=1}^{T} r_i X_j^i \leq V, \quad j = 1, \ldots, K \tag{5}$$

5. Integer constraints: receivers cannot be "divided".

$$\boldsymbol{R} \qquad \text{is binary} \tag{6}$$

   No bifurcation of flow (in a packet-switched network, this condition can be relaxed, in which case the stream might be "divided" into several routes):

$$\boldsymbol{X} \qquad \text{is binary} \tag{7}$$

The objective function (1) and constraints (2), (3), (4) and (5) define a non-linear optimization problem; the objective function is linear, but the constraint set (more specifically, equations (4) - the flow conservation equations) is not. When constraints (6) and (7) are added, it becomes a non-linear integer optimization problem.

However, the non-linearity in the constraint set comes just from the $\boldsymbol{R}\boldsymbol{X}^i$ product in equation (4). By using the fact that $\boldsymbol{R}$ and $\boldsymbol{X}^i$ are binary variables, and by increasing the number of equations and free variables, we can convert the routing/reconfiguration problem into a *linear* integer programming problem. We add to the set of free variables the $N \times K$ binary matrices $\boldsymbol{Z}^i$, $i = 1, \ldots, T$, subject to the following new constraints:

$$Z^i_{jk} \leq X^i_k \tag{8}$$

$$Z^i_{jk} \leq R_{jk} \tag{9}$$

$$i = 1, \ldots, T; \quad j = 1, \ldots, N; \quad k = 1, \ldots, K$$

Equation (4) then becomes :

$$\boldsymbol{L}\boldsymbol{X}^i - \boldsymbol{Z}^i\boldsymbol{1} = \boldsymbol{B}^i \tag{10}$$

where $\boldsymbol{1}$ is a $K \times 1$ vector with 1 in all positions.

In summary, by adding $\boldsymbol{Z}^i$ to the list of free variables, replacing equation (4) with equation (10), and adding inequalities (8) and (9) to the constraint set, the reconfiguration/routing problem becomes a linear integer programming problem, which can be solved by standard techniques such as the branch-and-bound method [23]. It should be noted that, for a given fixed topology (i.e., given $\boldsymbol{R}$), this problem reduces to the well-known multicommodity flow problem.

3.5 Second Formulation: Routing of Multicast Streams in a WDM Network with Tunable Transmitters

We now relax the following assumptions from the previous formulation:

- Streams can be multicast.
- Streams can have maximum latency constraints, measured in hops.
- The objective function is a linear combination of costs and delays, both measured in number of hops.

We still assume that tuning is one-to-one, i.e., physical multicast is not allowed. This would be the case in a WDM network where the transmitters are tunable. Of course, if we prohibit physical multicasting, this formulation also applies to a WDM network with tunable receivers.

Defining:

K : Total number of transmitters/receivers in the network

N : Number of nodes in the network

V : Bandwidth of each individual link

$\boldsymbol{P}$: Receiver distribution vector $(N \times 1)$; P_i is the number of receivers in node i

$\boldsymbol{L}$: Transmitter location matrix $(N \times K)$; $L_{ij} = 1$ if transmitter j is located in node i, otherwise $L_{ij} = 0$.

$\boldsymbol{R}$: Wavelength allocation matrix $(N \times K)$; $R_{ij} = 1$ if transmitter j is sending in a wavelength currently being received at node i, otherwise $R_{ij} = 0$.

T : Number of multicast streams.

s_i : Source node for multicast i

n_i : Number of destinations for multicast i

$\{d_{ik}\}$: Set of destinations for multicast i, $k = 1, \ldots, n_i$

r_i : Bandwidth requirement for multicast i

$\boldsymbol{X}^i$: $K \times n_i$ multicast routing matrix for multicast stream i. $X^i_{jk} = 1$ if transmitter j is used in the multicast path for stream i to reach destination d_{ik}, otherwise $X^i_{jk} = 0$, $k = 1, \ldots, n_i$.

Y^i : $K \times 1$ multicast path vector for stream i. $Y^i_j = 1$ if transmitter j is in the multicast path for stream i, otherwise $Y^i_j = 0$.

M_i : Delay for multicast request i, in hops.

D_i : Latency constraint for multicast request i, in hops.

$\boldsymbol{B}^i$: $N \times n_i$ source-destination matrix for multicast stream i; $B^i_{jk} = 1$ if $j = s_i$, $B^i_{jk} = -1$ if $j = d_{ik}$, and $B^i_{jk} = 0$ otherwise, $k = 1, \ldots, n_i$.

β_c : Weight of the cost in the optimization.

β_d : Weight of the delay in the optimization.

The optimum multicast routing routing problem in a WDM network can be formulated as follows:

GIVEN: $K, N, V, \boldsymbol{L}, \boldsymbol{P}, T, \beta_c, \beta_d; \{\boldsymbol{B}^i\}, \{r_i\}, \{D_i\}, i = 1, \ldots, T$

MINIMIZE:

$$\sum_{i=1}^{T} r_i \left(\beta_c \sum_{j=1}^{K} Y_j^i + \beta_d M_i \right) \tag{11}$$

WITH RESPECT TO: $\boldsymbol{R}; \boldsymbol{X}^i, \boldsymbol{Y}^i, M_i, \quad i = 1, \ldots, T$

UNDER CONSTRAINTS:

1. Physical communication is one-to-one, i.e., there is only one transmitter and one receiver per wavelength.

$$\sum_{i=1}^{N} R_{ij} = 1, \quad j = 1, \ldots, K \tag{12}$$

2. Node i has only P_i receivers.

$$\sum_{j=1}^{K} R_{ij} = P_i, \quad i = 1, \ldots, N \tag{13}$$

3. For every stream, there must be a path from its source to each of its destinations. This is equivalent to writing a set of flow conservation equations for routing one unit of flow from the source to each of the destinations:

$$(\boldsymbol{L} - \boldsymbol{R})\boldsymbol{X}^i = \boldsymbol{B}^i \quad i = 1, \ldots, T; \tag{14}$$

4. If a link is in the path from the source to any of the destinations, then it must be included in the multicast path.

$$X_{jk}^i \leq Y_j^i, \quad k = 1, \ldots, n_i, \quad j = 1, \ldots, K, \quad i = 1, \ldots, T; \tag{15}$$

5. The delay for a multicast is the delay to the farthest destination:

$$M_i - \sum_{j=1}^{K} X^i_{jk} \geq 0, \quad k = 1, \ldots, n_i, \quad i = 1, \ldots, T; \tag{16}$$

6. There is a maximum delay constraint for each of the multicast streams:

$$M_i \leq D_i, \quad i = 1, \ldots, T; \tag{17}$$

7. The total flow through a link cannot exceed its bandwidth:

$$\sum_{i=1}^{T} r_i \boldsymbol{Y}^i \leq V; \tag{18}$$

8. Integer constraints: no bifurcation of flow; a single path is taken from the source to each of the destinations.

$$\boldsymbol{X}, \boldsymbol{Y} \quad \text{are binary.} \tag{19}$$

Receivers cannot be "divided":

$$\boldsymbol{R} \qquad \text{is binary} \tag{20}$$

The objective function (11) and constraints (12) to (18) define a non-linear optimization problem; the objective function is linear, but the constraint set (more specifically, equations (14) - the flow conservation equations) is not. When constraints (19) and (20) are added, it becomes a non-linear integer optimization problem.

However, the non-linearity in the constraint set comes just from the $\boldsymbol{R}\boldsymbol{X}^i$ product in equation (14). By using the fact that $\boldsymbol{R}$ and $\boldsymbol{X}^i$ are binary variables, and by increasing the number of equations and free variables, we can convert the routing/reconfiguration problem into a *linear* integer programming problem. We add to the set of free variables the $N \times K$ binary matrices $\boldsymbol{Z}^{ij}$, $i = 1, \ldots, T$, $j = 1, \ldots, n_i$, subject to the following new constraints:

$$Z_{kl}^{ij} \leq X_{jl}^{i} \tag{21}$$

$$Z_{kl}^{ij} \leq R_{kl} \tag{22}$$

$$i = 1, \ldots, T; \quad j = 1, \ldots, n_i; \quad k = 1, \ldots, N; \quad l = 1, \ldots, K$$

Equation (14) then becomes :

$$\boldsymbol{L}\boldsymbol{X}_j^i - \boldsymbol{Z}^{ij}\boldsymbol{1} = \boldsymbol{B}_j^i \tag{23}$$

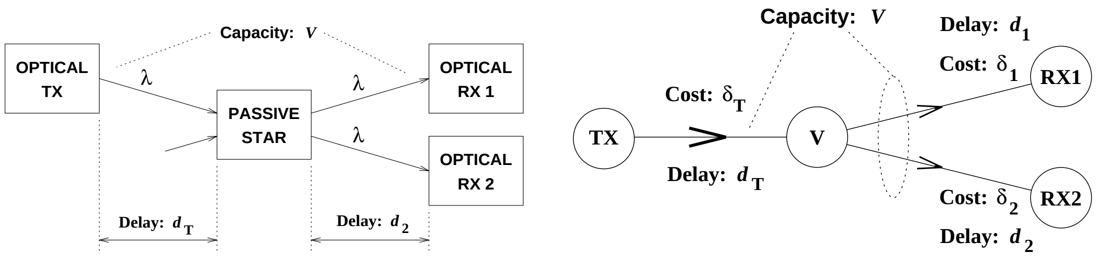
$$i = 1, \ldots, T; \quad j = 1, \ldots, n_i$$

where $\boldsymbol{1}$ is a $K \times 1$ vector with 1 in all positions.

In summary, by adding $\boldsymbol{Z}^{ij}$ to the list of free variables, replacing equation (14) with equation (23), and adding inequalities (21) and (22) to the constraint set, the reconfiguration/routing problem becomes a linear integer programming problem, which can be solved by standard techniques such as the branch-and-bound method [23]. It should be noted that, for a given fixed topology (i.e., given $\boldsymbol{R}$) and for unicast traffic, this problem reduces to the well-known multicommodity flow problem.

3.6 Third Formulation: Routing of Multicast Streams in a WDM Network with Tunable Receivers

An optical WDM network where the receivers are tunable is able to provide *physical multicasting*, by having multiple receivers tune to the same wavelength. As depicted in Figure 3, this physical multicasting can be modeled by creating, for each transmitter, a virtual node that is reached with delay and cost equivalent to the delay and cost from the transmitter to the "center" of the network (the WDM star). The link between the real node and the virtual node models the fact that the capacity out of the transmitter is V. The "replication" of the data happens at the virtual node.



Physical Multicasting

Model for Physical Multicasting

Figure 3: Physical multicasting in a WDM network

The difference between this formulation in this section and the previous one is that now we allow physical multicasting. To do that, it becomes necessary to augment the network with the virtual nodes, one for each transmitter. Also, we no longer assume that the network has the same total number of transmitters and receivers.

Initially, we define:

N : Number of actual (not virtual) nodes in the network
P_i : Number of receivers in node i, $i = 1, \ldots, N$
S_i : Number of transmitters in node i, $i = 1, \ldots, N$
P : Total number of receivers in the network; $P = \sum_{i=1}^{N} P_i$
S : Total number of transmitters in the network; $S = \sum_{i=1}^{N} S_i$
K : Total number of transmitters and receivers in the network; $K = S + P$
N' : Total number of nodes (real and virtual) in the network; $N' = N + S$

To each transmitter and each receiver in the network we assign a number; transmitters are numbered from 1 to S, and receivers from $S + 1$ to K. Using this numbering scheme, when we refer to "transceiver j", it can be either a transmitter or a receiver, according to the value of j. We also number the network nodes from 1 to N and the virtual nodes from $N + 1$ to N'; when we refer to "node i", it can be either a real node or a virtual node, depending on the value of i. Virtual node $N + i$ corresponds to transmitter i, $i = 1, \ldots, S$. Defining:

V : Bandwidth of each individual link
T : Number of multicast streams.

s_i : Source node for multicast i

n_i : Number of destinations for multicast i

$\{d_{ik}\}$: Set of destinations for multicast i, $k = 1, \ldots, n_i$

r_i : Bandwidth requirement for multicast i

$\boldsymbol{X}^i$: $K \times n_i$ multicast routing matrix for multicast stream i. $X^i_{jk} = 1$ if transceiver j is used in the multicast path for stream i to reach destination d_{ik}, otherwise $X^i_{jk} = 0$, $k = 1, \ldots, n_i$.

Y^i : $K \times 1$ multicast path vector for stream i. $Y^i_j = 1$ if transceiver j is in the multicast path for stream i, otherwise $Y^i_j = 0$.

M_i : Delay for multicast request i, in hops.

D_i : Latency constraint for multicast request i, in hops.

$\boldsymbol{L}$: Transmitter location matrix ($N \times S$); $L_{ij} = 1$ if transmitter j is located in node i, otherwise $L_{ij} = 0$.

$\boldsymbol{R}$: Receiver location matrix ($N \times P$); $R_{ij} = 1$ if receiver j is located at node i, otherwise $R_{ij} = 0$.

$\boldsymbol{F}$: Wavelength allocation matrix ($S \times P$); $F_{ij} = 1$ if receiver $S + j$ is tuned to the wavelength of transmitter i.

$\boldsymbol{B}^i$: $N \times n_i$ source-destination matrix for multicast stream i; $B^i_{jk} = 1$ if $j = s_i$, $B^i_{jk} = -1$ if $j = d_{ik}$, and $B^i_{jk} = 0$ otherwise, $k = 1, \ldots, n_i$.

β_c : Weight of the cost in the optimization.

β_d : Weight of the delay in the optimization.

The optimum multicast routing routing problem in a WDM network with tunable receivers can be formulated as follows:

GIVEN: $K, N, V, \boldsymbol{L}, \boldsymbol{R}, T, \beta_c, \beta_d; \{\boldsymbol{B}^i\}, \{r_i\}, \{D_i\}, i = 1, \ldots, T$

MINIMIZE:

$$\sum_{i=1}^{T} r_i \left(\beta_c \sum_{j=1}^{K} Y^i_j + \beta_d M_i \right) \tag{24}$$

WITH RESPECT TO: $\boldsymbol{F}; \boldsymbol{X}^i, \boldsymbol{Y}^i, M_i, \quad i = 1, \ldots, T$

UNDER CONSTRAINTS:

1. A receiver can be listening to only one wavelength:

$$\sum_{i=1}^{S} F_{ij} = 1, \quad j = 1, \ldots, P \tag{25}$$

2. For every stream, there must be a path from its source to each of its destinations:

$$\left[\begin{array}{c|c} \boldsymbol{L} & -\boldsymbol{R} \\ \hline -\boldsymbol{I_S} & \boldsymbol{F} \end{array}\right] \left[\begin{array}{c} \boldsymbol{Xt^i} \\ \hline \boldsymbol{Xr^i} \end{array}\right] = \left[\begin{array}{c} \boldsymbol{B^i} \\ \hline 0 \end{array}\right] \quad i = 1, \ldots, T \tag{26}$$

where $\boldsymbol{I_S}$ is the $S \times S$ identity matrix, $\boldsymbol{Xt^i}$ is a matrix with the first S rows of $\boldsymbol{X^i}$, and $\boldsymbol{Xr^i}$ is a matrix with the remaining P rows of $\boldsymbol{X^i}$. This matrix equation can be divided into the following two equations:

$$\boldsymbol{LXt^i} - \boldsymbol{RXr^i} = \boldsymbol{B^i} \tag{27}$$

$$-\boldsymbol{Xt^i} + \boldsymbol{FXr^i} = 0 \tag{28}$$

3. If a link is in the path from the source to any of the destinations, then it must be included in the multicast path.

$$X^i_{jk} \leq Y^i_j, \quad k = 1, \ldots, n_i, \quad j = 1, \ldots, K, \quad i = 1, \ldots, T; \tag{29}$$

4. The delay for a multicast is the delay to the farthest destination:

$$M_i - \sum_{j=1}^{K} X^i_{jk} \geq 0, \quad k = 1, \ldots, n_i, \quad i = 1, \ldots, T; \tag{30}$$

5. There is a maximum delay constraint for each of the multicast streams:

$$M_i \leq D_i, \quad i = 1, \ldots, T; \tag{31}$$

6. The total flow through a link cannot exceed its bandwidth:

$$\sum_{i=1}^{T} r_i \boldsymbol{Y}^i \leq V; \tag{32}$$

7. Integer constraints: no bifurcation of flow; a single path is taken from the source to each of the destinations.

$$\boldsymbol{X}, \boldsymbol{Y} \quad \text{are binary.} \tag{33}$$

Receivers cannot be "divided":

$$\boldsymbol{R} \qquad \text{is binary} \tag{34}$$

As in the previous formulation, the only non-linear equation in this optimization problem is equation (28), which has the $\boldsymbol{F}\boldsymbol{X}\boldsymbol{r}^i$ product. The problem is made linear by adding to the set of free variables the $S \times P$ binary matrices $\boldsymbol{Z}^{ij}$, $i = 1, \ldots, T$, $j = 1, \ldots, n_i$, subject to the following new constraints:

$$\begin{aligned} Z_{kl}^{ij} &\leq Xr_{jl}^i & (35) \\ Z_{kl}^{ij} &\leq F_{kl} & (36) \end{aligned}$$

$$i = 1, \ldots, T; \quad j = 1, \ldots, n_i; \quad k = 1, \ldots, S; \quad l = 1, \ldots, P$$

Equation (28) then becomes :

$$-\boldsymbol{X}\boldsymbol{t}_j^i + \boldsymbol{Z}^{ij}\mathbf{1} = 0 \tag{37}$$

$$i = 1, \ldots, T; \quad j = 1, \ldots, n_i$$

where $\mathbf{1}$ is a $P \times 1$ vector with 1 in all positions.

In summary, by adding $\boldsymbol{Z}^{ij}$ to the list of free variables, replacing equation (28) with equation (37), and adding inequalities (35) and (36) to the constraint set, the reconfiguration/routing problem becomes a linear integer programming problem.

3.7 A General Linear Programming Formulation for the Optimum Multicast Routing Problem

In this report, we have presented several optimum routing formulations, based on integer linear programming, each specifically tailored to a particular scenario. In this section, we present a general formulation that encompasses all the previous ones, and can be used in any of the previous scenarios (although not very efficiently). The main shortcoming the formulation presented in [21, 22] is that it cannot accommodate WDM networks. The main shortcoming of the unicast and multicast formulations presented so far for the WDM network is that it implicitly assumes unit link costs and delays. Ideally, we should be able to assign a cost and a delay to each of the transmitters and receivers in the WDM network; when there is a connection between a given transmitter and a given receiver, the delay and cost of the link created will be the sum of the transmitter and receiver costs and delays. For example, if the physical topology of the WDM network is the star, the propagation delay between two nodes corresponds to the propagation delay from the first node to the star (which is proportional to that node's distance to the star) plus the propagation delay from the star to the second node.

Another scenario not included in the previous formulations is the case of tunable transmitters and receivers, when the number of available wavelengths is *smaller* than the number of transmitters (if it is larger, then one would just tune each transmitter to a different wavelength and leave it fixed). In this case, the "transmitter virtual nodes" of the previous section represent the distinct wavelengths, and are not necessarily associated with a specific transmitter. Formally, the identity matrix $\boldsymbol{I_S}$ in equation 26 becomes a (non-square) transmitter assignment matrix.

Since this formulation is very similar to the previous one, we use the same symbols, with the following additions:

$\mathcal{C}$: $K \times 1$ cost vector; $\mathcal{C}_i$ is the cost associated with transmitter i, $i = 1, \ldots, S$, and $\mathcal{C}_{S+i}$ is the cost associated with receiver i, $i = 1, \ldots, P$.

$\boldsymbol{\mathcal{D}}$: $K \times 1$ delay vector; $\mathcal{D}_i$ is the delay associated with transmitter i, $i = 1, \ldots, S$, and $\mathcal{D}_{S+i}$ is the delay associated with receiver i, $i = 1, \ldots, P$.

W : Number of available wavelengths.

$\boldsymbol{G}$: $W \times S$ transmitter assignment matrix; $G_{ij} = 1$ if transmitter j is sending on wavelength i.

$\mathcal{E}$: Maximum number of receivers that can be connected to a transmitter. If transmitters are tunable, $\mathcal{E} = 1$; if receivers are tunable, $\mathcal{E} = P$. This formulation does not preclude the use of $1 \leq \mathcal{E} \leq P$.

The general formulation is:

GIVEN: $K, N, V, \boldsymbol{L}, \boldsymbol{R}, \boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{D}}, T, \beta_c, \beta_d, \mathcal{E}; \{\boldsymbol{B}^i\}, \{r_i\}, \{D_i\}, i = 1, \ldots, T$

MINIMIZE:

$$\sum_{i=1}^{T} r_i \left(\beta_c \boldsymbol{\mathcal{C}} \boldsymbol{Y}^i + \beta_d M_i\right) \tag{38}$$

WITH RESPECT TO: $\boldsymbol{F}, \boldsymbol{G}; \boldsymbol{X}^i, \boldsymbol{Y}^i, \boldsymbol{Zr}^{ij}, \boldsymbol{Zt}^{ij}, M_i, \quad i = 1, \ldots, T; j = 1, \ldots, n_i$

UNDER CONSTRAINTS:

1. A receiver can be listening to only one wavelength:

$$\sum_{i=1}^{S} F_{ij} = 1, \quad j = 1, \ldots, P \tag{39}$$

2. The number of receivers connected to a transmitter can be at most $\mathcal{E}$:

$$\sum_{j=1}^{P} F_{ij} \leq \mathcal{E}, \quad i = 1, \ldots, S \tag{40}$$

3. No more than one transmitter can be sending on each wavelength:

$$\sum_{j=1}^{S} G_{ij} \leq 1, \quad i = 1, \ldots, W \tag{41}$$

4. A transmitter sends in only one wavelength:

$$\sum_{i=1}^{W} G_{ij} \leq 1, \quad j = 1, ldots, S \tag{42}$$

5. For every stream, there must be a path from its source to each of its destinations:

$$\begin{aligned} \boldsymbol{LXt}^i - \boldsymbol{RXr}^i &= \boldsymbol{B}^i & (43) \\ -\boldsymbol{Zt}^{ij}\mathbf{1} + \boldsymbol{Zr}^{ij}\mathbf{1} &= 0 \qquad i = 1, \ldots, T; \quad j = 1, \ldots, n_i & (44) \end{aligned}$$

6. Flow to the receivers can only be sent if the link is in place:

$$\begin{aligned} Zr^{ij}_{kl} &\leq Xr^{i}_{jl} & (45) \\ Zr^{ij}_{kl} &\leq F_{kl} & (46) \end{aligned}$$

$$i = 1, \ldots, T; \quad j = 1, \ldots, n_i; \quad k = 1, \ldots, W; \quad l = 1, \ldots, P$$

7. Flow from the transmitters can only be sent if wavelengths have been allocated:

$$\begin{aligned} Zt^{ij}_{kl} &\leq Xt^{i}_{jl} & (47) \\ Zt^{ij}_{kl} &\leq G_{kl} & (48) \end{aligned}$$

$$i = 1, \ldots, T; \quad j = 1, \ldots, n_i; \quad k = 1, \ldots, W; \quad l = 1, \ldots, S$$

8. If a link is in the path from the source to any of the destinations, then it must be included in the multicast path.

$$X^{i}_{jk} \leq Y^{i}_{j}, \quad k = 1, \ldots, n_i, \quad j = 1, \ldots, K, \quad i = 1, \ldots, T \tag{49}$$

9. The delay for a multicast is the delay to the farthest destination:

$$M_i - \sum_{j=1}^{K} \mathcal{D}_j X^i_{jk} \geq 0, \quad k = 1, \ldots, n_i, \quad i = 1, \ldots, T \tag{50}$$

10. There is a maximum delay constraint for each of the multicast streams:

$$M_i \leq D_i, \quad i = 1, \ldots, T \tag{51}$$

11. The total flow through a link cannot exceed its bandwidth:

$$\sum_{i=1}^{T} r_i \boldsymbol{Y}^i \leq V \tag{52}$$

12. Integer constraints: no bifurcation of flow; a single path is taken from the source to each of the destinations.

$$\boldsymbol{X}, \boldsymbol{Y} \quad \text{are binary.} \tag{53}$$

Receivers cannot be “divided”:

$$\boldsymbol{F} \qquad \text{is binary} \tag{54}$$

The product of the allocation matrices and the flows must be binary:

$$\boldsymbol{Zr}, \boldsymbol{Zt} \qquad \text{are binary} \tag{55}$$

The above formulation is completely general:

- For fixed-topology networks, one just has to fix the $\boldsymbol{F}$ matrix and set $\mathcal{E}$ to 1; the costs and delays associated with the “transmitters” are set to zero, and the actual link costs and delays are associated with the receivers. In the particular case of unicast sessions, fixed-topology networks, no latency constraints, this formulation reduces to the traditional multicommodity flow problem

- For WDM networks, one can set $\mathcal{E}$ to 1 if the transmitters are tunable, or set $\mathcal{E}$ to P if the receivers are tunable. Note that, if the receivers are tunable, even under unicast traffic it might make sense to tune two receivers to the same transmitter - two unicast streams can be sharing that transmitter's bandwidth, each addressed to a different receiver.
- If the number of wavelengths is bigger or equal to the number of transmitters, one just has to set $\boldsymbol{G}$ to $\boldsymbol{I}_S$.

4 Heuristic Algorithms for the Reconfiguration and Routing Problem

The reconfiguration and routing problem, as formulated in section 3, is NP-complete, and the exact optimum solution given there has (in the worst case) exponential run time. In this section, we present a number of simpler heuristic solutions. We start by presenting a heuristic algorithm for the unicast case, and use this heuristic algorithm to build minimum-cost and minimum-delay heuristics for the multicast case.

Given a session with one or more streams, we seek to find the logical network topology and the routes for this session. From a high level point of view, the heuristic solutions proposed here start with an arbitrary initial logical topology, and make changes to it considering the streams in the session one at a time. The changes are made using the *Shortest Path with Reconfiguration Algorithm*, a variation of Dijkstra's Shortest Path algorithm proposed by us that works in a reconfigurable network environment. In the following, we first describe the Shortest Path with Reconfiguration Algorithm, and then give the complete reconfiguration and routing heuristics.

4.1 The Shortest Path with Reconfiguration Algorithm

Given a source node, Dijkstra's algorithm builds a shortest path *tree* from that node. The tree starts with the source node, and at each iteration a node is added to it in such a way that the paths in the tree are the shortest from the source. When used to find the shortest path between a particular pair of nodes, the algorithm terminates when the destination node is added to the tree, at which point the remainder of the tree is discarded and only the path between the source and the destination nodes is retained.

Our objective is to compute the shortest path in a WDM network, where the topology of the network is a free variable that can also be used to minimize the path length. In the best case, we would just tune a transmitter at the source and a receiver at the destination to the same wavelength, and obtain the shortest possible path, with length equals to one hop. However, this might not always be possible, since transmitters and receivers might be already connected to other nodes. In general, we classify the transmitters and receivers in the network either as *free* or *locked*, and the shortest path algorithm can only reconfigure the free transmitters and receivers, although it might make use of the locked ones in whatever topology they happen to be. If the WDM network supports physical multicasting (i.e., if it has tunable receivers), this is taken into account in the algorithm by *considering all the transmitters as free, without regard to the other connections.* The algorithm described below does exactly this: given the WDM network in a certain logical topology, where some links are free and some are locked, and a source-destination pair, it finds the shortest path between these two nodes, reconfiguring the free links if necessary. In the Appendix, we give a formal description of the algorithm.

Step 1: Using Dijkstra's algorithm, identify: (i) the shortest path between the source and the destination, and (ii) the node closest to the source which has a free transmitter (i.e., either the source itself or the first node added to the shortest path tree that has a free transmitter); this node, if found, will be denoted by *Node A*. Note that if the network is disconnected, there might not be a path between the source and the destination.

Step 2: Using Dijkstra' algorithm in reverse from the destination to the source (i.e., building the tree in reverse), find the node closest to the destination that has a free receiver; this node, if found, will be denoted by *Node B*.

Step 3: If either node A or node B or both were not found, stop. If a path between the source and the destination was found in step 1, it is the shortest path. Otherwise, there is no path. If both node A and node B were found, proceed to step 4.

Step 4: Let L_1 denote the length of the shortest path found in step 1 (make $L_1 = \infty$ if no path was found), and L_2 denote the length of the path obtained by tuning the transmitter in node A to the receiver in node B, and using the shortest path from the source to A, the newly-created A-B link, and the path from B to the destination. If $L_1 \leq L_2$, do not reconfigure the network and use the shortest path from step 1; otherwise, tune A to B and use the path just created, as described above.

NOTE: At most one reconfiguration is needed to obtain the shortest path (and the algorithm above finds it). This can easily be shown by contradiction: assume that the shortest path between nodes S and R requires that node A be reconfigured to connect to node B, and node C be reconfigured to connect to node D. In the absence of tuning constraints, we can reconfigure node A to connect directly to node D, finding a path that is shorter, which contradicts the initial hypothesis.

4.2 The Reconfiguration and Routing Heuristic for Unicast Streams

Given a session, the basic idea behind the wavelength assignment heuristic is to take an arbitrary initial topology, and apply the shortest path with reconfiguration to each of the components of the session. The shortest path with reconfiguration algorithm is applied to the streams in the session in decreasing order of bandwidth. In general, the first streams to be routed will be given shorter paths, as more network resources are available. Therefore, it is better to route first the higher-bandwidth streams, to minimize their usage of network resources.

Step 1: Choose an arbitrary initial wavelength assignment. Create a vector $\boldsymbol{U}$, containing the used bandwidth on each transmitter; initially, $U_i = 0$, $i = 1, \ldots, K$. Sort the streams in the session in order of bandwidth.

Step 2: Consider the stream with the highest bandwidth requirement that was not yet processed; let us denote it by stream j. Temporarily prune from the network topology the transmitter/receiver pairs that do not have enough free bandwidth to support the stream, i.e., belonging to the set $\{i : V - U_i < r_j\}$. Mark all the transmitter/receiver pairs belonging to the set $\{i : U_i > 0\}$ as locked, and the remainder as free.

Step 3: Execute the "Shortest Path with Reconfiguration Algorithm" described above for this stream. If successful, update the the $\boldsymbol{U}$ vector as follows: $U_i \leftarrow U_i - r_j$, $i \in$ path.

Step 4: If all streams in the session have been considered, terminate; otherwise, return to step 2.

After this algorithm is run, the initial network topology is transformed into a new topology which matches the session requirements. If all the invocations of the shortest path with reconfiguration algorithm in step 3 are successful, a set of routes for the session is also available; otherwise, the heuristic fails and declares the problem infeasible.

Note that, if the logical network topology has been defined, routing a session using this topology becomes the traditional multicommodity flow problem. More specifically, the $\boldsymbol{R}$ matrix in equation (4) ceases to be a free variable, thus making it a linear equation; the optimization problem then becomes equations (1), (4), (5) and (7).

As done in [21, 22] for the routing of multicast streams in fixed-topology networks, this integer linear programming problem can be solved by the traditional branch-and-bound method. Specific features of the problem can be used to prune the search space and speed-up the solution, as proposed by Crowder et al [24]. In fact, the same pruning rules presented in [21, 22] apply here; one just needs to remember that there is one single destination in the unicast case. The linear relaxation of the problem can also be efficiently solved by decomposition, with the difference that, in this case, there is only a single level of decomposition - a group of T unicast streams is decomposed into T unicast routing problems. The decomposition equations for this case are well-known and will not be presented here; the reader is referred to [25].

One can further optimize the routing solution by using the topology found by the heuristic, disregarding the routes found in step 3, and re-routing the session using the integer linear programming solution. In some cases, by doing this it is possible to solve a problem declared infeasible by the heuristic.

4.3 Using Simulated Annealing to Improve the Heuristic Solution

The *Simulated Annealing* method [26] is an optimization method designed for non-linear integer problems that are difficult to solve analytically. The method starts from a feasible solution, and perturbates this solution to see if it can be improved. Unlike traditional steepest-descent methods, simulated annealing can accept modifications to the current solution that do not improve it. This gives it the potential of moving away from a local optimum, and finding a better solution. The result given by the heuristic described in the previous section can be used as the starting point for the simulated annealing method; from that, the method can potentially identify a better solution, closer to the optimum.

The method mimics the annealing process for a metal or crystal. Initially the metal is melted, and its temperature is very high. The temperature is gradually lowered, and the metal will crystallize in a regular structure, with a minimum of energy. If the temperature is lowered too fast, the regular crystalline structure will not form. The same idea is applied to the optimization problem. The "temperature" controls the probability that a perturbation in the solution that does not improve the objective function is accepted. Initially, the "temperature" is high; it is then gradually lowered, and the solution should "coalesce" into the optimum. The algorithm works in "epochs" of constant temperature; each epoch is composed of a fixed number of perturbations in the current solution. A perturbation, in the case of the WDM network, corresponds to exchanging the connections of two transmitter-receiver pairs, as depicted in Figure 4 [27]. After the network topology is changed, the routes can be re-optimized using integer programming, as described in the previous section.

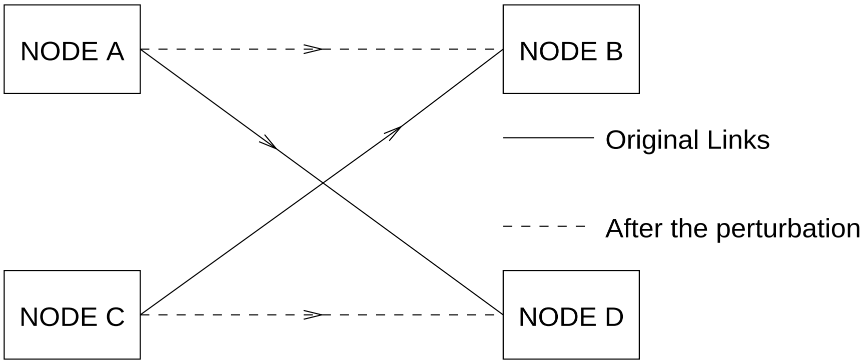


Figure 4: The perturbation

The algorithm keeps track of the "best" solution it has seen; let us denote the objective function of this solution as L^*. If a perturbation improves the objective function, it is accepted. Otherwise, it may be accepted with probability $\exp(-\frac{L-L^*}{T_e})$, where L is the new value of the objective function after the perturbation, and T_e is the current temperature. For routing in the WDM network, the "temperature" has no physical meaning; it is just a number selected by trial and error, based on the order of magnitude of the values of the objective function. To implement a simulated annealing optimization, one must find (experimentally, by trial and error) the appropriate values for the following parameters:

- Size (i.e., number of perturbations) of the annealing epoch.

- Temperatures:
  - Initial and final temperatures.
  - Rate of temperature decrease between annealing epochs.

4.4 Heuristic Algorithms for Multicast Routing In WDM Networks

In this section, we present minimum-cost and minimum-delay heuristic algorithms for WDM networks, based on the Shortest Path with Reconfiguration algorithm. The algorithms presented here are for a single multicast stream; bandwidth constraints are taken care of by pruning from the network those links with insufficient free capacity, and multiple streams are handled sequentially, as done for the unicast case.

4.4.1 Minimum Delay Heuristic Algorithm

On a fixed-topology network, minimum-delay multicast routing can be achieved by finding the shortest path from the source to each of the destinations and then merging these paths. For a single multicast, this is an exact algorithm, i.e., the routes found are optimal. The same idea can be applied to the WDM network: one would use the shortest path with reconfiguration algorithm between the source and each of the destinations, and merge the paths. However, unlike its fixed-topology counterpart, this algorithm might not be optimal even for a single stream, depending on whether or not physical multicasting is available.

In the fixed-topology network, using shortest path (with link delays as link labels) leads to minimum-delay paths. If this process is repeated from a source node to all its destinations, the delay to each destination will be minimum, and thus the delay of the multicast path (defined as the delay to the farthest destination) will be minimum. The paths can be computed independently, and then merged. In a WDM network, the Shortest Path with Reconfiguration algorithm is also optimum for a single destination. However, if physical multicasting is not allowed (or possible), as routes are computed, links get locked and the topology changes; the tree found is dependent on the order in which the routes to the individual destinations are computed. In this case, the global optimum cannot be decomposed into a number of individual subproblems. If physical multicasting is allowed (i.e., the receivers are tunable), then this algorithm is optimum, because: (i) all transmitters are always free, regardless of the tuning of the receivers, and (ii) if the path to two different destinations require the tuning of the same receiver, they will require the *same* tuning, and thus will not interfere with each other.

4.4.2 Minimum-Cost Routing Heuristic Algorithm

The minimum-cost multicast routing heuristic algorithm presented here is based on the Takahashi-Matsuyama (TM) minimum Steiner tree heuristic. The basic idea in the original TM algorithm is to start building a tree with the source node, and at each iteration add to the tree the destination closest to it. The same idea can be used in the WDM network, but here we use the shortest path with reconfiguration algorithm.

Formally, the minimum-cost heuristic is:

INPUTS: A WDM network, where some transmitters and receivers are locked into a certain topology, and some are free; and a multicast to be routed, characterized by its source node s, and its n destinations $\{d_1, d_2, \ldots, d_n\}$.

OUTPUTS: The updated topology for the WDM network, and the multicast path from the source to the destinations.

ALGORITHM:

Step 1: Add the source node s to the multicast path. Create a virtual node V where the path is "collapsed", i.e., all the nodes in the multicast path are removed from the network and node V "inherits" their transmitters and receivers.

Step 2: Find the shortest path with reconfiguration from V to all the multicast destinations not yet in V. Each path is computed independently from the others, i.e., without taking into account the changes in topology required by the other paths.

Step 3: From all the paths found in step 2, choose the shortest and discard the others. "Collapse" all the nodes in the path into V.

Step 4: If all the destinations have been added to V, stop. The desired multicast path is the result of merging all the paths in V. If there are still destinations not in V, return to step 2.

5 Performance Evaluation of The Reconfiguration and Routing Heuristic for Unicast Traffic

In this section, we present an evaluation of the reconfiguration and routing heuristic described in the previous section, considering a single session in an empty network. Ideally, one would compare the results of the heuristic with the exact (optimum) solution; however, we derive an upper bound in performance, which is much simpler to compute than the optimum, and use it in the evaluation. We also compare the performance of the WDM reconfigurable network with a fixed-topology network with the same number of nodes and links; for the evaluation, we chose the ShuffleNet [16]. The routes in the ShuffleNet were computed using integer programming [25, 23]. In summary, the main objectives of this section are to compare the performance of the heuristic proposed in the previous section for unicast traffic with the upper bound, and with the performance of a fixed-topology network. We also seek to evaluate the improvement in the heuristic brought upon by the simulated annealing method.

5.1 Evaluation Scenarios and Performance Measures

The first step in the evaluation is defining the evaluation scenarios and performance measures under which the algorithms are to be compared:

Evaluation Scenarios

For the evaluation, we consider networks with $N = 8$ nodes; each node has 2 optical transmitters and 2 optical receivers ($K = 16$). We consider the routing of a single session over an idle network (this is equivalent to making the session arrival rate, λ, much lower than the average session duration, $1/\mu$). The session is composed of T streams, $10 \leq T \leq 20$, and the sources and destinations of the streams are uniformly distributed over the network. The bandwidth requirement for each stream is chosen at random between 0 and 100% of the link bandwidth, using the following bimodal distribution (m is the average bandwidth requirement, expressed as a fraction of the link bandwidth V):

$$p_R(r) = \begin{cases} \frac{1-m}{m} & \text{if } r < m \\ \frac{m}{1-m} & \text{if } r \geq m \end{cases} \tag{56}$$

The average bandwidth required by the session, as a fraction of the total bandwidth in the network, is given by mT/K; we denote this quantity as the *Offered Load* to the network. For the evaluation, we vary the offered load between 0 and 0.9.

Performance Measures

The most basic performance measure is the *Session Acceptance Probability.* Given a large sample space of sessions, the session acceptance probability is the fraction of this sample space that can be routed in the network (i.e., the feasible region of the optimization problem described in 3 is not empty). Since we seek to minimize the *Average Path Length* (in hops), this is another useful performance measure. Note that these two performance measures are related: for a given algorithm, the average path length indicates the usage of network resources when routing a session. If this value is high, it is likely that the blocking probability will also be high.

5.2 An Upper Bound on the Session Blocking Probability

Given a session composed of T streams as described above, it might be impossible to route this session, regardless of the network reconfiguration algorithm. In this section, we establish a necessary condition for a session to be accepted.

If a session is to be accepted and routed, for each node in the network there should be at least one way of distributing the streams that originate from it (terminate at it) among its transmitters (receivers). For example, it is not possible to have three streams requesting 60% of a link's bandwidth originating at a node that has only two transmitters, although the three streams combined request less bandwidth than the total available. Formally, a **necessary** condition for the existence of a solution for the routing/reconfiguration problem in section 3, is that, for every node k, $k = 1, \dots, N$, at least one feasible solution is found for each of the problems below:

Problem 1: define $\mathcal{A}_k$ to be the set of streams that originate at node k; find a set α_{ij}, $i \in \mathcal{A}_k$, $j = 1 \dots, S_k$ such that:

$$\sum_{i \in \mathcal{A}_k} \alpha_{ij} r_i \leq V, \quad j = 1, \dots, S_k \tag{57}$$

$$\sum_{j=1}^{S_k} \alpha_{ij} = 1, \quad \alpha_{ij} \text{ is binary}, \quad \forall i \in \mathcal{A}_k$$

Problem 2: define $\mathcal{B}_k$ to be the set of streams that terminate at node k; find a set β_{ij}, $i \in \mathcal{B}_k$, $j = 1 \dots, P_k$ such that:

$$\sum_{i \in \mathcal{B}_k} \beta_{ij} r_i \leq V, \quad j = 1, \dots, P_k \tag{58}$$

$$\sum_{j=1}^{P_k} \beta_{ij} = 1, \quad \beta_{ij} \text{ is binary}, \quad \forall i \in \mathcal{B}_k$$

Although in general these problems could be solved by linear integer programming, for the purposes of this report we just implemented an exhaustive search, due to the relatively small number of streams per session in the cases evaluated.

5.3 Numerical Results

In this section, we present numerical acceptance probability and average path length results for the scenarios described above. In all cases, we are routing a single session on an empty network.

As indicated in section 4.3, before the simulated annealing method can be employed, we need to determine reasonable values for its parameters, based on test runs. These parameters are the number of perturbations in the annealing epoch, and the temperatures. Moreover, we need to estimate the size of the session sample space to estimate the acceptance probability. Therefore, we performed three sample runs, varying the number of perturbations in the annealing epoch, and the number of sessions in the sample space. We restricted ourselves to a single annealing epoch, with the temperature fixed at 1. Each session had $T = 12$ requests, and the average bandwidth per stream was set to $m = 0.35$. The average path length and acceptance probability are given in table 3, for the ShuffleNet, the reconfiguration and routing heuristic, and the simulated annealing (which uses the heuristic as a starting point).

Table 3: Number of Sessions and Annealing Epoch Size

			ShuffleNet		WDM Network			
					Heuristic		Sim. Anneal.	
Run	Pert.	Sessions	Len.	Prob.	Len.	Prob.	Len.	Prob.
1	100	100	2.0209	34%	1.11343	87%	1.10671	89%
2	100	500	2.03492	33.4%	1.10925	89.4%	1.10812	91.4%
3	1000	100	1.99385	36%	1.0983	86%	1.09366	90%

Len.: Average path length.

Prob.: Acceptance probability.

Based on the results from table 3, we decided to fix the number of perturbations at 100 and the number of sessions tried for each load at 150. We chose to keep the simulated annealing solution at one single epoch of temperature 1 until we could compare the results obtained with the upper bound derived in section 5.2.

We have simulated the scenarios described in section 5.1, and obtained both the session acceptance probability and the average number of hops, as a function of the offered load, for sessions composed of 10, 15 and 20 streams. We also obtained the same performance measures for an 8-node ShuffleNet, with 2 transmitters and 2 receivers per node (i.e., the same size as the WDM network under evaluation), using exactly the same sessions. The simulation results are given in Figures 5, 6 and 7, where we plot the session acceptance probability as a function of the offered load for the ShuffleNet and for the WDM reconfigurable network, for 10, 15 and 20 streams per session respectively, as well as the upper bound from section 5.2. For the WDM network, the plots show the session acceptance probability for the heuristic and for the simulated annealing. Since the heuristic solution was the starting point for the simulated annealing, its results have to be better or equal than those of the heuristic. The main conclusions from these plots are:

- The fact that the session acceptance probability for the proposed heuristic is close to the upper bound indicates that there is no need to pursue the optimum solution, since the room for improvement is *at most* the difference between the two solutions.

- The simulated annealing, in general, improved very little over the heuristic solution. Since there is little room for improvement anyway, we decided not to pursue annealing solutions with multiple epochs.

- As expected, the reconfigurable network significantly outperforms the fixed-topology network (the ShuffleNet), even under uniformly-addressed traffic. For example, for a 90% session acceptance probability, the reconfigurable network can carry twice the load of the ShuffleNet, for 15-stream sessions.

Figure 8 shows a comparison of the session acceptance probability for 10, 15 and 20 streams/session, both for the WDM network (using the routing and reconfiguration heuristic) and the ShuffleNet. As shown in Figure 8, for the same load, a session with a higher number

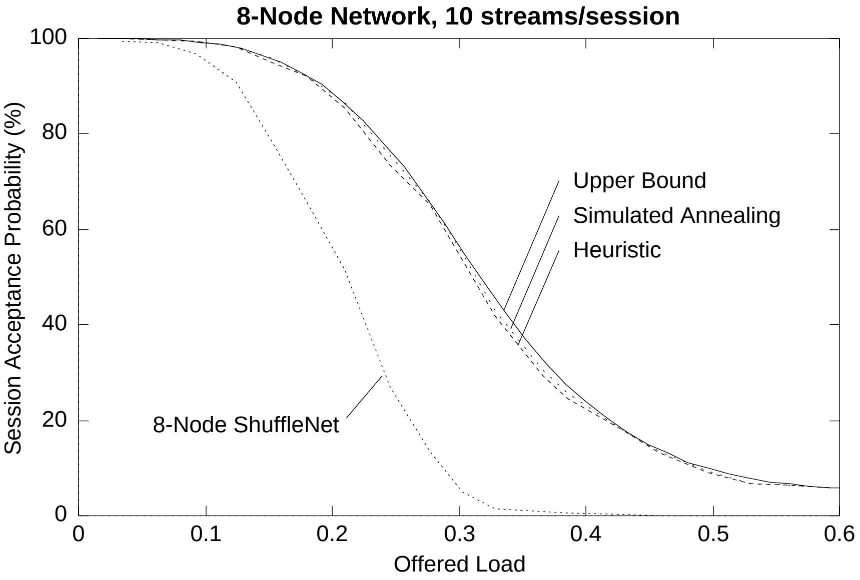


Figure 5: Session Acceptance Probability for 10 streams per session (150 sessions per point)

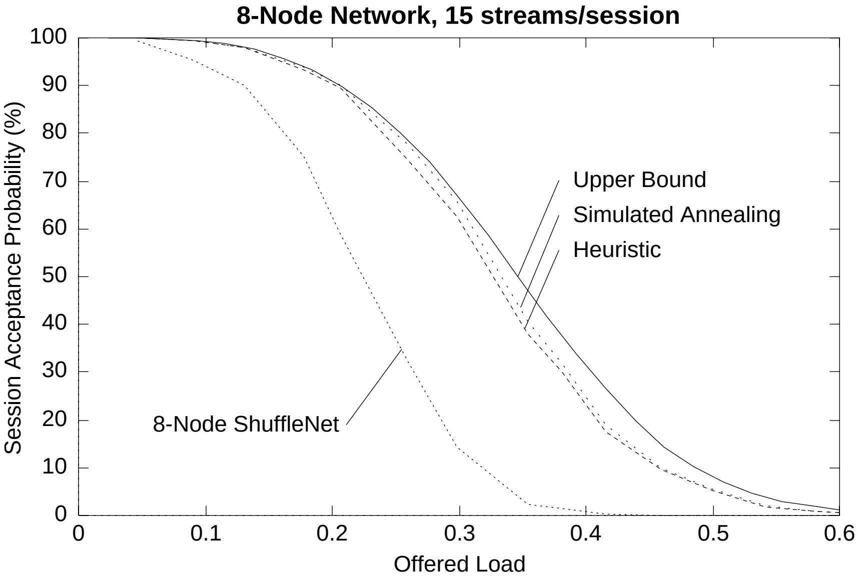


Figure 6: Session Acceptance Probability for 15 streams per session (150 sessions per point)

of streams will have a higher probability of being accepted, because the bandwidth of the individual streams will be lower, thus allowing more freedom in arranging them. This is always true for the ShuffleNet. For the reconfigurable network, however, at very high loads, this trend is reversed - the performance for sessions with smaller number of streams is better because (for the sessions accepted) it is possible to dedicate a link to each stream.

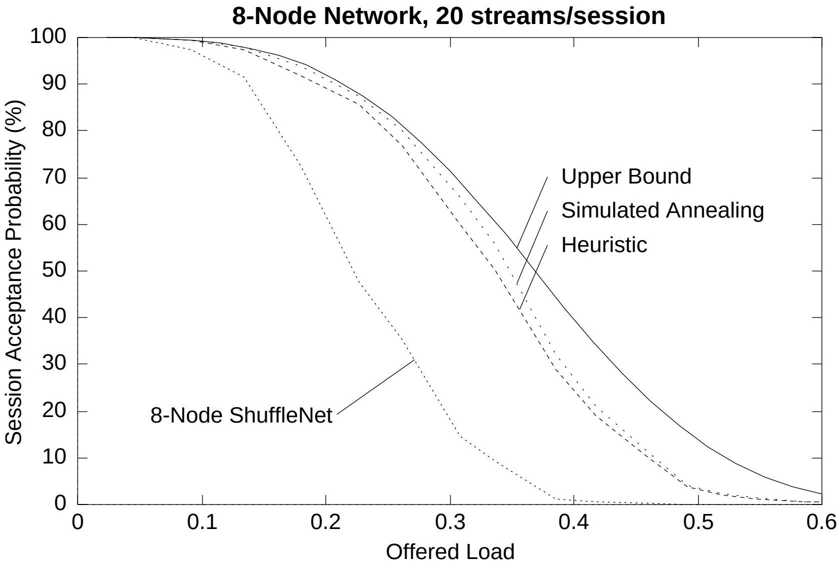


Figure 7: Session Acceptance Probability for 20 streams per session (150 sessions per point)

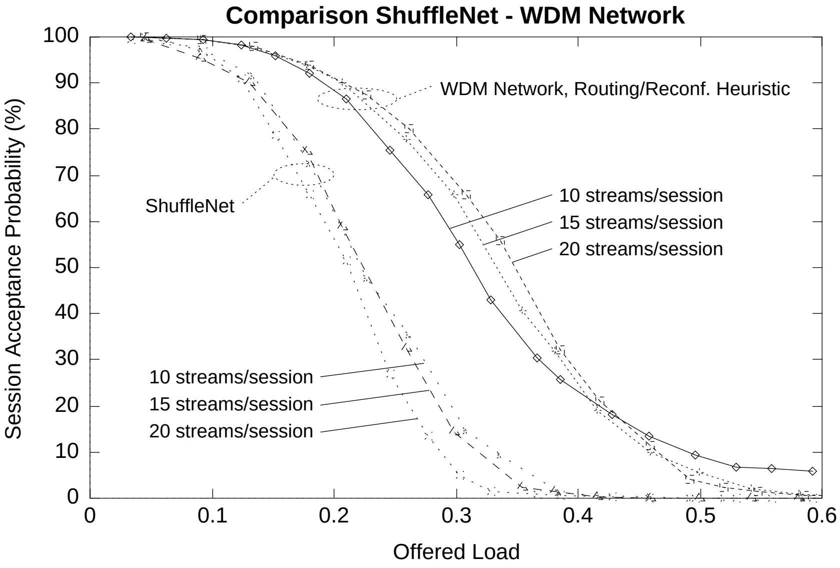


Figure 8: Comparison of the Session Acceptance Probability for the ShuffleNet and the WDM Network, 150 sessions/point

Figure 9 shows the average path length as a function of the offered load, for 20 streams per session (the plots for 10 and 15 streams/session are similar), and it further confirms our observation that the performance of the reconfigurable network is better than the fixed-topology one; while the average path length for the ShuffleNet is around 2 hops, the path length for the reconfigurable network, even at high loads, is close to 1 hop, which is, of course, the minimum possible. Again, there is very little improvement by using the simulated annealing method.

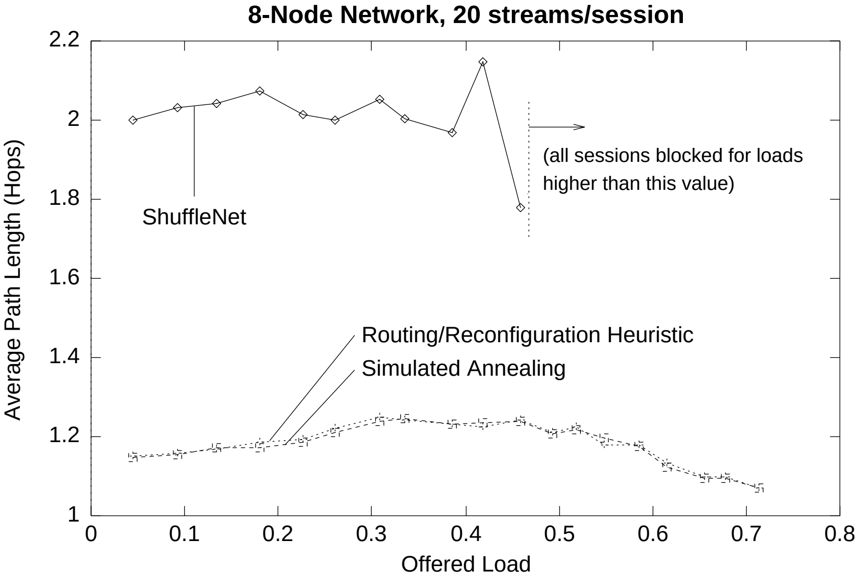


Figure 9: Average Path Length for 20 streams/session, 150 sessions/point

6 Evaluation of the WDM Network for a Dynamic Traffic Model Under Unicast Traffic

In the previous section, we evaluated the reconfiguration/routing algorithm in a static environment, i.e., the routing of a single session with a fixed number of streams on an idle network. Although this evaluation was useful to show that there is no motivation to pursue the optimum, it is not an indication of the actual performance of the algorithm in a realistic environment. In this section, we evaluate the WDM network and the routing/reconfiguration in a realistic environment, where sessions arrive, are routed (or dropped), and if accepted stay in the network for a certain period of time. We first describe the operation of the network in such a scenario, keeping in mind the traffic requirements, and then evaluate its performance, which we compare to that of a centralized switch.

6.1 Operation of the WDM Network in a Dynamic Environment

The main difference between a WDM network and a traditional mesh network is that the former is able to dynamically change its topology; with current optical components, this can be accomplished in sub-millisecond time. The ability to change the network topology at will during operation, in a de-centralized fashion, gives rise to the following issues:

Control of the network: Control of the network is distributed; therefore, messages about the network topology have to be exchanged by the nodes. There are two ways to do that: (i) have a separate control network, whose topology is fixed (e.g., a ring), and send the control messages over this network; and (ii) keep the optical network strongly-connected, and reserve a certain amount of bandwidth in each link for management purposes. In this latter case, reconfigurations that partition the network should not be allowed. We chose to keep the network topology strongly-connected at all times; therefore, in the Appendix we describe a simple modification of the Shortest Path with Reconfiguration algorithm to keep the network connected by performing a secondary reconfiguration, if necessary. Therefore, all transmitters and receivers in the optical network are kept connected (tuned) at all times, even if they are not being used for data traffic.

Re-routing established sessions in use: For stream traffic, if a link in use is reconfigured, the stream has to be interrupted for re-routing. Video/audio streams can tolerate interruptions and delay variations as long as the receiving end pre-buffers a certain amount of data, to keep playing during interruptions. For interactive traffic, however, the amount of pre-buffering cannot be very large as it adds latency to the communication; it is generally recognized that the latency for interactive communications should be less than 200 ms. The decision of *when* to reconfigure the network represents a tradeoff between performance (in terms of session blocking probability) and the number of times an established stream is re-routed during its lifetime. The extremes for this tradeoff are:

- Reconfigure only the idle transmitters and receivers : this has the advantage that existing connections are never disturbed, but at high loads, it is unlikely that a link is completely idle; the network will find itself "locked" in a random configuration that is far from optimal, and will generally perform worse than a regular fixed-topology network under the same traffic conditions.
- Reconfigure the network at each arriving session: one could consider the arriving session and the sessions already established in the network as a new "larger" session to be routed on an empty network. This has the advantage that the network is always optimal, but a given stream might be re-routed an excessive number of times.

Considering the issues listed above, we propose the following model of operation for the reconfigurable network, which is illustrated in Figure 10:

- The network starts with some arbitrary strongly-connected topology.
- When a session arrives, it is either routed or blocked; blocked sessions are cleared.
- The reconfiguration/routing algorithm used to accept a session and route it is:

  Step 1: Try to route the session on the current network topology, using the shortest path with reconfiguration algorithm; unused links can be reconfigured. Prior to routing each stream in the session, temporarily prune from the network topology those links that do not have enough free bandwidth to support it. A session can be accepted only if the routing of all its component streams is successful. If the session was successfully routed, accept this route. The streams already established in the network will not be disturbed. Otherwise, proceed to step 2.

  Step 2: Since no path was found in step 1, the only alternative to accept this session would be to re-arrange existing connections. We use the unicast heuristic presented in section 4 to compute the new topology and routes, considering as our "session" the existing streams and the new session being added.

Step 3: If the reconfiguration/routing in step 2 was successful, we accept the new session and implement the reconfiguration. Otherwise, we block the incoming session and the network topology remains unchanged.

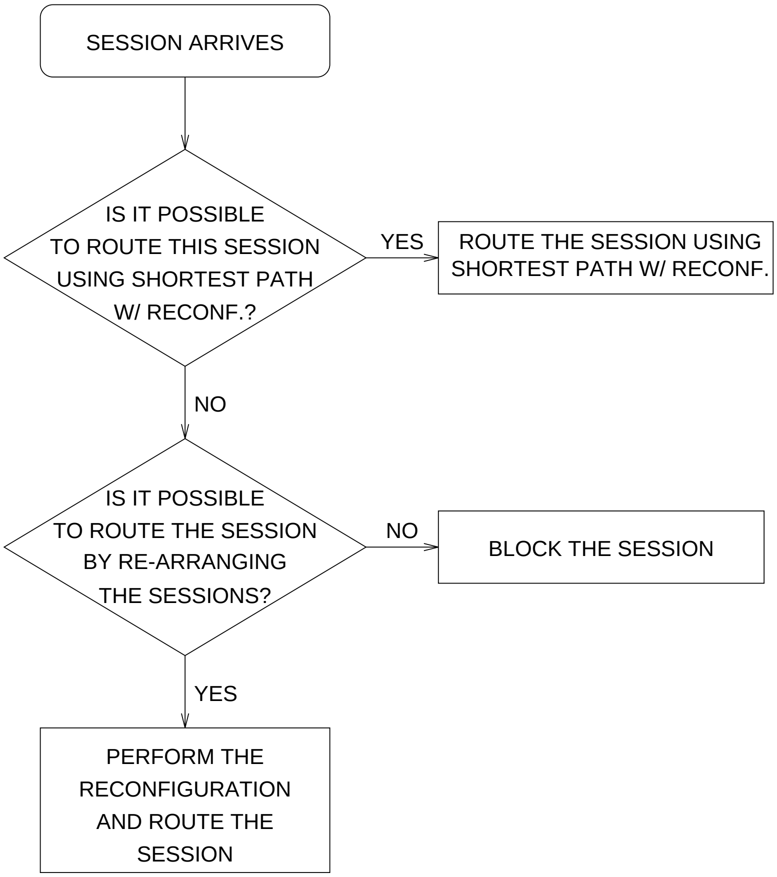


Figure 10: Dynamic operation of the WDM Reconfigurable Network

In summary, we reconfigure the network only when the cost of not doing so is to block a session; this way, we achieve the same blocking probability as if we were to reconfigure at every arriving session, while minimizing the number of times a given stream is re-routed.

6.2 Numerical Results

In this section, we present simulation results for the WDM network in a dynamic environment, where sessions arrive according to a Poisson process, and if not blocked, stay for an exponentially-distributed amount of time. The performance measures of interest are:

- Session blocking probability: probability that an arriving session cannot be routed and is blocked.
- Average time between successive reconfigurations.
- Average path change for re-routed streams; the path change is defined as the difference between the longest and the shortest paths experienced by the stream during its lifetime. This is a measure of the delay jitter introduced by the reconfiguration. This jitter has to be taken into account when defining the receive buffer sizes for video and audio streams.

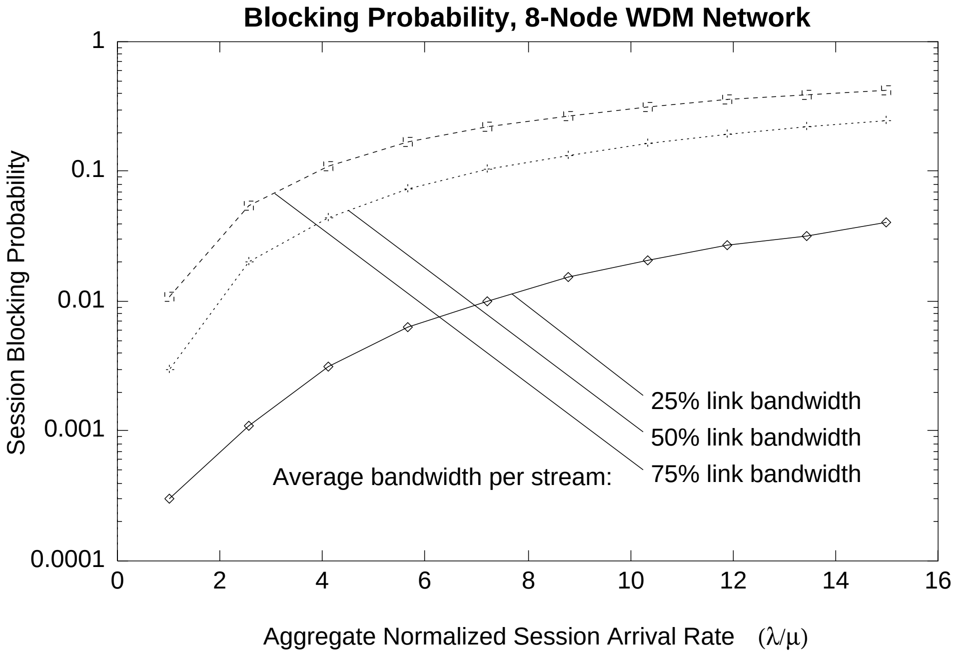


Figure 11: Session Blocking Probability (50,000 routes per point)

Figure 11 shows the session blocking probability in the WDM network as a function of the session arrival rate, and Figure 12 gives the breakdown of the blocking probability as a function of the stream bandwidth, for case where the average bandwidth requested is 50%. As Figure 12 indicates, streams requesting higher bandwidths are more likely to be blocked.

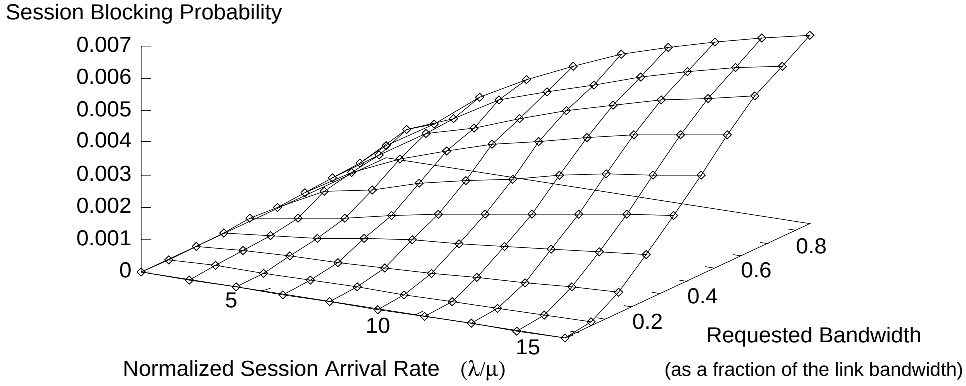


Figure 12: Session blocking probability, as a function of the stream bandwidth and the session arrival rate; average session bandwidth of 50%

Figure 13 shows the average path length as a function of the session arrival rate, for various values of the average stream bandwidth. At low arrival rates, the average path length is close to 1, since there is a high likelihood that the transceivers will be free. As the traffic load increases, the path length increases, as transceivers become locked and the streams are forced to take longer paths. Figure 13 also shows that the higher the average stream bandwidth, the lower the path length; the reason is that, since high-bandwidth streams use more resources, the network "fills up" faster, and only the sessions that lead to shorter path lengths can be accepted.

Figure 14 shows the average time between re-routes as a function of the session arrival rate. Note that the time is given as a multiple of the session duration, i.e., a value of 10 means that the average time between reconfigurations is 10 times the lifetime of a session; in the average, approximately only one out of 10 sessions will be re-routed during its lifetime. The figure shows that, at low loads, re-routing is seldom employed, becoming more frequent only at very high traffic loads. An interesting effect shown in Figure 14 is that, for a given session arrival rate, the average time between re-routes exhibits a minimum in relation to the average stream bandwidth. This happens because, when the average bandwidth is lower,

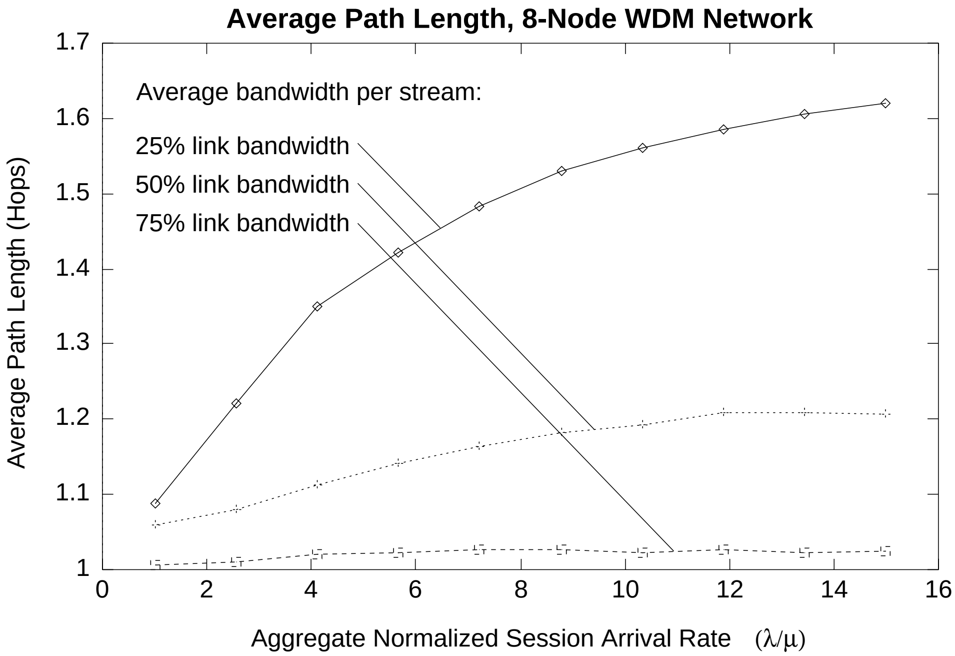


Figure 13: Average Path Length in the WDM Network

the network blocks less frequently; therefore, there are longer intervals between re-routing. As the bandwidth increases, blocking increases, but the new sessions can still be accepted after re-routing. However, at high bandwidths, it is less likely that the arriving session can be accepted even with re-routing; re-routing is again a less frequent event.

We also computed the average change in the path (difference between the maximum and the minimum paths during the lifetime of the stream), and found it to be under 0.6 hops in all cases, as shown in Figure 15, where we plot the average path change as a function of the session arrival rate. The figure indicates that, at low loads, most streams are not re-routed; as the load increases, the average path change increases. Lower bandwidth streams will suffer higher path changes than streams requesting higher bandwidths.

We note that the WDM network can be thought of as a "distributed switch", where the switching function is performed at the nodes and the "center" of the network is completely passive. However, the same function could be performed by a centralized switch (such as an ATM switch); routing is trivial (all paths are of the form source → switch → destination), and, if the switch is non-blocking, there is no contention inside the network: as long as there

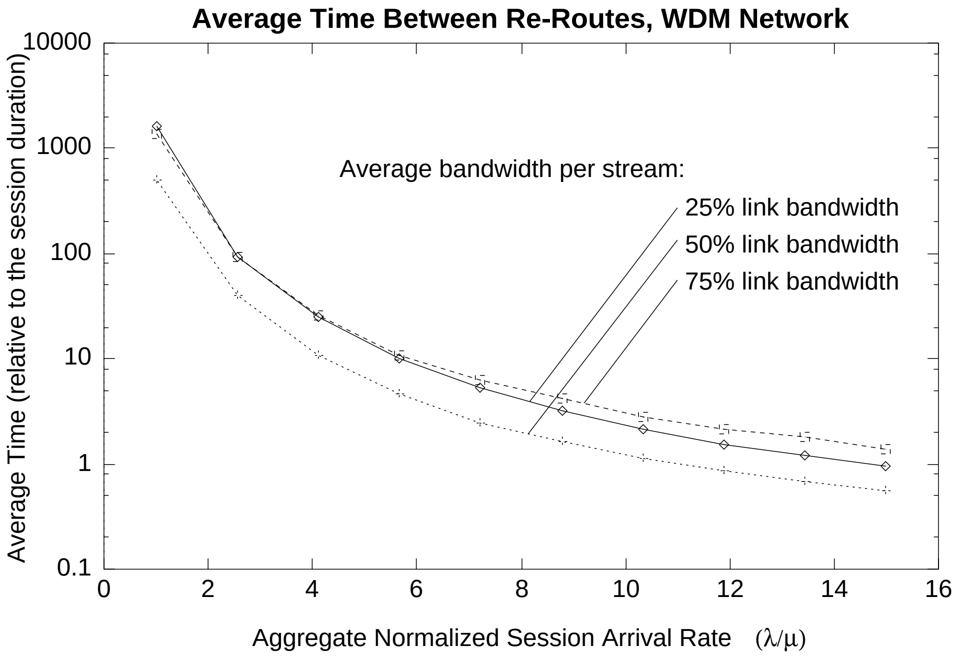


Figure 14: Average Time Between Re-Routes in the WDM Network

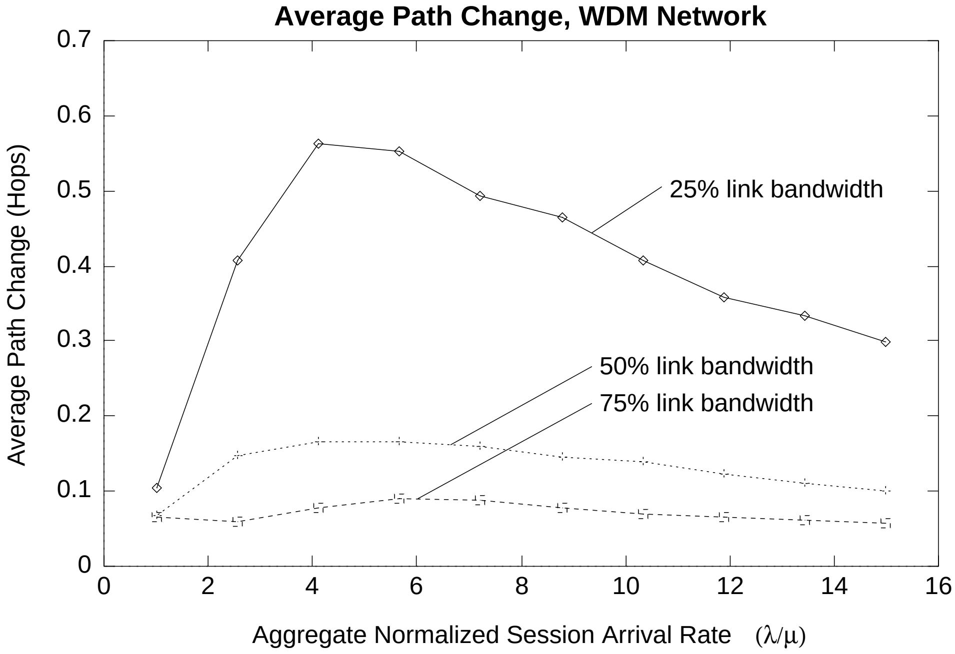


Figure 15: Average Path Change in the WDM Network

is available bandwidth in the link from the source to the switch and in the link from the switch to the destination, the stream can be accepted. So, it is important to determine if, from a performance point of view, there is any advantage in using a WDM network.

The two network configurations, using a "distributed switch" (WDM) and a centralized switch, are shown in Figure 16. For the centralized switch, there are two pairs of optical transmitters/receivers, one to carry the traffic from the node to the switch, and the other to carry the traffic from the switch to the node. A WDM network with the same number of optical transmitters and receivers would have two transmitters and receivers per node. We claim that the complexity of the two networks depicted in Figure 16 is approximately the same. The optical transceivers are more complex in the WDM case, but the "center" of the network is passive. On the other hand, in the centralized switch scenario, all the complexity is moved to the center of the network, and the optical transceivers are simpler. To complete the evaluation scenario, we still have to choose the transmitter/receiver data rates in the WDM case (V_{WDM}) and in the centralized switch case (V_{SW}). Two cases are possible:

- Same data rate for all transmitters and receivers in both situations, $V_{SW} = V_{WDM}$. This scenario corresponds to using the "same" transmitters and receivers both for the switch and for the WDM network. The switch has the advantage of no internal blocking, and a shorter path length; the WDM network has the advantage of having twice the output bandwidth per node, but some of this capacity has to be used to forward traffic from other nodes. We will denote this switch as "switch 1".

- Same output bandwidth for each node in both situations, $V_{SW} = 2V_{WDM}$ ($S_i = P_i = 2$), i.e., although the same number of transmitters and receivers is used in both networks, the ones connected to/from the switch have twice the data rate as the ones in the WDM network. The performance of this switch is the same as the upper bound on the performance of the WDM network previously discussed. We will call this "switch 2".

We have simulated the WDM network described above, as well as switch 1 and switch 2. In all cases, the same requests are offered to the three networks. For the all the evaluations in this section, sessions are composed of a single stream.

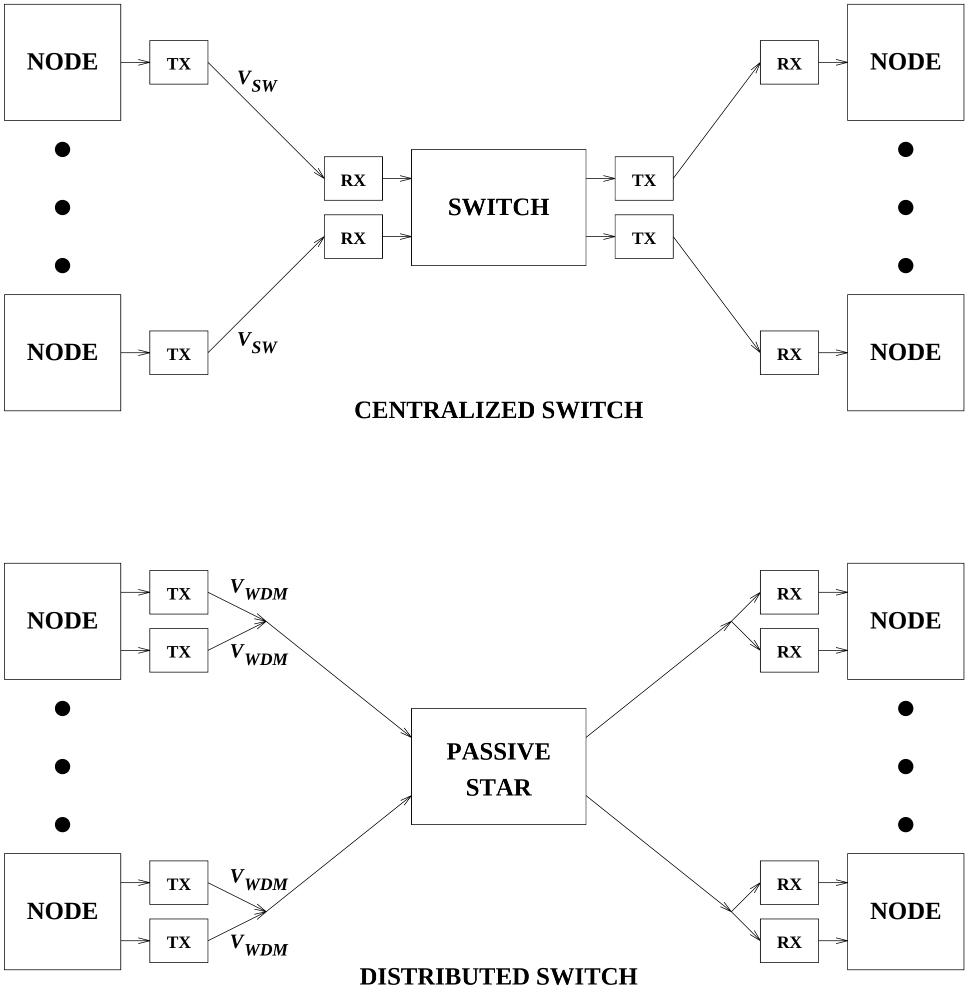


Figure 16: Distributed versus Centralized Switching

Figure 17 shows the session blocking probability for the three networks as a function of the session arrival rate. The plot shows that, as expected, the performance of the WDM network is lower than that of switch 2, but not significantly. The performance of switch 1, however, is much lower than that of the WDM network. In Figure 17, the average bandwidth requested per stream was set to 25% of the link bandwidth; we repeated the evaluation for other values of the stream bandwidth and found similar results.

In summary, we have shown that the performance of a WDM reconfigurable network is much superior to that of a centralized switch with roughly the same amount of optical hardware (switch 1 in the discussion), and is comparable to the performance of a switch using optical transmitters and receivers at twice the rate of the WDM counterparts (switch 2 in the

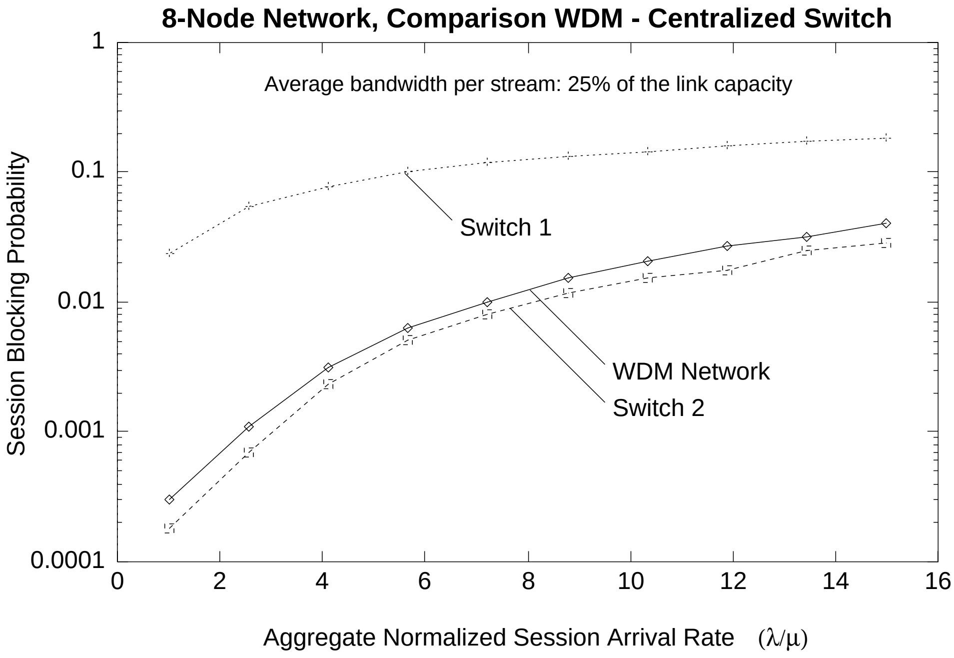


Figure 17: Blocking Probability results (single stream per session)

discussion). The WDM network will introduce an additional delay jitter due to stream re-routing. We have shown that, under reasonable traffic conditions, this re-routing represents a small effect, and will affect more the low-bandwidth streams than the higher-bandwidth ones. Moreover, if a specific stream cannot be re-routed for some reason[2], one just needs to mark the transmitters/receivers it is using as locked, and that stream will not be disturbed by network reconfiguration.

7 Evaluation of the Multicast Routing Heuristic Algorithms for WDM Networks

In this section, we present an evaluation of the heuristic algorithms proposed in section 4 for multicast routing in a WDM network. As done in [28, 29], we first evaluate the perfor-

---

[2] The amount of buffering in the end stations limits the number of times a stream can be re-routed. If a stream is destined to a station with very small buffers, it might not have enough "jitter budget" to support re-routing.

mance in a baseline case, and then determine how variations in the baseline case change the performance measures.

7.1 The Baseline Case

The baseline case used here is similar to the one used in the evaluation in [28, 29]. The following elements characterize the baseline case:

Traffic Model: We use the same traffic model as in the baseline case in [28, 29]: single-stream sessions, sources and destinations uniformly distributed in the network, stream bandwidths set to 10% of the link bandwidth, number of destinations uniformly distributed between 1 and 10. Sessions arrive according to a Poisson process, and if accepted remain in the network for an exponential period of time. No latency constraints were assumed.

Network Scenario: 12-node networks, each node having 2 transmitters and 2 receivers. The network starts with a strongly-connected topology generated at random. We consider the following three kinds of networks:

- networks with tunable transmitters (no physical multicast);
- networks with tunable receivers (physical multicast allowed); and
- fixed-topology networks (corresponding to the initial topology of the WDM network, which is a strongly-connected topology generated at random).

Transmitter/receiver costs (vector $\mathcal{C}$ in section 3.7 were set to 0.5, making the cost of a path equal to the number of hops in the path. The distances between the nodes and the WDM star coupler were generated at random, uniformly between 0 and 15; for each node, delays for its transmitters and receivers (vector $\mathcal{D}$ in section 3.7) were set to the distance to the star coupler.

Algorithms Evaluated: For each network scenario, we evaluated the minimum-cost and the minimum-delay heuristic. If the network under consideration allowed physical multicast, it was employed. For the fixed-topology case, the minimum-delay heuristic reduces to the traditional shortest path, and the minimum-cost heuristic to the Takahashi-Matsuyama algorithm. As done in the WDM unicast case, we allowed rerouting of existing streams only when doing so would allow the network to accept a session that it would otherwise block. For the baseline case, we do not impose the constraint that the network must remain strongly connected.

Performance Measures: The main performance measure is the blocking probability. Other performance measures, such as average time between reconfigurations, algorithm run time and average cost/delay of the established routes were also obtained.

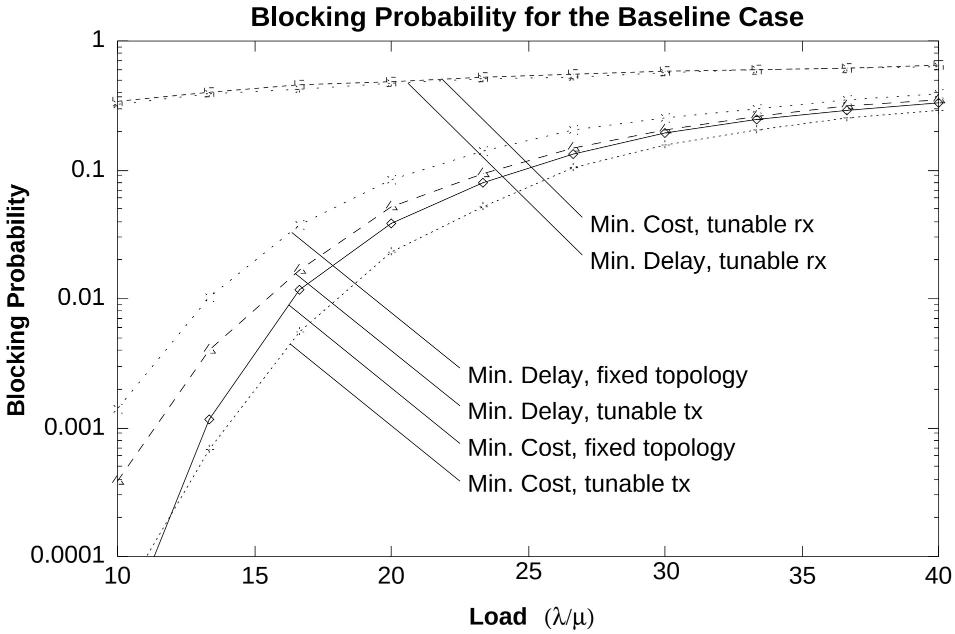


Figure 18: Blocking probability for the baseline case: single-multicast sessions, stream bandwidth 10% of the link capacity, 50,000 routes/point

The blocking probability as a function of the session arrival rate in the baseline case is shown in figure 18, for the various algorithms and network scenarios. The main conclusions from that figure are:

- Networks where physical multicast is allowed present much higher blocking than networks where it is not allowed. For example, when $\lambda/\mu = 10$, the blocking probability is under 0.2% for the fixed-topology networks and for the networks with no physical multicast, while it is about 30% for networks with physical multicast. For such networks, there is basically no difference between minimum-cost and minimum-delay algorithms. The main reason for this result is that in the baseline case, the number of transmitters and receivers in the network is the same; therefore, if two receivers are tuned to the same transmitter, there is a transmitter in the network that it not connected and cannot be used. The gain in flexibility does not make up for the lost capacity.

- Minimum-cost algorithms perform better than minimum-delay algorithms, as expected. Also, for the same kind of algorithm (i.e., minimum cost or minimum delay), the WDM networks perform better than the fixed-topology networks. However, it is interesting to note that the blocking probability for the fixed-topology under minimum-cost routing is *lower* than the blocking probability of the WDM network under minimum-delay routing.

The plots in Figure 18 have shown that the blocking probability for the networks where physical multicast is allowed is much higher than when it is not. This seems counter-intuitive: the networks with physical multicast have an *additional* degree of freedom, therefore they should have "better" performance. Figure 19 shows where this better performance is seen: in the cost and delay of accepted sessions. Minimizing the blocking probability is not the objective of minimum cost/delay algorithms. The two plots in Figure 19 show clearly that the lowest costs and delays are achieved in the networks where physical multicast is allowed. However, this does not mean at all that blocking probability will be low! We should point out that, while in a traditional (fixed-topology) network the cost measure has a well defined meaning (usage of network resources), in the optical network this meaning is lost, because it does not capture the fact that the tunability of transmitters and/or receivers is a resource that can be used (and "spent"). The delay measure, however, has still the same meaning.

Figure 20 shows the blocking probability as a function of the number of destinations for various load values. The routing algorithm is the minimum cost heuristic, with tunable

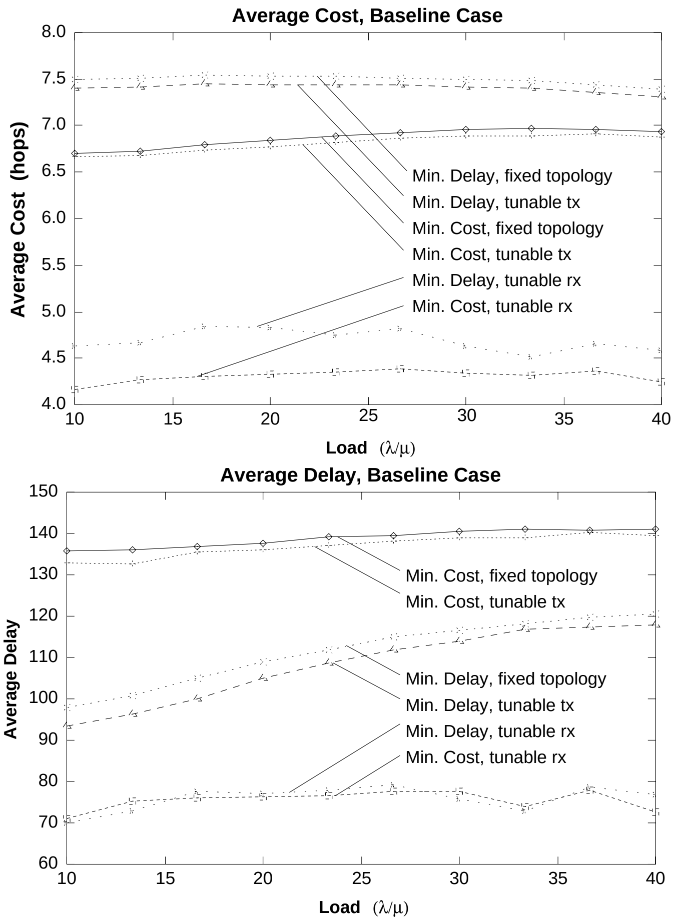


Figure 19: Comparison of costs and delays for established sessions in the baseline case

transmitters (no physical multicast). The figure indicates that the blocking probability is a weak function of the number of destinations[3]. The same result holds for the other algorithms and network scenarios in the baseline case.

Figure 21 shows the average time between stream re-routes as a function of the session arrival rate in the baseline case. The time is normalized to the session duration, i.e., an average re-route time of 10 indicates that the average time between re-routing of established streams is 10 times the average session duration (or, on the average, only one session in 10 will suffer rerouting in its lifetime). The figure indicates that only a small fraction of the streams will be re-routed during their lifetimes, for reasonable traffic loads. Moreover, it is very unlikely that a stream will will be rerouted more than once. Re-routing is more severe in

---

[3] In [28, 29] we have shown that this also happens with fixed-topology networks.

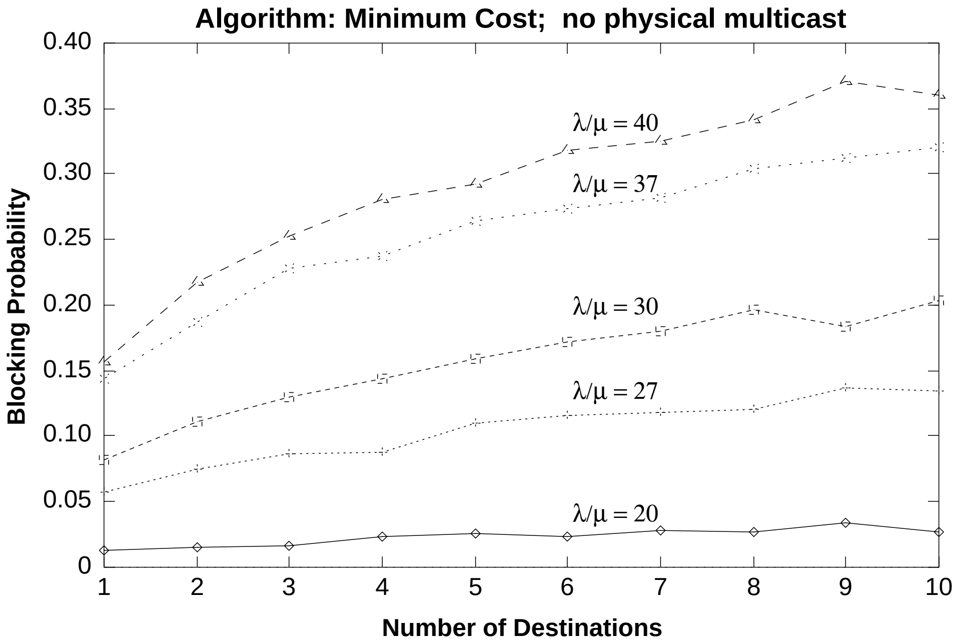


Figure 20: Blocking probability as a function of the number of destinations; baseline case, no physical multicast, minimum cost

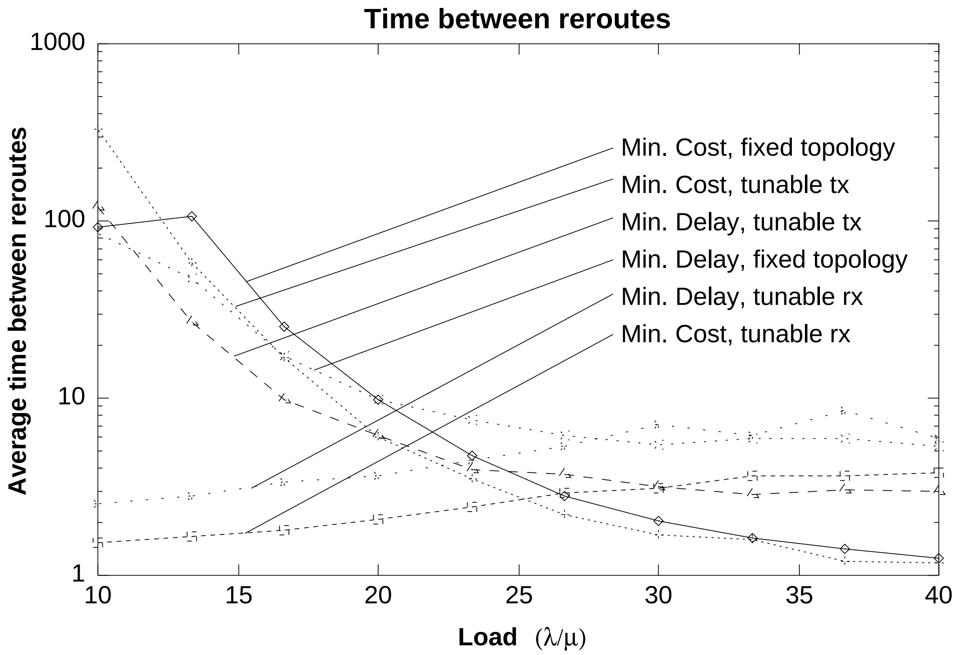


Figure 21: Average time between stream re-routes, baseline case

networks where physical multicast is allowed, due to the high blocking. We also determined that the average path change when a stream is re-routed is very small; the average change in cost is under 0.2 hops for the topologies with no physical multicast, and under 0.6 hops for the topologies with physical multicast.

7.2 Changing the Network Size

In this section, we investigate the effects of increasing the network size. The network size can be increased by: (i) increasing the number of nodes, and/or (ii) increasing the number of transmitters and receivers per node. We consider the following two variations to the baseline case:

- Increase the number of transmitters and receivers per node to 3, keeping the number of nodes fixed at 12.
- Increase the number of nodes from 12 to 20, keeping the number of transmitters and receivers per node fixed at 2.

We keep the traffic model the same as in the baseline case.

Increasing the Number of Transmitters/Receivers

Figure 22 shows the blocking probability as a function of the session arrival rate for a 12-node network, with 3 transmitters and 3 receivers per node. We note that the difference in performance between the fixed-topology network and the WDM network has decreased considerably. This is explained by the fact that the increased node degree leads to a richer topology, with shorter paths; the additional degree of freedom of rearranging links in the WDM network is less important in this scenario. Other observations:

- Re-routing happens less frequently than in the baseline case.
- Costs (in hops) are approximately the same as the baseline case.
- For higher loads, the blocking probability is more sensitive to the number of destinations.

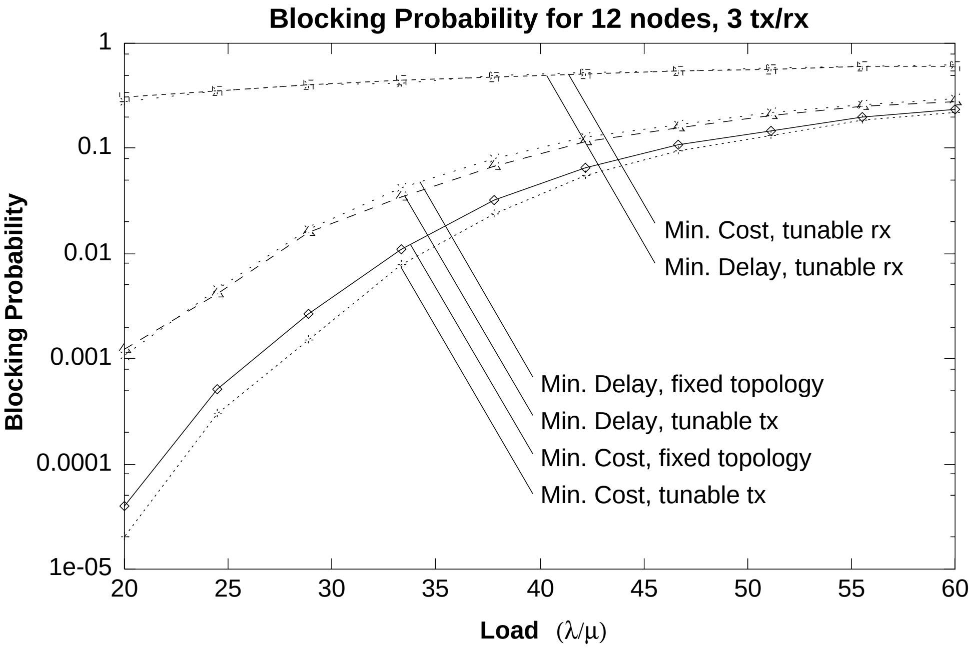


Figure 22: Blocking probability for 12-node networks, 3 transmitters/receivers per node, 50,000 routes/point

Increasing the Number of Nodes

Figure 23 shows the blocking probability as a function of the session arrival rate for a 20-node network, with 2 transmitters and receivers per node, under the same traffic as the baseline case. The same qualitative comments about the algorithms made in the baseline case can be repeated here. We note that the advantage of the WDM network over the fixed-topology network has increased. Other observations:

- Since the paths get "longer", the average change in cost can be as high as 1 hop for the networks where physical multicast is allowed, and 0.6 hops where it is not.
- For higher loads, the blocking probability is more sensitive to the number of destinations. For example, at $\lambda/\mu = 60$, using the minimum cost algorithm with no physical multicast (tunable transmitters), the blocking probability is about 20% for unicasts, and increases to about 45% for 10-destination multicasts.

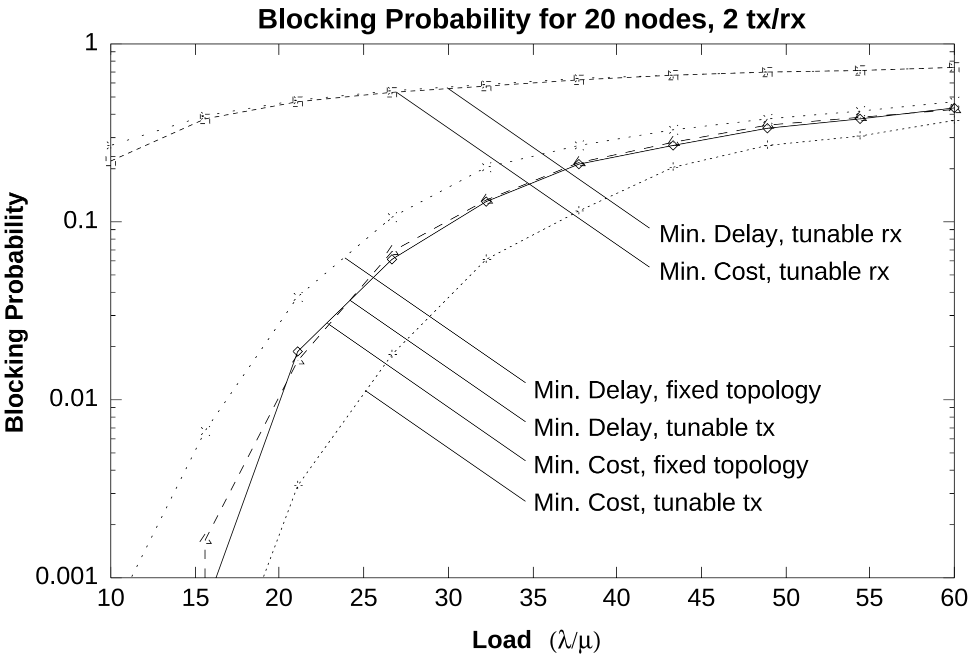


Figure 23: Blocking probability for 20-node networks, 2 transmitters/receivers per node, 15,000 routes/point

7.3 Keeping the Network Connected

In the runs for the baseline case, we did not require the network to stay strongly connected as its topology changes. In this section, we impose the additional constraint that at every invocation of the Shortest Path with Reconfiguration Algorithm, the resulting network is tested and, if necessary, a secondary reconfiguration is implemented to keep connectivity, as described in the Appendix. The blocking probability results are shown in Figure 24. The results for fixed-topology networks are the same as in the baseline case and are repeated here for the sake of comparison.

The main observation from Figure 24 is that while there is basically no change in the results for networks where physical multicast is not possible, there is a significant improvement for networks where it is used. This is made clearer in figure 25, where we compare the blocking probability for the minimum-cost heuristics in the connected and not connected cases.

Other observations include:

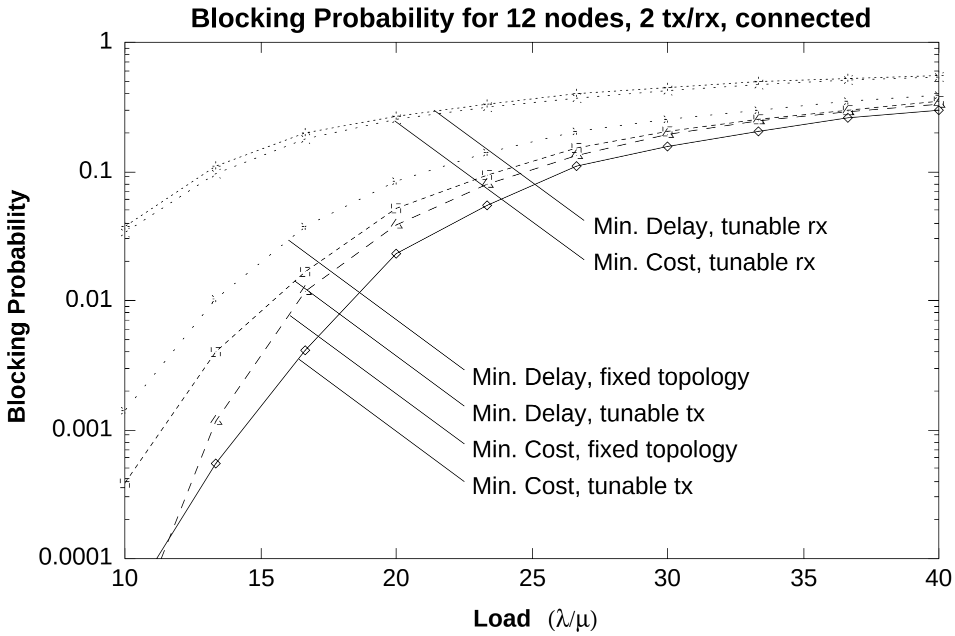


Figure 24: Blocking probability for 12-node networks, 2 transmitters/receivers per node; strong connectivity maintained, 50,000 routes/point

- For networks allowing physical multicast, keeping the topology connected increases the average time between re-routes and decreases the change in cost, especially for minimum-delay algorithms.

- Keeping the network connected increases the cost/delay obtained by the minimum cost/delay heuristics.

7.4 Changing the Number of Receivers

In the previous sections, we have shown that allowing physical multicast in the WDM network when the number of transmitters and receivers is equal greatly degrades the performance. Keeping the network connected offsets some of this degradation, but it is still severe. However, we expect that in a network where receivers are abundant (i.e., more receivers than transmitters), allowing physical multicast should improve the performance.

To investigate this fact, we computed the blocking probability for a number of 12-node

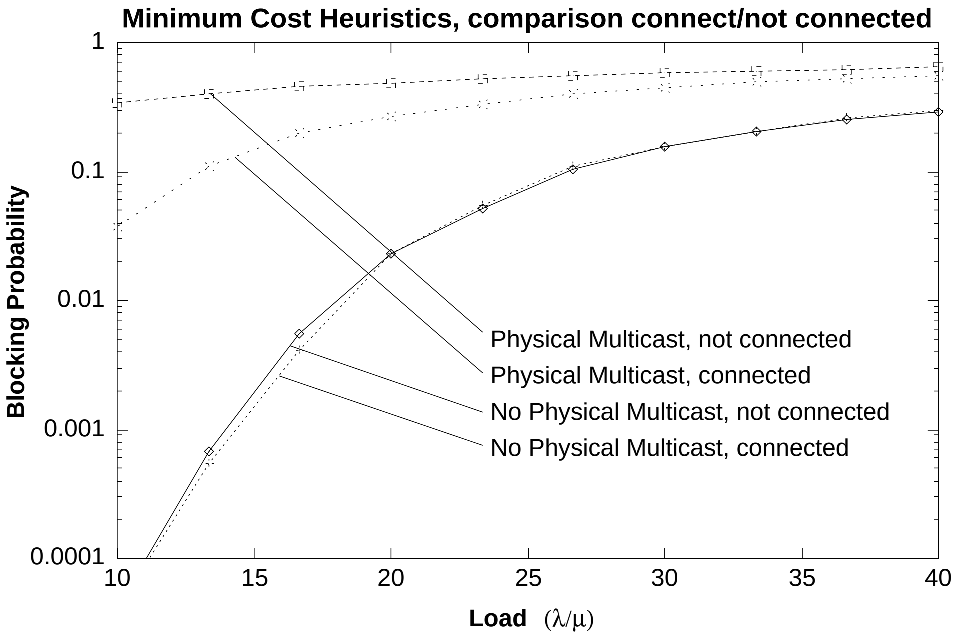


Figure 25: Comparison of blocking probability for minimum cost heuristics, when the network is kept connected versus when it is not

networks, keeping the number of transmitters per node fixed at 2, and varying the number of receivers per node from 2 to 8. The session arrival rate was kept fixed, and no connectivity constraints were imposed in the network. The blocking probability results (as a function of the number of receivers) are shown in figure 26. The baseline case corresponds to 2 receivers per node.

Figure 26 shows that, as expected, networks where physical multicast is not possible cannot make effective use of extra receivers. When physical multicast is allowed, adding receivers will improve the performance. For this particular case, the point where the performance of the two networks becomes the same is between 5 and 6 receivers. For higher loads, the point of equal performance shifts to a lower number (for example, for $\lambda/\mu = 40$, the crossover point is between 4 and 5 receivers).

The results shown in figure 26 were obtained considering: (i) the network topology was not constrained to be connected at all times, and (ii) receiver/transmitter costs were assigned as in the baseline case (i.e., all costs set to 0.5). We know that keeping the network topology

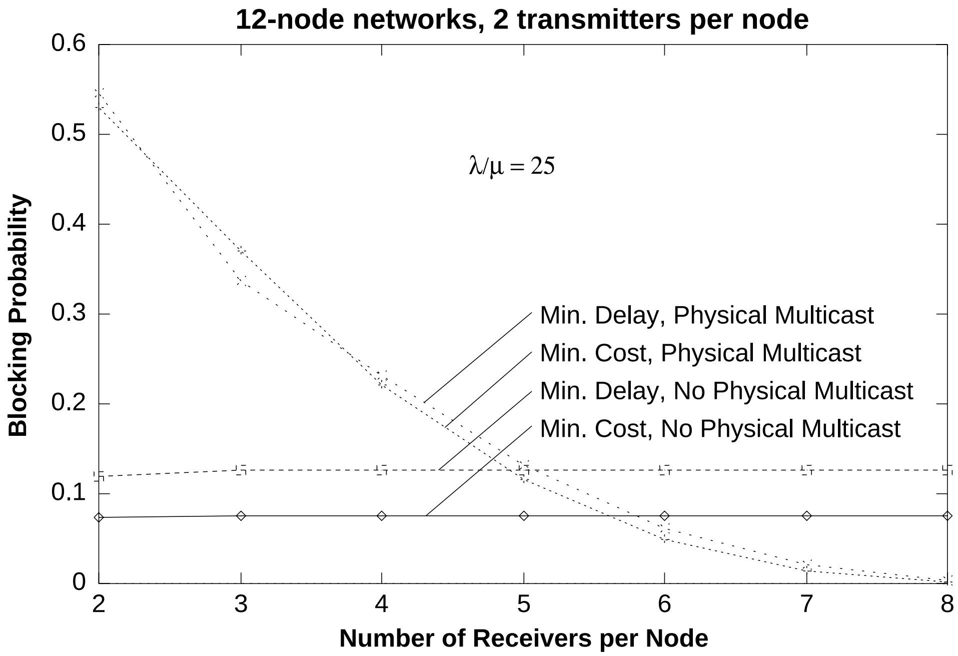


Figure 26: Increasing the number of receivers per node, keeping the number of transmitters fixed, 20,000 routes/point

connected improves the performance when physical multicast is allowed, and we would like to investigate the role played by the link costs in the performance.

Figure 27 contains two plots, one for connected topologies, and the other for topologies that are not required to stay connected. Both plots contain the blocking probability as a function of the number of receivers in the network, for the minimum-cost heuristic on a network allowing physical multicast; the number of transmitters was fixed at 2. Each plot contains three curves, corresponding to the following transmitter/receiver cost assignments:

- Receiver cost 0.5, transmitter cost 0.5 (this is the assignment used in the evaluation so far).
- Receiver cost 1.0, transmitter cost 0.0.
- Receiver cost 0.0, transmitter cost 1.0.

The results from figure 27 indicate that when the network is kept connected, there is little sensitivity to the way the link cost is split between the transmitters and receivers. However, when the network is not restricted to connected topologies, the best assignment is to assign all the cost to the receivers (making it "expensive" to employ the physical multicasting).

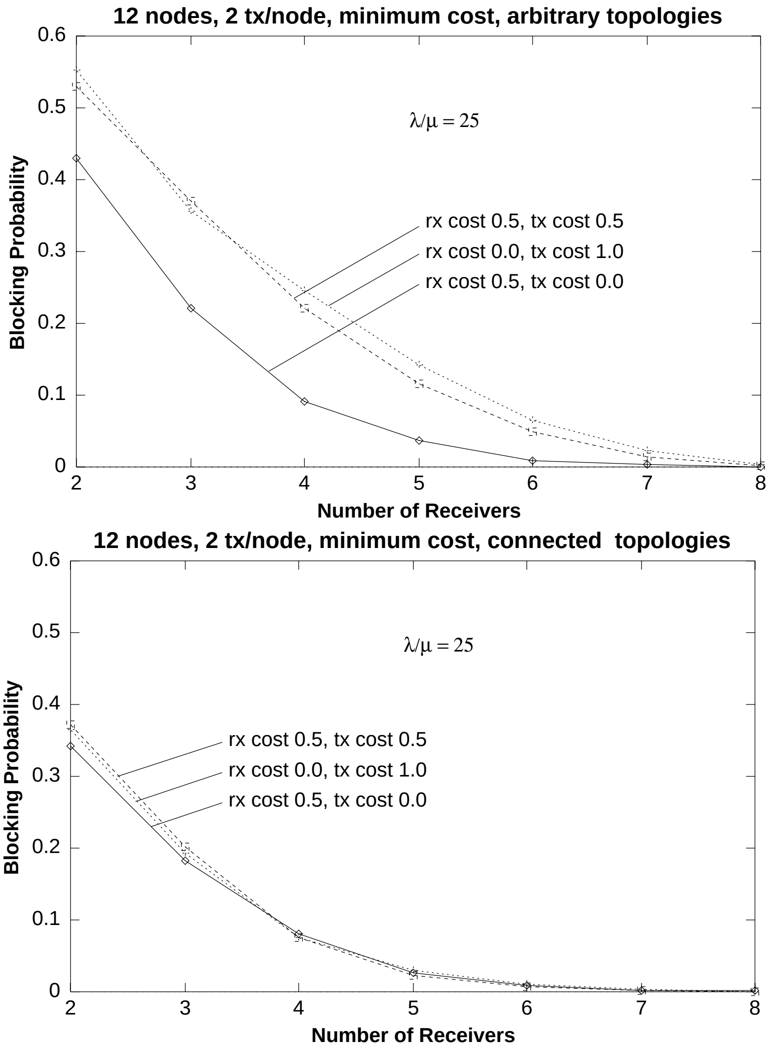


Figure 27: Effect of changing the transmitter/receiver costs, minimum cost algorithm, 20,000 routes/point

If physical multicast is not allowed, the minimum-cost algorithm is not sensitive to the assignment of costs.

8 Conclusions

In this report, we considered the problem of routing multimedia streams in a WDM network, taking into account their requirements of bandwidth, multipoint communications, and latency, and making use of the additional degree of freedom (the topology) of the network. We presented an exact solution for the optimum routing and reconfiguration problem, based on linear integer programming. Since this solution is complex (and has exponential worst-case run time because the problem is NP-complete), we also proposed heuristic algorithms to find sub-optimal solutions, and derived an upper bound in performance. For unicast traffic, we showed that the heuristic solution is close to the upper bound, which obviates the need to pursue the more complex exact solution. Finally, we evaluated the performance of the WDM network in a realistic environment, and showed that it compares favorably with a centralized electronic switch of equivalent complexity.

The main conclusion from the evaluation of multicast routing algorithms in the WDM network concerns the use of physical multicasting in a network where receivers are tunable. The unrestricted use of physical multicasting can decrease the cost and delay of accepted sessions, but at the price of a much higher blocking probability. The previous sections have shown that, by keeping the network connected and/or by properly assigning link costs, the penalty paid in performance can be reduced. However, the blocking probability is still much higher than in the case where no physical multicasting is allowed, except when the number of receivers is large. The fundamental reason for this behavior is simply that, by allowing physical multicasting, the network can be re-arranged in a topology where all receivers are in use, but some transmitters are not connected to any receiver. This represents a net *decrease* in network capacity, and leads to increased blocking probability. Keeping the network strongly connected and/or shifting the cost to the receivers improve the performance because less transmitters are allowed to remain disconnected. By the same token, when there are plenty of receivers, it is unlikely that transmitters will be disconnected. In summary:

- Networks where physical multicasting is allowed are able to make use of extra receivers as they are added; however, performance with a small number of receivers per node is very low; techniques such as keeping the network strongly connected and/or properly assigning transmitter/receiver costs can help, but not much.

- Networks where physical multicasting is not allowed have much better performance when the number of receivers is small, but cannot make use of additional receivers.

These observations indicate that the best way to use physical multicasting is:

1. The network should be kept strongly connected at all times.
2. At all times, there should be *at least one* receiver connected to each transmitter.
3. Physical multicasting can be used as long as it does not violate rules 1 and 2 above. Note that this is true regardless of the type of traffic (unicast or multicast) in the network. Even unicast traffic can benefit from the extra flexibility of allowing physical multicasting, as long as rules 1 and 2 are satisfied.

By using the above rules, we can guarantee good performance both when the number of receivers is small and when it is large. Note that, if the number of receivers is equal to the number of transmitters, no physical multicasting will be allowed.

Based on the results of this report, we derive the following set of guidelines for designers/implementors of WDM networks:

- If the network has tunable transmitters and fixed receivers, the total number of transmitters and receivers in the network should be the same. To upgrade a node's capacity, one has to add transmitters and receivers to it in pairs.
- It is recommended that the network be built with tunable receivers and fixed transmitters. In this case, a node can be upgraded just by adding receivers. When operating the network, physical multicasting should be allowed only if there are more receivers than transmitters, as described above.

- Under multicast traffic, the minimum cost heuristics achieve lower blocking. It remains to be seen if delay is a problem.

Appendix
Formal Description of The Shortest Path with Reconfiguration Algorithms

In this Appendix, we describe the basic Shortest Path with Reconfiguration algorithm, an extension to Dijkstra's shortest path algorithm for reconfigurable networks, and the secondary reconfiguration algorithm, which is used to keep the network strongly-connected.

A The Basic Shortest Path with Reconfiguration Algorithm

The algorithm described in this section finds the shortest path between two nodes in a reconfigurable optical network. Some transmitters and receivers in the network might already be tuned due to other communications in progress; these cannot be reconfigured, but they might be used in whatever topology they happen to be. The algorithm has two degrees of freedom: the path, and the topology, potentially reconfiguring the transmitters and receivers that are idle. A numeric *label* is assigned to each transmitter and receiver in the network. If a transmitter with label l^t is tuned to a receiver with label l^r, an unidirectional link with label $l^t + l^r$ is created. The shortest path minimizes the sum of the labels in the path. The label can represent, for example, the propagation delay; in this case, if the network topology is a star, the label associated with the transmitter is the propagation delay from the node to the star, and the label associated with the receiver is the propagation delay from the star to the node.

INPUTS:

- A WDM network with N nodes; node i has S_i optical transmitters and P_i optical receivers, $i = 1, \ldots, N$. The optical transmitters and receivers are either *locked* or *free*; the locked transmitters and receivers are tuned, forming a given topology, and the free ones are not connected at all and can be reconfigured. If the network supports physical multicasting, *all* transmitters are to be considered free.

- A set of nonnegative transmitter labels $\{l^t_{ij}\}$, $i = 1, \ldots, N$, $j = 1, \ldots, S_i$.
- A set of nonnegative receiver labels $\{l^r_{ij}\}$, $i = 1, \ldots, N$, $j = 1, \ldots, P_i$.
- A source node s and a destination node d.

OUTPUT: The path from s to d with the minimum length (sum of the transmitter and receiver labels in the path), using, if necessary, free transmitters and receivers.

ALGORITHM:

Step 1: For all transmitters in the network, do the following: if transmitter j in node i is free, add a *transmitter virtual node* V^t_{ij} to the network topology, and a link (using transmitter j) from node i to node V^t_{ij}, with label l^t_{ij}.

Step 2: For all receivers in the network, do the following: if receiver j in node i is free, add a *receiver virtual node* V^r_{ij} to the network topology, and a link (using receiver j) from node V^r_{ij} to node i, with label l^r_{ij}.

Step 3: Create a set of nodes P, initially empty, and a set of nodes T, initially containing all the nodes in the network. For each node in the network, associate a label l; initially, assign $l(s) = 0$, $l(i) = \infty$ for $i \neq s$. Create also two sets of nodes A and B, initially empty.

Step 4: Let k be the node with the smallest $l(k)$ (if there are many, choose one at random). If $l(k) = \infty$, there is no path between s and d using the locked receivers and transmitters; go to step 7. Otherwise, move k from the set T to the set P. If node k is a virtual node and the set A is empty, also put node k in A.

Step 5: If k is not a virtual node, update the node labels as follows: $\forall j \in P$, if there is a link from node k to node j, make $l(j) \leftarrow \min\{l(j), l(i) + l_{kj}\}$, where l_{kj} is the label of the link between k and j. If k is a virtual node, there is no need to update the node labels.

Step 6: If $k \neq d$, return to step 4; otherwise, the shortest path using the locked transmitters and receivers has been found; proceed to step 7.

Step 7: Create a set of nodes P', initially empty, and a set of nodes T', initially containing all the nodes in the network. For each node in the network, associate a label l'; initially, assign $l'(d) = 0$, $l'(i) = \infty$ for $i \neq d$.

Step 8: Let k be the node with the smallest $l'(k)$ (if there are many, choose one at random). If $l'(k) = \infty$, go to step 10. Otherwise, move k from the set T' to the set P'. If $k = s$, go to step 10. If node k is a virtual node and the set B is empty, put node k in B and go to step 10.

Step 9: If k is not a virtual node, update the node labels as follows: $\forall j \in P'$, if there is a link from node j to node k, make $l'(j) \leftarrow \min\{l'(j), l'(i) + l_{jk}\}$. Go to step 8.

Step 10: If the set A or the set B or both are empty, terminate. The shortest path is the one found in step 6, if any. If both A and B are non-empty, let us denote by a the node in A and by b the node in B. Let $L_1 = l(d)$, $L_a = l(a)$ and $L_b = l'(b)$. If $L_1 \leq L_a + L_b$, the path found in step 6 is the shortest and there is no need to reconfigure the network. Otherwise, a shorter path can be created by connecting the virtual nodes a and b. If a corresponds to the virtual node V^t_{ij} and b corresponds to the virtual node V^r_{kl}, then transmitter j in node i must be tuned to receiver l in node k, and the shortest path will be the path from s to i, the newly-created i-k link, and the path from k to d.

B The Secondary Reconfiguration Heuristic

For control purposes, it is necessary to keep the WDM network strongly connected at all times, so that all the nodes can exchange control messages. The transmission of control messages can be done by flooding, as in the OSPF routing protocol [30].

In the basic shortest path with reconfiguration algorithm presented in section A, the free transmitters and receivers are considered to be idle. In general, however, they will be connected in a certain topology, to keep the network strongly connected. When the shortest path algorithm described in section A is applied, it might change the network topology, and the resulting topology can potentially be disconnected. We denote the reconfiguration from the shortest path algorithm as the *primary reconfiguration*. The heuristic in this section will try to identify a *secondary reconfiguration* to keep the network strongly connected.

As indicated above, the network will have as many free transmitters and receivers connected as possible (but available for reconfiguration). If the total number of transmitters in the network is equal to the total number of receivers, then all free transmitters and receivers can be used. Reconfiguring the network so that a transmitter and a receiver can be connected might also include reconfiguring the previous connections held by them, as illustrated in figure 28.

If the primary reconfiguration causes the network to become disconnected, then this new topology has at most two disconnected subsets; a way to implement the secondary reconfiguration would be to identify two good candidate nodes to "bridge" the two subsets. Therefore, we need an algorithm that can:

1. Test a topology for connectivity.
2. If the topology is disconnected, locate the appropriate nodes where the "bridging" between the subsets can be performed.

Both of these functions can be efficiently performed by a simplified version of the Floyd-Warshall shortest path algorithm [31]. In the the next two sections, we describe the simplified Floyd-Warshall algorithm and give the secondary reconfiguration heuristic.

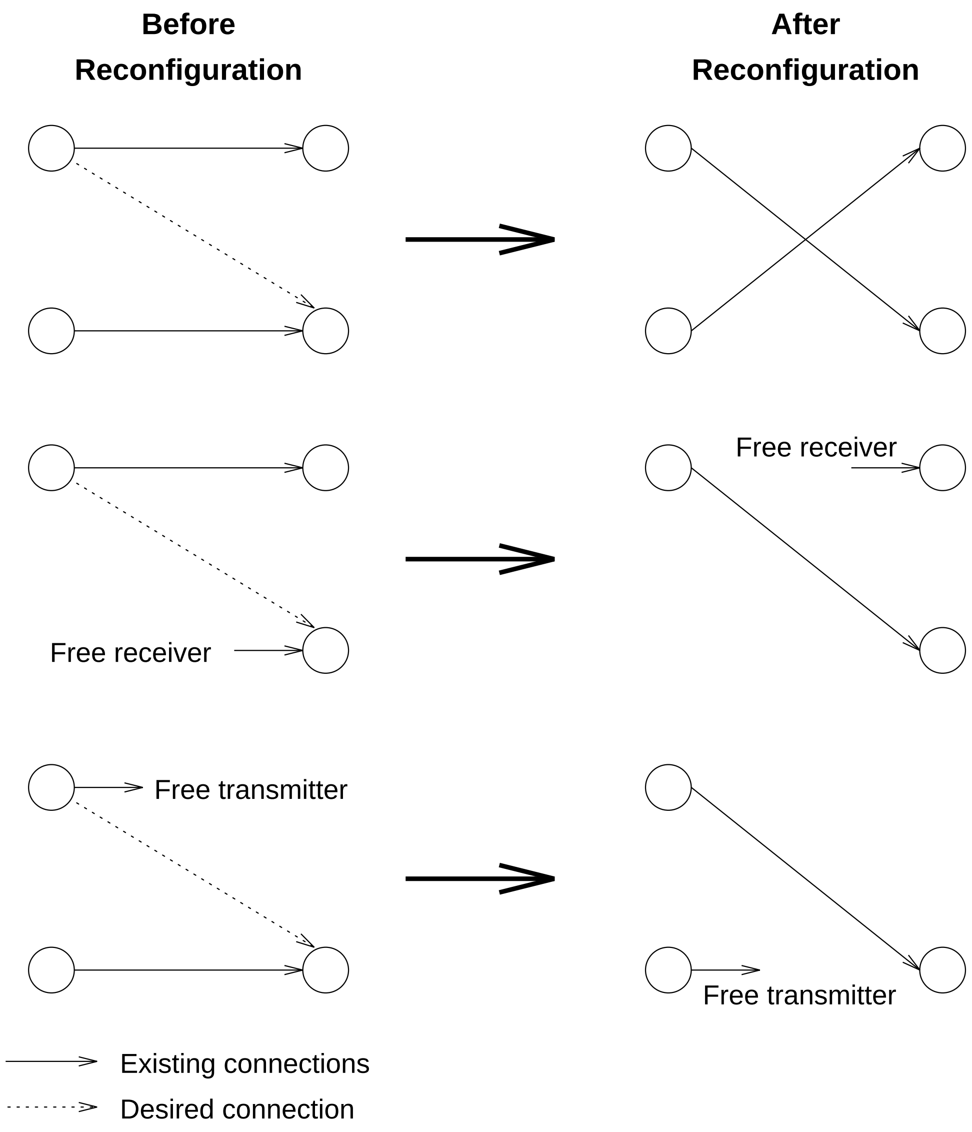


Figure 28: Reconfiguring the network

B.1 The Simplified Floyd-Warshall Algorithm

INPUT:	The network topology (N is the number of nodes).
OUTPUT:	The connectivity matrix $\boldsymbol{D}$, where $D_{ij} = 1$ if there is at least one path from node i to node j, and $D_{ij} = 0$ otherwise.

ALGORITHM:

Step 1: Define the matrix $\boldsymbol{D}(0)$ as follows:

$$D_{ij}(0) = \begin{cases} 1 & \text{if node } i \text{ is connected to node } j\text{, or } i = j \\ 0 & \text{otherwise} \end{cases}$$

Step 2: Calculate the matrix $\boldsymbol{D}(k)$ from the matrix $\boldsymbol{D}(k-1)$ as follows:

$$D_{ij}(k) = \begin{cases} 1 & \text{if } D_{ik}(k-1) \textbf{ and } D_{kj}(k-1) \text{ are both } 1 \\ D_{ij}(k-1) & \text{otherwise} \end{cases}$$

Step 3: Repeat step 2 for $k = 1, \ldots, N$; matrix $\boldsymbol{D}(N)$ is the desired connectivity matrix $\boldsymbol{D}$.

B.2 Description of the Secondary Reconfiguration Heuristic

The secondary reconfiguration heuristic is executed as a part of the shortest path with reconfiguration algorithm in section A. If that algorithm decides on a primary reconfiguration, the secondary reconfiguration algorithm is invoked; if it fails, the primary reconfiguration is not permitted. More specifically, in step 10, if $L_1 > L_a + L_b$, the secondary reconfiguration algorithm is executed. If it is successful, then the shortest path algorithm proceeds as originally described, possibly with a secondary reconfiguration in place. If it fails, then the primary reconfiguration is dropped; the final path between the source and destination will be the path found using the current topology (or no path at all). Note that this is not optimal in any sense; one could elect to search for other reconfigurations besides the primary that would lead to longer paths but would not partition the network.

The secondary reconfiguration algorithm is:

INPUTS:	The network topology, the list of locked and free transmitters and receivers, and the primary reconfiguration.
OUTPUTS:	A flag, indicating failure or success; in case of success, the algorithm may recommend a secondary reconfiguration.

ALGORITHM:

- Step 1: Execute the primary reconfiguration in the network topology given and use the simplified Floyd-Warshall algorithm to compute the connectivity matrix $\boldsymbol{D}$. If all the entries in $\boldsymbol{D}$ are 1, return and indicate success; the new topology is connected and there is no need for a secondary reconfiguration. Otherwise, mark the transmitter and receiver used in the primary reconfiguration as locked and proceed to step 2.

- Step 2: Search the $\boldsymbol{D}$ matrix for an entry d_{ij} such that $d_{ij} = 0$, node i has a free transmitter and node j has a free receiver. If the WDM network supports physical multicasting, consider all transmitters as free.

- Step 3: Reconfigure the network so that a link between nodes i and j is created (exchanging any other connection as depicted in figure 28). Compute the new connectivity matrix $\boldsymbol{D}'$.

- Step 4: If all the entries in $\boldsymbol{D}'$ are 1, stop. The secondary reconfiguration has been found. Return the secondary reconfiguration and indicate success. Otherwise, return to step 2 to search for another entry. If all zero entries in $\boldsymbol{D}$ have been considered already, terminate and indicate failure.

References

[1] L. Kazovsky et al, “WDM Local Area Networks,” *IEEE LTS*, vol. 3, no. 2, pp 8-15, May 1992.

[2] W. B. Jones, *Introduction to Optical Fiber Communication Systems*, Holt, Rinehart and Winston, New York, 1988.

[3] Y. Birk et al, “Bus-Oriented Interconnection Topologies for Single-Hop Communication among Multi-Transceiver Stations,” *IEEE INFOCOM '88*, New Orleans, LA, March 1988, pp 558-67.

[4] M. Mehdi Nassehi et al, “Fiber Optic Configurations for Local Area Networks,” *IEEE Journal on Selected Areas in Communications*, Nov. 1985, vol. 3, no. 6, pp 941-9.

[5] K. Kobayashi and I. Mito, “Single Frequency and Tunable Laser Diodes,” *Journal of Lightwave Tech.*, vol. LT-6, no. 11, Nov. 1988, pp 1623-33.

[6] T. P. Lee and C. E. Zah, “Wavelength-Tunable and Single-Frequency Semiconductor Lasers for Photonic Communications Networks,” *IEEE Communications Mag.*, Oct. 1989, pp 42-51.

[7] D.A. Smith et al, “Integrated-Optic Acoustically-Tunable Filters for WDM Networks,” *IEEE JSAC*, vol. 8, no. 6, Aug. 1990, pp 1151-1159.

[8] H. Kobrinski and K.-W. Cheung, “Wavelength-Tunable Optical Filters: Applications and Technologies,” *IEEE Communications Mag.*, Oct. 1989, pp 53-63.

[9] B. Mukherjee, “WDM-based local lightwave networks. I. Single-hop systems,” *IEEE NETWORK*, vol.6, no.3, May 1992, pp 12-27.

[10] B. Mukherjee, “WDM-based local lightwave networks. II. Multihop systems,” *IEEE NETWORK*, vol.6, no.4, July 1992, pp 20-32.

[11] I. Habbab, M. Kavehrad, C. Sundberg, "Protocols for Very High-Speed Optical Fiber Local Area Networks Using a Passive Star Topology," *IEEE Journal on Lightwave Tech.*, vol. LT-5, no. 12, Dec. 1987, pp 1782-1793.

[12] N, Mehravari, "Improved Multiple Access Schemes for Very High-Speed Optical Fiber Local Area Networks Using a Passive Star Topology," *IEEE GLOBECOM 1989*, Dallas, Nov. 27-30, 1989.

[13] I. Chlamtac and A. Ganz, "Design Alternatives of Asynchronous WDM Star Networks," *IEEE ICC 1989*, Boston, June 11-14, 1989.

[14] A. Ganz and Z. Koren, "WDM Passive Star - Protocols and Performance Analysis," *IEEE INFOCOM 1991*, Bal Harbour, Florida, April 9-11, 1991.

[15] N. R. Dono et al., "A wavelength division multiple access network for computer communication," *IEEE JSAC*, vol. 8, no. 6, pp. 983-994, Aug. 1990.

[16] A.S. Acampora, M.J. Karol, M.G. Hluchyj, "Terabit Lightwave Networks: The Multihop Approach," *AT&T Technical Journal*, vol. 66, issue 6, Nov/Dec 1987, pp 21-34.

[17] J. Bannister, L. Fratta, M. Gerla, "Topological Design of the Wavelength-Division Optical Network," *IEEE INFOCOM 1990*, San Francisco, June 3-7, 1990.

[18] J. Bannister, L. Fratta, M. Gerla, "Routing in Large Metropolitan Area Networks Based on Wavelength-Division Multiplexing Technology," in *High-Capacity Local and Metropolitan Area Networks - Architecture and Performance Issues*, edited by G. Pujolle, Springer-Verlag, 1990.

[19] J.-F. Labourdette and A. Acampora, "Logically Rearrangeable Multihop Lightwave Networks," *IEEE Trans. on Communications*, vol.39, no.8, pp 1223-30, Aug. 1991.

[20] J.-F. Labourdette and A. Acampora, "Partially Reconfigurable Multihop Lightwave Networks," *IEEE GLOBECOM 1990*, San Diego, Dec. 2-5, 1990.

[21] C. Noronha and F. Tobagi, "Optimum Routing of Multicast Streams," *IEEE INFOCOM '94*, Toronto, Canada, June 1994, pp 865-73.

[22] C. Noronha and F. Tobagi, "Optimum Routing of Multicast Audio and Video Streams in Communications Networks," *Technical Report CSL-TR-94-618*, Computer Systems Laboratory, Stanford University, April 1994.

[23] F.S. Hillier and G.J. Lieberman, *Introduction to Operations Research*, 5th ed., New York, McGraw-Hill, 1990.

[24] H. Crowder, E. L. Johnson, M. Padberg, "Solving Large-Scale Zero-One Linear Programming Problems," *Operations Research*, Vol. 31, No. 5, Sept-Oct 1983, pp 803-34.

[25] M. S. Bazaraa, J. J. Jarvis and H. D. Sherali, *Linear Programming and Network Flows, 2nd ed.*, John Wiley & Sons, New York, 1990.

[26] S. Kirkpatrick, C. Gelatt, M. Vecchi, "Optimization by simulated annealing," *Science*, 220(4598):671-680, May 1983.

[27] J. Bannister and M. Gerla, "Design of the Wavelength-Division Optical Network," *IEEE ICC'90*, Atlanta, April 16-19, 1990.

[28] C. Noronha and F. Tobagi, "Evaluation of Multicast Routing Algorithms for Multimedia Streams," *IEEE International Telecommunications Symposium*, Rio de Janeiro, Brazil, August 1994.

[29] C. Noronha and F. Tobagi, "Evaluation of Multicast Routing Algorithms for Multimedia Streams," *Technical Report CSL-TR-94-619*, Computer Systems Laboratory, Stanford University, April 1994.

[30] J. Moy, "OSPF Version 2," RFC 1583, Proteon, Inc., March 1994.

[31] Bertsekas, D., and Gallager, R., *Data Networks*, Prentice-Hall, New Jersey, 1987.