Design Principles for Multi-Hop Wavelength and Time Division Multiplexed Optical Passive Star Networks*

Sheau-Ru Tong and David H.C. Du

Computer Science Department
University of Minnesota
200 Union St. S.E., Minneapolis, MN, 55455.

February 5, 1993

Abstract

Wavelength Division Multiplexing (WDM) has been shown to be one of the most promising ways to exploit the enormous bandwidth of a single mode optical fiber. Networks employing WDM are commonly based on *optical passive star* couplers. The major cost and limitations of such networks lie in the interface (both transmitters and receivers) providing optical-electronic conversion between stations and communication media. According to current technology, an interface based on fast tunable wavelength devices is still in an infant stage of development. On the contrary, fixed wavelength devices are much cheaper and already commercially available. When fixed wavelength devices are used, to reduce the number of transmitters and receivers in each station several stations may transmit and receive on the same wavelength. Therefore, *Time-Division Multiplexing (TDM)* is employed for those stations accessing the same wavelength. The resulting networks may require taking a multi-hop path to deliver packets from one station to another (i.e., *multi-hop WTDM* networks). In this paper we study multi-hop WTDM networks when each station has only one fixed wavelength transmitter and one fixed wavelength receiver. We propose a graph model called the *Receiving Graph model* to represent these networks such that their inherent properties can be easily understood and alternative designs can be compared. Furthermore, based on this model we discussed several design principles for such networks and some theoretical performance limitations are presented.

Key Words: Interconnection Networks, high-speed networks, Optical Passive Star, Wavelength and Time Division Multiplexing.

*Partially supported by a grand from Bellcore

1 Introduction

Emerging high bandwidth applications, such as voice/video services, distributed data bases, and network supercomputing, are driving the use of single-mode optical fibers as the communication media for the future [2][3][4] [26].

However, due to speed limits of electronic network access interfaces, the accessible bandwidth is far less than the bandwidth available in a single mode fiber. One solution is to use *Wavelength Division Multiplexing (WDM)*. The WDM scheme exploits the bandwidth available in an optical fiber by modulating different wavelengths of light in the electromagnetic spectrum to provide several channels of smaller bandwidth which match the speed of the electronic interfaces [5] [6][21]. User stations are tapped onto an optical fiber via optical transmitters and receivers. The transmission from one station to another is accomplished by tuning the receiver of the receiving station to the transmitter's wavelength of the sending station. This allows many concurrent transmissions, one on each different wavelength, to be performed simultaneously.

A physical star topology is frequently suggested for implementing an optical network in which optical fibers are interconnected via an optical passive star coupler [18][13] [17] [16] [15]. Figure 1 shows N stations connected via an optical passive star. Every transmitter broadcasts its signal to all the receivers with a splitting loss equal to $1/N$ introduced by the passive star coupler. As several different wavelength signals are broadcast simultaneously, a combined signal appears at all receivers. By tuning to an appropriate wavelength, receivers may extract a desired signal from any of the wavelengths. There are several favorable features of this architecture such as one-to-one and multicast connections can be easily implemented, there is no inner switch blocking, the signal attenuation is logarithmically increased with N, no required external power source for the passive star to guarantee reliability and to eliminate interference, and is without high hardware complexity for the switching fabric as in the electronic crossbar switch.

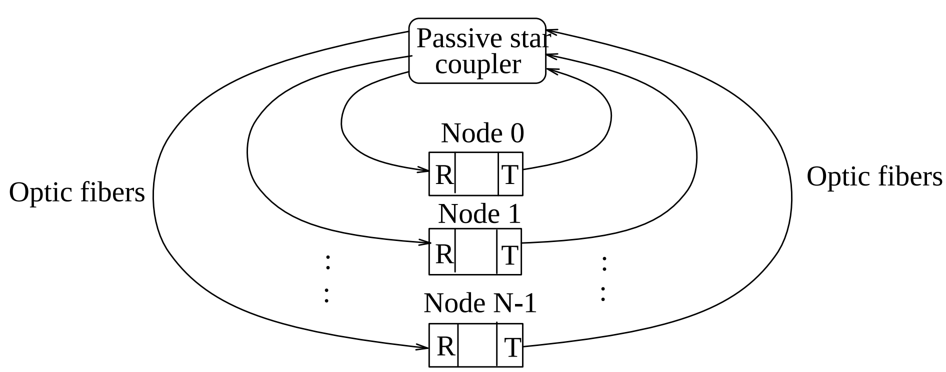


Figure 1: Physical star topology based on an optical passive star coupler

Before a transmission can take place, pre-coordination is required by a transmitter and receiver to determine the wavelength upon which they will tune. Several transmissions may share the same wavelength along the time domain, so a contention resolution scheme should be provided to guarantee contention-free transmissions. Several passive star networks have been proposed and are summarized in Table 1 according to the protocol type, the number of wavelengths, and the number of transmitters and receivers per station.

Networks	Protocols	# wave	# trans	# rcvr
LAMDANET[20]	Broadcast	N	1 fixed	N fixed
STAR-TRACK[22]	Control track	N	1 fixed	1 tunable
FOX (two stars)[7]	Slotted ALOHA	N	2 tunable	2 fixed
HYPASS (two stars)[8]	In-Buffered/ Out-control	N	1 fixed, 1 tunable	1 fixed, 1 tunable
Knockout Switch[23]	Knockout	$4N$	1 fixed	4 tunable
RAINBOW[6]	Receiver polling	N	1 fixed	1 tunable
ALOHA based [10]	Slotted ALOHA	$2N$	1 tunable	1 tunable
ALOHA based[15]	Slotted/ resrv ALOHA	$2N$	1 fixed, 1 tunable	1 fixed, 1 tunable
DT-WDMA[14]	Slotted ALOHA	N	2 fixed	1 fixed, 1 tunable
Swift[11]	WTDM	≥ 2	1 tunable	2 tunable
Perfect Shuffle[1]	WDM	pN	p fixed	p fixed
de Bruijn[24]	WDM	pN	p fixed	p fixed
(p, k)-ShuffleNet[9]	WTDM	N/p	1 fixed	1fixed
Bus-Mesh[18]	WTDM	$< \sqrt{N}$	1 fixed	1 fixed

Table 1: Requirements for the previously proposed N-station passive star networks.

LAMDANET[20] requires an array of N receivers at each station so all signals can be received simultaneously. STAR-TRACK[22] uses a separate electronic control track to avoid contention prior to transmission. FOX[7] was designed for interconnecting processors and memory modules and employs two passive stars, one for data transmission and the other for acknowledgment. In each star, a fast tunable transmitter and a fixed wavelength receiver are required by each station. Also, a slotted ALOHA contention resolution scheme is used. HYPASS[8] also requires two stars, one for data transmission and one for acknowledgment. In the star for data transmission, a fixed wavelength transmitter and a fast tunable receiver is used in each node. In the star for acknowledgment, a fast tunable transmitter and a fixed wavelength receiver are deployed for implementing an output control scheme to avoid contention. The Photonic Knockout Switch[23] uses an additional electronic network to resolve contention. RAINBOW[6] uses a method of simple receiver polling to avoid contention. Several alternatives employing ALOHA/Slotted ALOHA based protocols have been proposed[10][15][14]. All of them require either fast tunable transmitters or fast tunable receivers for data transmission. Some of them require an extra receiver-transmitter pair in each station to sense channel availability. With the increasing optical fiber bandwidth, the cost of pre-coordination and contention resolution becomes prohibitively high and should be avoided [12][11]. Furthermore, the success of the above schemes (except LAMDANET) depends on the availability of wide tuning-range high speed tunable transmitters and receivers. At the present time, most of the tunable devices are still expensive and in the infancy stage of development. Additionally, because the tuning range is inversely related to the tuning speed [5], the number of wavelengths can be tuned far less than required ($O(N)$) in many systems. On the contrary, fixed wavelength transmitters and receivers are much cheaper and stable. Since no tuning is required, they can be set to any wavelength within the whole low loss spectrum region (i.e., more wavelengths can be supported). Therefore, they are more suitable for a large scale network.

One way to remove the requirements of pre-coordination and contention resolution is to use time-division multiplexing on each wavelength. We call this type of protocol a *Wavelength- and Time-Division Multiplexed protocol (WTDM).* Swift, proposed in [11], uses a WTDM protocol to logically implement a completely connected topology and an adaptive multihop routing algorithm is incorporated to improve the performance under light load. However, each station still requires one fast tunable transmitter and two fast tunable receivers.

By using fixed wavelength devices, a packet sent out by a transmitter can only be received by a limited subset of stations whose receivers are set to the transmitter's wavelength. We say there is a *direct connection* between two stations if the transmitter's wavelength of one node equals the receiver's wavelength of the other. The pattern of interconnection forms a *connected topology* of the network. The communication between two stations may require going through several intermediate stations. We call this type of network a *multi-hop network*. Several multihop WDM networks have been proposed which use different regular connected topologies such as a re-circulating multistage Perfect Shuffle[1] or de Bruijn graph[24]. The use of regular connected topologies provides several advantages; including simple routing, predictable path length and enhanced maximum throughput. Despite that these WDM networks may avoid the limitations of tunable transmitters and receivers, there is no wavelength sharing between stations. That is, each link in a connected topology corresponds to a unique wavelength. This may cause bandwidth underutilization and the number of fixed wavelength transmitters and receivers in each station are based on the degree of the connected topology. To avoid these shortcomings and reduce hardware cost and yet to fully utilize the broadcast capability of a passive star, we may employ TDM protocol on each channel (wavelength) to allow several stations to share a wavelength and thereby reduce the number of fixed wavelength transmitters and receivers in each station. In this paper, we study design principles for multi-hop WTDM optical passive star networks with a constraint of *one* fixed wavelength transmitter and *one* fixed wavelength receiver in each station.

Consider a WTDM optical passive star based network in which only one fixed wavelength transmitter and receiver are used in each station. If we allow each station to broadcast to the rest of the stations (i.e., the transmitter's wavelength of each station should be the same as the receivers' wavelength of the rest of the stations), then only one wavelength can be exploited and all transmitters share that wavelength. That is, the network operates a pure TDM protocol. Although this results a fully connected topology (i.e., the distance between any two stations is always a single hop away), the low bandwidth exploited (only one wavelength is used) may severely restrict its performance. In order to exploit higher bandwidth, we have to use more wavelength channels. That is, fewer transmitters share one wavelength and several wavelengths can be used for transmission at the same time. The extreme case is to allow each transmitter to use a different wavelength. Since each station has only one receiver, each station can only listen to (receive from) one channel. This effectively results in a pure multi-hop WDM network and its connection pattern is a uni-directional ring. Thus, on average a packet needs to go through $\frac{N}{2}$ intermediate stations to reach it destination. The traffic overhead caused by the long distance (in terms of the number of hops) between stations may be greater than the advantage gained from higher bandwidth (N wavelengths) exploited.

The (p, k)-ShuffleNet proposed in [9] improves the work of [1] by using a WTDM protocol and requiring only one fixed wavelength transmitter and one fixed wavelength receiver in each station (where p and k are the degree and the number of stages of ShuffleNet topology respectively). The total number of stations $N = k \cdot p^k$. Also the total number of wavelengths used is N/p. It is also pointed out [9] that for a given set of stations, by using a ShuffleNet with more stages k (higher diameter), we can exploit more wavelengths and obtain higher system throughput. However, a larger network diameter implies a longer packet delay in a wide-area environment where the propagation delay between stations is much more significant than the packet transmission time and queuing delay incurred in each hop. Therefore, to minimize the delay, a 2-stage ShuffleNet is preferred.

In [18] another WTDM network, called *Bus- Mesh*, was proposed. It requires the same hardware requirement for each station. The Bus-Mesh network guarantees the path length between any two stations is bounded by 2 and it is also demonstrated that Bus-Mesh outperforms the 2-stage ShuffleNet under certain conditions. However, the number of wavelengths that can be exploited by Bus-Mesh is bounded by $\sqrt{N}$.

In a local environment propagation delay may be only a few times as long as the transmission time. Furthermore, packet delay is also related to the amount of bandwidth each station can obtain. Therefore, to minimize diameter and sacrifice throughput may not be the best strategy. Later we will see, in certain cases, connected topologies with a higher diameter that may offer shorter delay as well as higher throughput.

Specifically, we intend to answer the following questions in this paper.

- Is there a general methodology to design such multi-hop WTDM optical passive star networks?
- What are the fundamental relationships between several design parameters such as the number of wavelengths exploited, the connected topology, and the number of stations in the network?
- What are the best design strategies for multi-hop WTDM networks in different environments?
- What are the performance limitations of these types of networks?

In order to study and understand multi-hop WTDM networks, a model which can reveal the inherent properties of such networks is needed. Based on the broadcast nature of a passive star, we first defined a graph model which is called a *receiving graph model*. Based on the model, we are able to answer many performance related questions including some theoretical performance limitations of such networks. We also discuss several design principles for such networks. We propose a general design methodology for such networks which is based on the relationship between receiving graphs and a given connected topology. Several design alternatives are also presented. It will be shown that both Bus-Mesh and ShuffleNet are two special cases of the proposed design methodology.

This paper is organized as follows. In Section 2, we briefly describe basic multihop WTDM networks by using the Bus-Mesh as an illustration. In Section 3 we propose a graph model, called a *receiving graph model*, which can represent a WTDM network with one fixed wavelength transmitter and receiver in each station. Several design parameters are also identified and the trade-offs between them are discussed. In Section 4, we show how to construct a receiving graph which represents a particular WTDM network for a given connected topology. Transmission scheduling and routing algorithms for such WTDM networks are also described. In Section 5, we define two performance metrics and derive their approximated analytical models based on general graphs. The design strategies for optimizing the metrics and the theoretical bounds of the metrics are also presented. Moreover, we choose m-ary n-cube and Shuffle-exchange as example interconnection topologies to demonstrate constructing WTDM networks. Their effectiveness is discussed. In Section 6, several practical design issues, such as synchronization, propagation delay and dynamic bandwidth allocation, are addressed. Possible solutions are also provided. In the final section, we draw some conclusions.

2 Basic Multi-hop WTDM Network

In this section, we shall illustrate the basic WTDM network by describing the Bus-Mesh network proposed in [18]. This type of networks is based on a physical optical passive star as shown in Figure 1. Each station transmits (receives) via a fixed wavelength transmitter (receiver) which is tapped onto a passive star. We assume that all wavelengths have the same bandwidth which is bounded by the maximum signal modulation/demodulation speed of a station. A basic data unit, called a *packet*, is of fixed size. The time domain is divided into time slots of equal duration with a slot long enough to contain a packet. The time slots are logically arranged into repeating cycles with each station transmitting once within a cycle at a predetermined wavelength. We call this a *transmission cycle*. During each time slot, on each wavelength, only one station is entitled to transmit. Each station always sets its receiver to a predetermined wavelength. Given two stations, say stations a and b, if the transmitter's wavelength of station a is the same as the receiver's wavelength of station b, we say there is a connection from station a to station b. The pattern of interconnection forms a connected topology of the network. A straightforward representation of the connected topology is to use a node to represent a station and a directed link between two nodes to indicate a connection. This is used in [9][1][24]. However, this method can not effectively capture the unique characteristics of the network, such as its broadcast characteristics, channel sharing capabilities, and the number of fixed wavelength transmitters and receivers. Another representation is proposed in Bus- Mesh [18]. For example, Figure 2 is one way to view the connected topology of a 12-station Bus-Mesh network. We shall illustrate another view in the next section.

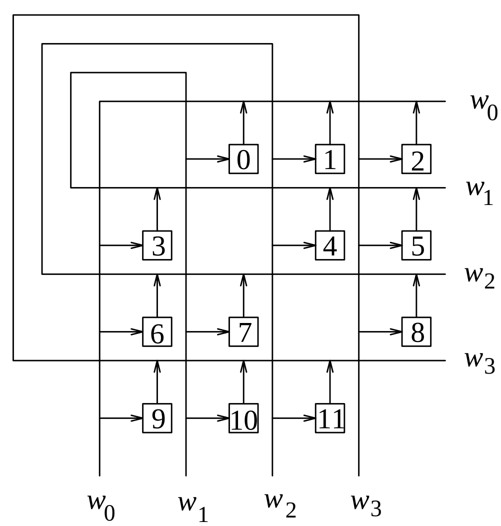


Figure 2: The connected topology of a 12-station 4-wavelength Bus-Mesh network

Wavelengths are logically represented by a set of "buses". Each station is represented by a node. Each stations (node) transmits and listens on two different wavelengths (buses) (Notice that no node is located on the diagonal). For a W-wavelength network, at least $W-1$ stations transmit on (or receive from) each wavelength. Therefore, $N \geq W \times (W-1)$. In the above case $W=4$ and $N = W \times (W-1) = 12$. Table 2 shows a possible *transmission cycle*. The column index t_i means the relative time slot number and the row index w_j means the wavelength id. An entry $k \rightarrow l, m, n$ in column t_i and row w_j means that station k has the right to transmit at wavelength w_j in time slot t_i

and this transmission can be simultaneously received by stations l, m, n directly (i.e., stations l, m, n have their receivers all tuned to wavelength w_j). The same transmission cycle is repeated as time goes on. The transmission cycle length is equal to the number of stations that share the same wavelength. Clearly, since each station has only one chance to transmit during a cycle, the longer the cycle length is, the smaller portion of bandwidth a station may use. For instance, in t_0, stations 0, 3, 6 and 9 are entitled to transmit, since they tap onto distinct buses (transmit on distinct wavelengths). Each of them grants one third of the bandwidth of a single wavelength.

	t_0	t_1	t_2
w_0	$0 \rightarrow 3, 6, 9$	$1 \rightarrow 3, 6, 9$	$2 \rightarrow 3, 6, 9$
w_1	$3 \rightarrow 0, 7, 10$	$4 \rightarrow 0, 7, 10$	$5 \rightarrow 0, 7, 10$
w_2	$6 \rightarrow 1, 4, 11$	$7 \rightarrow 1, 4, 11$	$8 \rightarrow 1, 4, 11$
w_3	$9 \rightarrow 2, 5, 8$	$10 \rightarrow 2, 5, 8$	$11 \rightarrow 2, 5, 8$

Table 2: Transmission cycle for a 12-station 4-wavelength Bus-Mesh network

Each station is also equipped with an output queue to temporarily buffer outgoing packets. During each transmission turn only one packet from the output queue is sent out. Each packet includes a *destination station* address in its header. Upon receiving a packet, a station decides whether the packet is addressed to itself or not by examining the destination station address included in the packet header. If yes, the packet is accepted; otherwise, the station further decides whether he is responsible for relaying the packet or not. If not, the station simply discard the packet; otherwise, he will buffer the packet in his output queue for later transmission. For example, suppose station 2 has a packet for station 8. After station 2 broadcasts in t_2 at w_0, stations 3, 6, 9 receive the packet. Nodes 3 and 6 realize they are not responsible for relaying the packet, so the packet is discarded. Station 9 buffer the packet in his output queue temporarily. In t_0, station 9 transmits the packet at wavelength w_3. Then stations 2, 5 and 8 will all receive the packet, but only station 8 will realize the packet is addressed to him. The others discard the packet. In Bus-Mesh any station can reach any other station within 2 transmissions, so the diameter is 2. Essentially, it is the minimum diameter in an environment using one fixed transmitter and one fixed wavelength receiver in each station (we will explain why later).

From the above example, we can identify several design parameters which potentially dominate the performance of this network, such as the number of stations, the number of wavelengths used, the transmission cycle length, the bandwidth reserved for each station and the diameter. However, Bus-Mesh only represents a specific type of connected topologies and it is still very hard to compare the performance of different WTDM networks.

3 Receiving Graph Model

In order to understand WTDM networks in which each station has only one fixed wavelength transmitter and one fixed wavelength receiver, we need a convenient way to represent them. Therefore, we propose a graph model called a *receiving graph model* in this section.

In the *receiving graph model*, each station in the network corresponds to a *node*. Thus, there are N nodes, where N is the number of stations in the network. (In the following context, "node" and "station" are interchangeable, unless specified explicitly.) Since each node has only one fixed wavelength receiver, according to the receiver's assigned wavelength, all nodes can be partitioned into W sets, where W is the number of wavelengths used. We shall call each set a *receiving node*. Thus, a receiving node represents a set of stations (nodes) which use the same receiver wavelength and is associated with this unique wavelength. Therefore, we shall label a receiving node with the receiver wavelength id. If a node transmits on a particular wavelength, there is a directed edge originating from the node (inside a receiving node) to the receiving node associated with the transmitter's wavelength. Since each node can only transmit at one fixed wavelength, each node has only one outgoing edge pointing to a receiving node. The outgoing degree of a receiving node equals the number of nodes inside. The incoming degree of a receiving node equals the number of nodes who share the same transmission wavelength associated with the receiving node. We call this set of nodes the *transmitting group* of the wavelength. Since each node only transmits once in a transmission cycle, the cycle length is equal to the number of nodes who share this transmission wavelength (i.e., the size of the associated transmitting group). Note that for simplicity we assume the in-degree of each receiving node is the same, although it is not necessary to be the case in a WTDM network. The transmission cycle can be easily constructed by arranging all the nodes in the same transmitting group to transmit in any order (However, to be more effective, we will arrange them in a specific pattern which will be elaborated on later).

Suppose we have N nodes and W wavelengths. Let N nodes be denoted as n_0, n_1, ..., n_{N-1} and W wavelengths w_0, w_1, ..., w_{W-1}. The receiving node associated with wavelength w_i is denoted as rn_i. For example, in Figure 2 all stations receiving from the same bus form a receiving node. Then the Bus-Mesh can be redrawn like a 4-node receiving graph shown in Figure 3. The nodes are shown by a box shape (only node indices are shown) and the receiving nodes are shown by a round shape. The receiving graph has an outgoing degree of 3 as well as an incoming degree of 3. That means three nodes receive at the same wavelength and three nodes share the same wavelength (in a TDM fashion) for transmission. In Table 2 the set of three nodes appearing on the right hand side of the arrows in the same row corresponds to the receiving node associated with that row (e.g., the set of n_3, n_6 and n_9 in row w_0 corresponds to rn_0), and the set of three nodes appearing on the left hand side of the arrows in the same row corresponds to the transmitting group associated with that row (e.g., the set of n_0, n_1 and n_2 in row w_0 corresponds to the transmitting group of w_0).

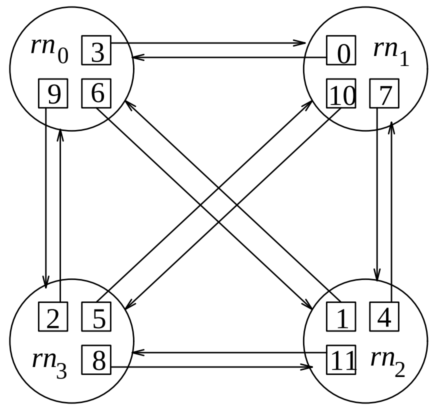
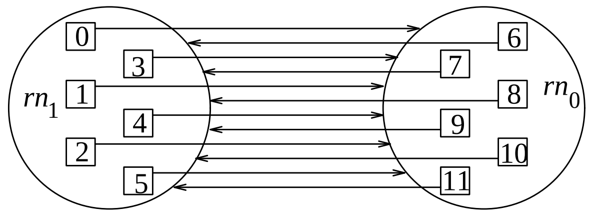


Figure 3: The receiving graph representation of the Bus-Mesh shown in Figure 2.

When a node, say n_s, has a packet to transmit to another node ,say n_d, it has to transmit on a fixed wavelength (recall each station has only one fixed wavelength transmitter). This corresponds to starting from n_s to follow its out-going edge to a receiving node. Since there are several nodes inside this receiving node, one of them will be chosen as an intermediate node to relay this packet to another receiving node. That is, this node receives the packet and then transmits on its transmission wavelength. This process is repeated until the receiving node containing the destination node is reached. To minimize the number of hops traversed, we need to follow the shortest path from the first receiving node (i.e., the receiving node first reached from n_s or the receiving node corresponding to the transmission wavelength of n_s) to the final receiving node (i.e., the receiving node who contains n_d).

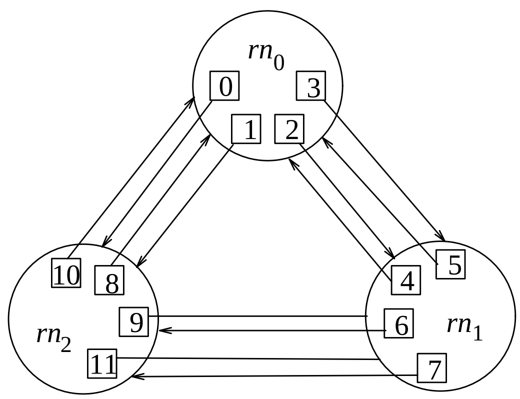
If we ignore the nodes inside each receiving node and consider that there is a directed edge from a receiving node (rn_a) to another receiving node (rn_b) as long as there is a node inside rn_a which has an out-going edge to rn_b, this results in a *simplified receiving graph*. Therefore, the maximum shortest path length (in terms of the number of hops traversed) between any two nodes equals one plus the diameter of the simplified receiving graph. Surprisingly, the simplified receiving graph of Bus-Mesh is a completely connected graph. As we know the diameter of a completely connected graph is one, this explain why the minimum number of hops traversed for a packet transmitted from one node to another in Bus-Mesh is bounded by two.

Based on the concept of a receiving graph, we can easily come up with several alternative designs which may use a different number of wavelengths for a WTDM network with 12 nodes. For instance, we can construct a receiving graph by using only two wavelengths (as shown in Figure 4). In this case each receiving node contains 6 nodes. Using 3 wavelengths, then each receiving node contains 4 nodes (as shown in Figure 5). Using 6 wavelengths, then each receiving nodes contains 2 nodes (as shown in Figure 6). The corresponding transmission cycles are also shown in the figures. The length of each transmission cycle is equal to the outgoing (incoming) degree of the corresponding receiving graph. Note: it is clear that each receiving graph corresponds to a WTDM network.



	t_0	t_1	t_2	t_3	t_4	t_5
w_0	$0 \rightarrow 6,..,11$	$1 \rightarrow 6,..,11$	$2 \rightarrow 6,..,11$	$3 \rightarrow 6,..,11$	$4 \rightarrow 6,..,11$	$5 \rightarrow 6,..,11$
w_1	$6 \rightarrow 0,..,5$	$7 \rightarrow 0,..,5$	$8 \rightarrow 0,..,5$	$9 \rightarrow 0,..,5$	$10 \rightarrow 0,..,5$	$11 \rightarrow 0,..,5$

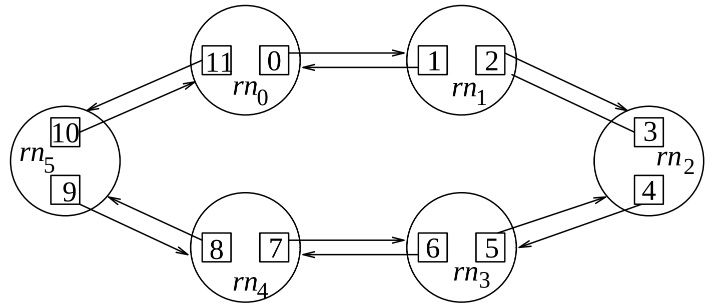
Figure 4: The receiving graph using two wavelengths and its transmission cycle.



	t_0	t_1	t_2	t_3
w_0	$4 \rightarrow 0,1,2,3$	$5 \rightarrow 0,1,2,3$	$8 \rightarrow 0,1,2,3$	$10 \rightarrow 0,1,2,3$
w_1	$2 \rightarrow 4,5,6,7$	$3 \rightarrow 4,5,6,7$	$9 \rightarrow 4,5,6,7$	$11 \rightarrow 4,5,6,7$
w_2	$0 \rightarrow 8,9,10,11$	$1 \rightarrow 8,9,10,11$	$6 \rightarrow 8,9,10,11$	$7 \rightarrow 8,9,10,11$

Figure 5: The receiving graph using three wavelengths and its transmission cycle

From the above examples, for WTDM networks with the same number of nodes it can be seen that there is a trade-off between the number of wavelengths exploited and the diameter of the simplified receiving graph. As we use fewer wavelengths, the number of receiving nodes becomes fewer and the number of nodes inside a receiving node becomes larger. That is, the simplified receiving graph has a higher degree since



	t_0	t_1
w_0	$1 \rightarrow 0, 11$	$10 \rightarrow 0, 11$
w_1	$0 \rightarrow 1, 2$	$3 \rightarrow 1, 2$
w_2	$2 \rightarrow 3, 4$	$5 \rightarrow 3, 4$
w_3	$4 \rightarrow 5, 6$	$7 \rightarrow 5, 6$
w_4	$6 \rightarrow 7, 8$	$9 \rightarrow 7, 8$
w_5	$8 \rightarrow 9, 10$	$11 \rightarrow 9, 10$

Figure 6: The receiving graph using six wavelengths and it transmission cycle.

each node inside the receiving node contributes an edge going to another receiving node. Thus, the simplified receiving graph becomes much denser and it should have a shorter diameter. However, the length of the transmission cycle becomes longer (i.e., the size of the transmitting group or the number of nodes inside a receiving node). That means the portion of the bandwidth that each node uses is reduced. On the contrary, if more wavelengths are used, there are more receiving nodes (i.e., the number of wavelengths used) and the number of nodes inside a receiving node is smaller (i.e., the simplified receiving graph has a lower degree). Thus, the receiving graph becomes more sparse and should have a longer diameter. However, the cycle length becomes shorter. That means each node uses a higher portion of the bandwidth. It is very hard to determine which one has a better performance. What are the best designs in terms of the number of wavelengths used, the cycle length and the topology for the simplified receiving graph? This will be discussed in more details in Section 5.

We should point out that the simplified receiving graph need not be a regular directed graph as shown in the examples. It can be tailored into a graph for a specific traffic pattern such that the traffic overhead is minimized. However, this issue is out of the scope of this paper. In this paper, we focus on a general network communication pattern. That is, we assume that each node may generate packets for any other node at any given time. For each wavelength, there is a set of nodes transmitting on it and a set of nodes receiving from it. We would like each wavelength to be fully and equally utilized. Therefore, we shall assume the number of nodes transmitting and receiving on a wavelength is the same for any wavelength. Let β denote this number. Then

$$\beta = \frac{N}{W}.$$

Essentially, this tells us that each receiving node has both β incoming and β outgoing edges. Therefore, the simplified receiving graph is a regular directed graph of degree β and size W.

4 Designing WTDM Networks

In the previous section, we have shown a way to construct a *receiving graph* based on a given multi-hop WTDM network. In this section we would like to show a systematic way to design a multi-hop WTDM based on our understanding of the receiving graph. In several previous examples, we have mentioned the simplified receiving graph. A simplified receiving graph is more like a regular graph. Therefore, our approach is to show how to construct a receiving graph (which corresponds a multi-hop WTDM network) based on a given regular graph. We shall refer to this process as a *virtual graph embedding.*

4.1 Virtual Graph Embedding

Assume the number of stations N in the network is a multiple of the number of exploited wavelengths W (i.e., $N = C \times \alpha \times W$, where both C and α are positive integers). (More general cases that C is a positive real number will be discussed later.) We can choose a regular directed graph with W nodes and degree α (i.e., both in-degree and out-degree of each node is α). To distinguish from the receiving graph, we shall call the given graph a *virtual graph* and its nodes *virtual nodes.* For a given virtual graph, the process of constructing a receiving graph is referred to as *virtual graph embedding.* For each directed edge in the virtual graph, we attach a box-shaped node at the starting point. Similar to the receiving graph model, each virtual node corresponds to a wavelength used for receiving. All the box-shaped nodes in a virtual node correspond to all the stations tuned to this receiving wavelength. Each box-shaped node also has an out-going edge pointed to a virtual node. That is, the station corresponding to the box-shaped node transmits on the wavelength corresponding to the virtual node pointed to by its out-going edge. Therefore, a box-shaped node can be distinguished by ($tran\_id$, rec_id), where $tran\_id$ and rec_id denote its transmission and receiving wavelengths respectively. Then the resulting graph looks more like a receiving graph. For example, Figure 7(a) shows a virtual graph of size 3 and degree 2. After the above process, it is transformed into the graph shown in Figure 7(b).

The number of box-shaped nodes in each virtual node equals the degree of the virtual graph (i.e., α). Since there are W virtual nodes, the total number of box-shaped nodes is $W \times \alpha$. This is like a receiving graph of $W \times \alpha$ nodes. Since $N = C \times W \times \alpha$, we need to duplicate each box-shaped node in each virtual node C times. Accordingly the outgoing

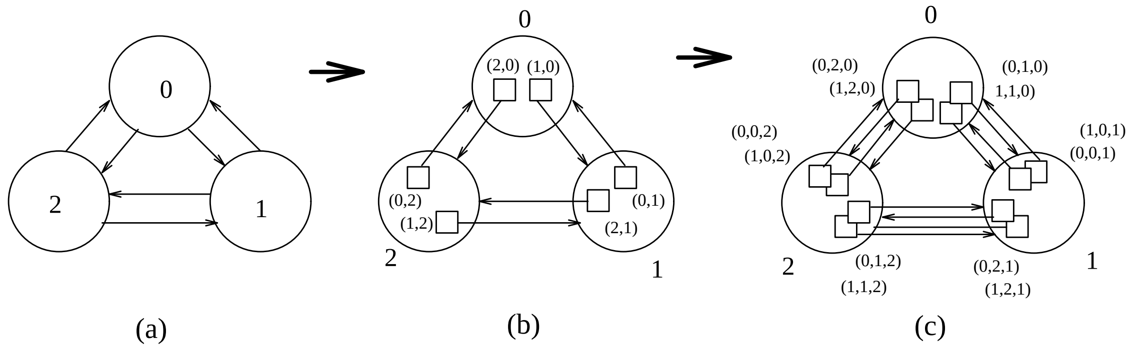


Figure 7: The process of virtual graph embedding.

edges from each virtual node are also duplicated C times. Then the resulting graph can be considered as a receiving graph of N nodes. However, for a given pair of wavelengths ($tran\_id$, rec_id) there are C box-shaped nodes corresponding to it. We shall denote a box-shaped node by a triplet ($stack\_id$, $tran\_id$, rec_id), where $stack\_id$ is an integer between 0 and $C-1$. All nodes with the same $stack\_id$ are considered in the same *group* (*stack*). Conceptually, we can imagine that a receiving graph is constructed by stacking C copies of a virtual graph together and adding a box-shaped node at each edge starting point.

For example, after the duplication process, the graph shown in Figure 7(b) is transformed to the receiving graph shown in Figure 7(c). It corresponds to a 12-node 3-wavelength WTDM network.

4.2 Transmission Cycle

So far we have provided a way to determine the transmission and receiving wavelengths of each station. In this section we discuss the issues of determining the transmission cycle. As mentioned before, a transmission cycle can be described by means of a table in which the row index represents the transmission wavelength id number and the column index represents the relative time slot in the cycle. Each entry tells us which node is entitled to transmit to which other nodes (which receiving node). We have W wavelengths, so the table has W rows. At any given time slot at most W packets, one on each different wavelength, can be transmitted. Each node in each transmitting group transmits only once in a transmission cycle, so the table has $\frac{N}{W}$ columns. Let $N = W \times C \times \alpha$, the *transmission cycle length* (which equals the length of the table), denoted as Δ, will be

$$\Delta = \frac{N}{W} = C \times \alpha. \qquad (1)$$

Note that since each node receives a signal from a predetermined wavelength, the set of nodes receiving on a given wavelength is static and known to all the nodes transmitting on the wavelength. Furthermore, each node only needs to know the transmission schedule of its own transmitting group and does not have to know those of other transmitting groups. Therefore, the only necessary information which needs to be stored in each node is a row in the transmission cycle.

As mentioned before, since no contention will occur among different wavelength transmissions, the schedule of the nodes in the same transmitting group (row) can be in any order. However, there may exist several paths between two nodes. The waiting time from a packet arriving at a node till the node is entitled to transmit the packet may be different form node to node. Therefore, it is desirable to find the path between two nodes in which a packet can be delivered with the shortest delay. We also would like to be as fair to each node pair as possible. To do so, we first suggest to how to arrange the transmission cycle on a stack-by-stack basis. Then we show how to find the shortest delay path.

Basically, all nodes with the same stack id number are scheduled in a period of contiguous time slots. We refer to it as *transmission subcycle*. Since each virtual node in a stack of a virtual graph is of degree α, the *transmission subcycle length*, denoted as Δ_s, will be

$$\Delta_s = \alpha = \frac{N}{C \cdot W}. \tag{2}$$

Thus the whole transmission cycle is composed of C transmission subcycles each of which is for a different stack of the virtual graph. Within a transmission subcycle, the order of transmissions for all nodes in the same row is arbitrary, but the same pattern is repeated in every transmission subcycle. That is, provided that transmission subcycles are numbered from 0 to $C-1$, the order of transmissions for the nodes in a row w_t can be represented by $(0, t, r_0)$ ... $(0, t, r_{\alpha-1})$ $(1, t, r_0)$ ... $(1, t, r_{\alpha-1})$ ... $(C-1, t, r_0)$ ... $(C-1, t, r_{\alpha-1})$.

For example, Table 3 shows the transmission cycle of the receiving graph shown in Figure 7(c), where "$*$" means any legal value. All nodes pointing to the same receiving nodes are located in the same row. Columns t_0 and t_1 are dedicated for subcycle 0 (stack 0), and columns t_2 and t_3 for subcycle 1 (stack 1).

	t_0	t_1	t_2	t_3
w_0	$(0,0,2) \to (*,*,0)$	$(0,0,1) \to (*,*,0)$	$(1,0,2) \to (*,*,0)$	$(1,0,1) \to (*,*,0)$
w_1	$(0,1,0) \to (*,*,1)$	$(0,1,2) \to (*,*,1)$	$(1,1,0) \to (*,*,1)$	$(1,1,2) \to (*,*,1)$
w_2	$(0,2,0) \to (*,*,2)$	$(0,2,1) \to (*,*,2)$	$(1,2,0) \to (*,*,2)$	$(1,2,1) \to (*,*,2)$

Table 3: The transmission cycle of the receiving graph shown in Figure 7(c). It consists of two transmission subcycles.

4.3 Routing

In this subsection, two types of routing algorithms are proposed. One aims at minimizing packet delay and is suitable for special control packets. The other aims at balancing traffic load and is suitable for general data packets.

In a WTDM network, a k-hop path from a node $n_{(s_s,t_s,r_s)}$ to another node $n_{(s_d,t_d,r_d)}$ can be generally represented as

$$\begin{aligned} & n_{(s_s,t_s,r_s)} \rightarrow n_{(s_1,j_1,t_s)} \rightarrow n_{(s_2,j_2,j_1)} \rightarrow n_{(s_3,j_3,j_2)} \rightarrow \dots \\ & \rightarrow n_{(s_{k-2},j_{k-2},j_{k-3})} \rightarrow n_{(s_{k-1},r_d,j_{k-2})} \rightarrow n_{(s_d,t_d,r_d)}. \end{aligned} \quad (3)$$

In the path, each intermediate node receives packets from a wavelength, and then retransmits it on another wavelength, so that a node's receiving wavelength is equal to the transmission wavelength of the node's predecessor (except for the source node). In the corresponding receiving graph, this can be viewed as a path of receiving nodes. Those receiving nodes correspond to the wavelengths which the nodes transmit on. That is,

$$rn_{t_s} \rightarrow rn_{j_1} \rightarrow rn_{j_2} \dots \rightarrow rn_{j_{k-3}} \rightarrow rn_{j_{k-2}} \rightarrow rn_{r_d}. \quad (4)$$

We shall call such a path a *(k-1)-hop virtual path*. Clearly, any path corresponds to a virtual path and the length of the path equals the length of its corresponding virtual path plus one. A virtual path may correspond to several paths. Recall that, for each edge in a virtual graph, there are C edges in the receiving graph associated with it and each belongs to a distinct stack. Thus, a move from a receiving node to another receiving node can be taken through any one of them. For instance, all the alternative paths from $n_{(s_s,t_s,r_s)}$ to $n_{(s_d,t_d,r_d)}$ corresponding to the virtual path shown in (4) are

$$\begin{aligned} & n_{(s_s,t_s,r_s)} \rightarrow n_{(*,j_1,t_s)} \rightarrow n_{(*,j_2,j_1)} \rightarrow n_{(*,j_3,j_2)} \rightarrow \dots \\ & \rightarrow n_{(*,j_{k-2},j_{k-3})} \rightarrow n_{(*,r_d,j_{k-2})} \rightarrow n_{(s_d,t_d,r_d)}. \\ & \text{where } * \text{ mean any integer from 0 to } C-1. \end{aligned} \quad (5)$$

In total, there are C^{k-1} alternative k-hop paths. Furthermore, if there are v alternative $(k-1)$-hop virtual paths between rn_{t_s} and rn_{r_d}, the total number of alternative k-hop paths between n_{s_s,t_s,r_s} and n_{s_d,t_d,r_d} will be equal to

$$v \cdot C^{k-1} \quad (6)$$

Which path should be actually followed? Note that the virtual paths form rn_{t_s} to rn_{r_d} can easily be found based on the routing algorithm of a virtual graph. However, each of the virtual paths corresponds to a set of paths. Despite that all the paths in the same set can properly deliver packets, because of the timing differences of node transmissions, the packet delay (the time period from a packet being generated by a source node to it arriving at a destination node in terms of time slots) may be significantly different for different paths. In practice, certain types of packets (e.g., control packets) require fast delivery and are usually very small. For this type of traffic, we prefer to choose the path with the shortest delay. However, for general data packets, the system throughput seems to be more important than the packet delay. Good load balancing can avoid unnecessary traffic congestion and improve the system throughput. Therefore, for this type of traffic, we would like to choose the path with the lightest traffic load. In the following we propose two different routing algorithms: the *fast packet delivery routing algorithm* for control type packets and the *load balancing routing algorithm* for general type packets.

4.3.1 Fast Packet Delivery Routing Algorithm

The basic idea of the *fast packet delivery routing algorithm* is to first find the shortest length virtual path and then choose one of the corresponding paths with the shortest delay. Let VG be a virtual graph and $R_{VG}(\cdot)$ be the routing algorithm of VG. We assume the routing algorithm $R_{VG}(\cdot)$ provides the shortest length virtual path and is based on destination address routing. That is, given a current receiving node address c and a destination receiving node address d, $R_{VG}(c,d)$ will return the next receiving node address in the shortest length path. For instance, the routing for hypercube or shuffle-exchange belongs to this type. We further assume the signal propagation delay is the same between any pair of nodes. Consider $n_{(s_s,t_s,r_s)}$ has a packet for $n_{(s_d,t_d,r_d)}$. The destination node address (i.e., $n_{(s_d,t_d,r_d)}$) is included in the packet header. Initially, $n_{(s_s,t_s,r_s)}$ transmits the packet on w_{t_s}. All nodes in rn_{t_s} (i.e., $n_{(*,*,t_s)}$) receive the packet. If r_d (provided in the packet header) is equal to t_s, the destination receiving node is reached. All nodes in rn_{t_s} further compare stack id and transmission wavelength id with those provided in the packet header. Only $n_{(s_d,t_d,r_d)}$ (the destination node) accepts the packet and the others discard the packet. If $r_d \neq t_s$, the same routing algorithm $R_{VG}(t_s,r_d)$ is executed in each of $n_{(*,*,t_s)}$ to decide the next receiving node address. Suppose $R_{VG}(t_s,r_d)=i_1$. Only $n_{(*,i_1,t_s)}$ have their associated edges direct to rn_{i_1} and any one of them can properly relay the packet, so they will keep the packet for further checking. The others just discard the packet. Equal propagation delay implies all $n_{(*,i_1,t_s)}$ receive the packet at the same time. Upon receiving the packet, each of them simultaneously computes how long it is going to wait before the next transmission. If the time interval is less than or equal to Δ_s, the node becomes the winner and is responsible for relaying the packet. Note that since a transmission cycle is arranged on a stack-by-stack basis and any two nodes ($\in n_{(*,i_1,t_s)}$) in adjacent stacks are separated by the subcycle length Δ_s, there is one and only one winner. The others discard the packet. The same process is repeated until $n_{(s_d,t_d,r_d)}$ is reached.

Clearly, in the above routing algorithm, we always look for the node who is soonest scheduled for transmission to relay packets. If a node is always available for relaying the just received packet, the shortest routing delay based on the shortest length virtual path is guaranteed. This may be the case for light traffic load. However, for moderate or heavy traffic load, the probability of having one or more packets queued in a node may be very high. In that case, the algorithm may not be able to guarantee the shortest delay path. In general, only certain control packets demand this kind of fast delivery and their number is relatively small. Therefore, we can give them a higher priority (i.e., be transmitted prior to others) over other traffic. They can be delivered without being blocking or queued in the intermediate nodes.

4.3.2 Load Balancing Routing Algorithm

The principle of the *load balancing routing algorithm* is that we always choose the node with least queuing to relay packets. To achieve this, each node must somehow have knowledge of the situation of other nodes in the same receiving node, such as their queue lengths. Upon receiving a packet, all the nodes within this community should eventually come up with a consistent choice and this one is elected to relay the packets. Owing to the fact that packets are broadcast to each node in the same receiving node and each node is running the same routing algorithm, and only one packet can be transmitted during a transmission cycle, each node can figure out how many packets are left in the others' queues. To implement this, each node is equipped with $\frac{N}{W}$ counters with each for a node in its own receiving node (including itself). Whenever a node receives a packet, the corresponding counters in each node will be incremented by one. Whenever a transmission cycle passes, each counter is decremented by one, unless it has reached zero. Based on this counter information, a routing algorithm can be devised to decide which node should relay packets. This decision should avoid a cycle and at the same time balance the traffic load. There are several existing adaptive routing algorithms for regular topologies and we believe that they can be easily modified and adapted into this environment. We will not discuss them here.

Note that the number of packets actually queued in a specific node may be slightly higher than the information collected in each node because the packets internally generated by the node are not taken into account. Therefore, there may exist some potential of congestion of which others are not aware. However, from the fairness point of view, each node should share the same responsibility of relaying packets. Furthermore, under a uniform communication assumption, each node has the same probability of generating packets. It should also be the case that each node is responsible for controlling the amount of traffic entering the network to avoid a overloading situation. Therefore, although the collected information in each node may not really reflect the actual information, we believe that it is sufficient to provide a balanced loading situation.

4.4 Partial Stack Design

The virtual embedding provides a great deal of flexibility in designing a WTDM network. Several dimensions of freedom, such as the topology of a virtual graph, the virtual graph size and degree, and the number of stacks, allow us to design a WTDM network in a desired manner. Nevertheless, in practice, given N nodes and a virtual graph of size W and degree α, we may not always be able to find an integral C such that $N = C \times W \times \alpha$. To remedy this problem, we allow C to be a positive real number.

	t_0	t_1	t_2	t_3
w_0	$(0,0,2) \rightarrow (*,*,0)$	$(0,0,1) \rightarrow (*,*,0)$	$(1,0,2) \rightarrow (*,*,0)$	
w_1	$(0,1,0) \rightarrow (*,*,1)$	$(0,1,2) \rightarrow (*,*,1)$	$(1,1,0) \rightarrow (*,*,1)$	
w_2	$(0,2,0) \rightarrow (*,*,2)$	$(0,2,1) \rightarrow (*,*,2)$		

Table 4: The transmission cycle for a 8-node WTDM network using a virtual graph to do the virtual embedding.

Considering C is a non-integer, a receiving graph consists of $\lceil C \rceil$ stacks. Stacks 0 to $\lceil C \rceil - 2$ are "*complete*" stacks of a virtual graph (i.e., without missing any edge (node) in each stack), and stack $\lceil C \rceil - 1$ is a "*partial*" stack of the virtual graph (i.e., some edges (nodes) are missing from the stack). The fractional part of C represents the ratio of the existing edges over the original edges. The smaller the ratio is, the larger the number of edges (nodes) missing from the partial stack.

Likewise, the transmission cycle has $\lceil C \rceil$ transmission subcycles. Subcycles 0 to $\lceil C \rceil - 2$ are fully scheduled for transmission, and subcycle $\lceil C \rceil - 1$ is only partially filled up. Each missing edge in stack $\lceil C \rceil - 1$ corresponds to an empty slot in subcycle $\lceil C \rceil - 1$. In order to be fair to each transmitting group, missing edges should be evenly distributed among all the rows (transmitting groups). By doing so, the number of empty slots left in each row of subcycle $\lceil C \rceil - 1$ is about the same. These empty slots can be assigned to any node in the same row on a demand or uniform-distribution basis. For example, consider an 8-node WTDM network using the virtual graph shown in Figure 7(a). To do the virtual embedding, we have $W = 3$, $\alpha = 2$ and $C = 1.33$. The corresponding transmission cycle is shown in Table 4. It consists of two subcycles. Subcycle 0 is corresponding to a complete stack, and subcycle 1 a partial stack. There is one empty slot left for rows w_0 and w_1 and two for row w_2. The empty slots can be assigned to any node in the same row for transmission.

For the cases of $C < 1$, we have only one partial stack of the virtual graph. Because some edges are missing, the routing algorithm of the virtual graph can not be directly applied. In order to properly deliver packets, certain supplementary procedures should be employed in those nodes such that the missing edges can be substituted by alternative routes. Of course, this will introduce extra complexity in routing and the path length may increase. Therefore, $C < 1$ is not a good design choice.

On the other hand, for the cases of $C \geq 1$, we have at least one complete stack of the virtual graph. That is, each edge in the virtual graph corresponds to at least one edge in the receiving graph. Recall that in (5), a path between two nodes can go through any stack combination. Therefore, for each virtual path, there is at least one path corresponding to the virtual path. That means the routing algorithm of the virtual graph will not be affected. For the sake of routing simplicity, the constraint of $C \geq 1$ is very important. Thus, in the following discussion we will enforce this constraint.

5 Design Strategies

In the previous section, we discussed how to design a WTDM network based on a set of given parameters such as the number of nodes, the number of wavelengths, and a given virtual graph. In this section, we will further evaluate the performance of WTDM networks in terms of these parameters. We assume that there are N stations and W wavelengths available. We would like to know what is the best virtual graph that can be used to design WTDM networks. Since we are looking for the performance of WTDM networks constructed from any virtual graph (not any particular type of graphs), a general and formulation of performance is needed. Thus, two performance metrics are considered: the *average network throughput for high-load* and the *average packet delay for low-load*. Under a light traffic load, the packet delay is more crucial. Under a heavy traffic load, the network throughput becomes more important. We believe that these two metrics are capable of characterizing the performance of WTDM networks in a general sense. Furthermore, they provide some guidelines in choosing the best virtual graph.

First let us define some terms related to general virtual graphs. The *length (number of hops)* of a path between two nodes in a graph is the number of edges along the path. The *distance* (denoted as H) between two nodes is the minimum length of the paths between them. The *diameter* (denoted as D) is defined as the the longest distance between any pair of nodes. Since the number of nodes in a virtual graph corresponds to the number of wavelengths used in the virtual embedding process, we shall denote the number of nodes and the degree of a virtual graph as W and α respectively. For ease of discussion, we assume the virtual graph is a simple graph (A simple graph refers to a graph with no duplicated edges) with no self-loop and $2 \leq \alpha \leq (W-1)^2$ (The case of $\alpha = 1$ results in a unidirectional ring and is of no interest to us). Considering a particular node in a given virtual graph, the maximum number of nodes which are one hop away from this node is α, and the maximum number of nodes which are two hops away is α^2, and so on. Therefore,

$$W \leq 1 + \alpha + \alpha^2 + \alpha^3 + ... + \alpha^D = \frac{\alpha^{D+1} - 1}{\alpha - 1}. \quad (7)$$

It follows that

$$D \geq \log_\alpha(W(\alpha-1)+1) - 1 \geq \log_\alpha W - 1. \quad (8)$$

Suppose we consider the case of equality holding for (7), then the average distance, denoted as $\overline{H}$, can be written as follows.

$$\begin{aligned} \overline{H} &= \frac{\sum_{i=1}^{D} i \cdot \alpha^i}{W-1} \\ &= \frac{D\alpha^{D+2} - (D+1)\alpha^{D+1} + \alpha}{(W-1)(\alpha-1)^2} \\ &\geq \frac{(\log_\alpha W)(\alpha-1) - \alpha}{\alpha^2}. \end{aligned} \quad (9)$$

As we mentioned before, the length of a virtual path plus one equals the length of the corresponding physical path in the resulting receiving graph. Thus, the average distance of a pair of nodes in a WTDM network, denoted as $\overline{H_{RG}}$ can be approximated as follows.

$$\overline{H_{RG}} \geq \frac{(\log_\alpha W)(\alpha - 1) - \alpha}{\alpha^2} + 1. \tag{10}$$

5.1 Average Network Throughput for High-Load

The *average network throughput for high-load*, denoted as $Thpt$, is defined as the number of packets which can be successfully delivered from a source node to a destination node in one slot. For a given receiving graph, if we assume the routing algorithm can always find the shortest length path, $Thpt$ can be approximated by the total exploited wavelengths W (since W packets are carried in each slot-time ; one per wavelength) divided by the average distance of a pair of nodes $\overline{H_{RG}}$. For simplicity we assume each intermediate node has infinite number of buffer spaces. Then we have

$$\begin{aligned} Thpt &= \frac{W}{\overline{H_{RG}}} \\ &\leq \frac{W\alpha^2}{(\log_\alpha W)(\alpha - 1) - \alpha + \alpha^2}. \end{aligned} \tag{11}$$

Let the above upper bound be denoted as $Thpt_u$ and the number of available wavelengths be denoted as W_{avl}. Meanwhile, recall that $N = C \times \alpha \times W$ and $2 \leq \alpha \leq (W - 1)$. Therefore, our objective is to

$$\text{Maximize } Thpt_u$$

$$\text{subject to } \begin{cases} N = C \times \alpha \times W, \\ 2 \leq \alpha \leq W - 1, \\ W \leq W_{avl} \text{ and } C \geq 1, \\ \text{where } N, W, \alpha \in Z^+ \text{ and } C \in R^+. \end{cases} \tag{12}$$

That is, for given N nodes and W_{avl} wavelengths, we would like to decide the number of stacks $\lceil C \rceil$ and a virtual graph with degree α and W nodes such that $Thpt_u$ is maximized.

From (11) we can observe that, for a fixed W, as α increases, the divisor, $\overline{H_{RG}}$, decreases. Thus $Thpt_u$ increases. For a given α, as W increases, both dividend, W, and divisor, $\overline{H_{RG}}$, increase. However, the growth of W is much faster than $\overline{H_{RG}}$. Thus, overall $Thpt_u$ increases as W increases. Intuitively, we should choose W and α as large as possible to maximize $Thpt_u$. This also implies that C should be as small as possible. Since $N = C \times \alpha \times W$ implies $\alpha \leq \frac{N}{W}$, α should be bounded by $\min\{W - 1, \frac{N}{W}\}$.

First we consider the case of $W_{avl} - 1 \leq \frac{N}{W_{avl}}$ (i.e., $W_{avl} \leq \sqrt{N}$ when N is large). This implies $W - 1 \leq \frac{N}{W}$ and $2 \leq \alpha \leq \min\{W - 1, \frac{N}{W}\} = W - 1$. To maximize $Thpt_u$, we should choose α as large as possible, that is, $\alpha{=}W - 1$. It follows that $C = \frac{N}{W(W-1)}$, and $Thpt_u \approx \frac{W^3}{W^2-1}$ (when W is large). Clearly, for this case, the larger the W is, the higher the α is. Therefore, $Thpt_u$ is maximized at $W = W_{avl}$, $\alpha = W_{avl} - 1$ and $C = 1$. Furthermore, the maximum $Thpt_u$ can be obtained in this range of W_{avl} when $W_{avl} - 1 = \frac{N}{W_{avl}}$ ($W_{avl} \approx \sqrt{N}$ when N is large) and $Thpt_u = \frac{N^{\frac{3}{2}}}{N-1} = O(\sqrt{N})$.

Next we consider the case of $W_{avl} - 1 > \frac{N}{W_{avl}}$ (i.e., $W_{avl} > \sqrt{N}$ when N is large). Since the range of $W \leq \sqrt{N}$ has been discussed before, the range of interest left is $W_{avl} \geq W > \sqrt{N}$. Within this range, it implies $W - 1 > \frac{N}{W}$ and $2 \leq \alpha \leq \min\{W-1, \frac{N}{W}\} = \frac{N}{W}$. Similarly, to maximize $Thpt_u$, α should be chosen as large as possible. That is $\alpha = \frac{N}{W}$. This will force C to be 1. $Thpt_u$ can be proven to be $O(\frac{N}{\log_{\frac{N}{W}} N - \frac{N}{W}})$. Clearly, this value is greater than the $Thpt_u$ in the previous case.

Since, there is a inversely linear relationship between α and W, increasing W may cause the reduction of α and vis versa. The α and W which maximize $Thpt_u$ are not easy to be determined at this point. However, they can be found by substituting α as $\frac{N}{W}$ into (11) to solve the W which maximizes $Thpt_u$. Then the corresponding α can be obtained. Unfortunately, it is fairly complicated to derive the exact formulas for such α and W. Thus, instead we use numerical computation to calculate these values. For example, in Table 5, we assume W_{avl} is as big as possible (i.e., $W_{avl} = N$) and list the best design choices in terms of C, α and W for different network sizes. C is very close to one in all cases. Surprisingly, the best values of α for various N are either 2 or 3. To be more precise it is 2 for small to moderate size networks (e.g., N=500 or 1,000) and 3 for large size networks (e.g., N=5,000 or 10,000). On the other hand, the best values of W are $\frac{N}{2}$ for small size networks and $\frac{N}{3}$ for large size networks.

N	(C, α, W)
500	(1, 2, 250)
1,000	(1, 2, 500)
5,000	(1.00004, 3, 1666)
10,000	(1.0001, 3, 3333)

Table 5: The optimal designs in terms of C, α, W for different N's.

In practice, the number of available wavelengths W_{avl} is several hundred (e.g., 256) and much less than $\frac{N}{3}$ (when N is in thousands). Therefore, in general, we should first increase W as close to W_{avl} as possible, and then choose α as large as possible. This will force C to be one. We summarize the optimal design strategies and the corresponding $Thpt_u$'s subject to different W ranges in Table 6. The general design principle can be stated as follows.

- First, exploit as many wavelengths as possible as long as it is no greater than $\frac{N}{2}$ for small N or $\frac{N}{3}$ for large N,

- Then choose degree α as large as possible (This will force C as small as possible).

Range of W_{avl}	$3 \le W_{avl} \le \sqrt{N}$	$\sqrt{N} < W_{avl}$
Best choice	$C = \frac{N}{W_{avl}(W_{avl}-1)}$, $\alpha = W_{avl} - 1$, $W = W_{avl}$	$C = 1$, $\alpha = \frac{N}{\min\{W_{avl}, \frac{N}{2}\}}$, $W = \min\{W_{avl}, \frac{N}{2}\}$ (for small N)
		$C = 1$, $\alpha = \frac{N}{\min\{W_{avl}, \frac{N}{3}\}}$, $W = \min\{W_{avl}, \frac{N}{3}\}$ (for large N)
$Thpt_u$	$\frac{W_{avl}^3}{W_{avl}^2 - 1} = O(W_{avl})$	$O(\frac{N}{\log_\alpha N - \alpha})$

Table 6: The optimal design choices for maximizing $Thpt_u$.

5.2 Average Minimum Packet Delay

We define the *packet delay* for a packet as the time period from being generated by a source node to arriving at a destination node. The *minimum packet delay* for a packet is the time delay including the necessary waiting time for transmissions, but no queuing in the intermediate nodes along the path. That is, once a packet arrives at a node, it can always be transmitted in the node's next transmission turn. We denote this value as L. Basically, L involves several factors, such as the transmission cycle length Δ, the subcycle length Δ_s, the average distance $\overline{H_{RG}}$ and the propagation delay. We assume the propagation delay, denoted as τ, is the same from any node to any other node, and the fast delivery routing algorithm proposed in the previous section is enforced. Envision that a network has very light traffic load and a node generates a packet to send. The first delay incurred by the node is waiting for its transmission turn. On average, it has to wait half cycle (i.e., $\frac{\Delta}{2}$ slots). After transmission, the signal takes τ slots to propagate to the receiver end. Upon receiving the packet, the node responsible for transmitting the packet then has to wait for its next transmission turn. Since the fast delivery routing is used and very light traffic load is assumed, the node will be scheduled for transmitting the packet within the next Δ_s-slot period. Therefore, on average it has to wait half subcycle (i.e., $\frac{\Delta_s}{2}$ slots). The same delay is needed for each intermediate node. Finally, the destination node receives the packet. No extra waiting time is needed in the destination node. In general, the average L can be approximated as

$$\begin{aligned} \text{Average } L &= \frac{\Delta}{2} + (\tau + \frac{\Delta_s}{2})\overline{H_{RG}} - \frac{\Delta_s}{2} \\ &\ge \frac{N}{2W} + (\tau + \frac{\alpha}{2})(\frac{\log_\alpha W(\alpha - 1) - \alpha + \alpha^2}{\alpha^2}) - \frac{\alpha}{2}. \end{aligned} \tag{13}$$

Let the above lower bound be denoted as L_l. Similarly, considering the constraints on parameters, the objective function can be specified as

$$\text{Minimize } L_l$$

$$\text{subject to} \begin{cases} N = C \times \alpha \times W, \\ 2 \le \alpha \le W - 1, \\ W \le W_{avl}, \\ C \ge 1 \text{ and } \tau \ge 0. \\ \text{where } N, W, \alpha \in Z^+, \tau \in Z \text{ and } C \in R^+. \end{cases} \tag{14}$$

That is, given N nodes, W_{avl} wavelengths and propagation delay τ, we would like to decide the values of α, W and C such that the corresponding receiving graph can result with the minimum L_l.

From (13), an observation can be made on α and W, respectively. First, for a given W, as α increases, the subcycle Δ_s in the second and third terms increases, but $\overline{H_{RG}}$ in the second term decreases. In general, the decreasing of $\overline{H_{RG}}$ is much faster than the increasing of α. That will make the second and third terms decrease , and so does L_l. Therefore, for a given W, we prefer to increase α as much as possible. Note that C is reduced as much as possible. Next, for a given α, as W increases, Δ in the first term decreases, but $\overline{H_{RG}}$ in the second term increases. Clearly, the decreasing of Δ is much faster than the increasing of $\overline{H_{RG}}$. However, since the propagation delay τ is also involved in the second term and if it is large, the value of the second term may be magnified. Intuitively, for a small τ, we prefer to enlarge the first term (i.e., choose W as large as possible) to reduce the cycle length. Thus L_l is reduced. This will force $C = 1$. On the other hand, as τ is getting larger, L_l becomes more sensitive to $\overline{H_{RG}}$. Therefore, we should choose a moderate W to maintain a small $\overline{H_{RG}}$ and, meanwhile without sacrificing too much on the cycle length. One extreme case is when τ is very large, then the second term dominates L_l. In this case, we should maintain the shortest $\overline{H_{RG}}$ to minimize L_l. Clearly, the shortest $\overline{H_{RG}}$ can be obtained when we embed a completely connected virtual graph (i.e., $W = \alpha + 1$). Essentially, this corresponds Bus-Mesh proposed in [18].

In order to find the design choice which results in the minimum L_l, as we did in the analysis of $Thpt_u$, two ranges of W_{avl} are examined, separately. For the range of $3 \le W_{avl} \le \sqrt{N}$, $2 \le \alpha \le W - 1$. To minimize L_l, α should be as large as possible, that is, $\alpha = W - 1$. Another range is $\sqrt{N} < W_{avl} \le \frac{N}{2}$. Thus $2 \le \alpha \le \frac{N}{W}$. To minimize L_l, again α should be as large as possible, that is, $\alpha = \frac{N}{W}$. Then the question which follows is "what is the value of W which minimizes L_l?" Again, it is hard to derive the mathematical formulas for such W, so we use numerical computation to analyze the behavior of L_l. Specifically, we examine the cases of $\tau \le 100$ slots, which, we believe, can cover most realistic environments. Assuming $W_{avl} = N$, we observe that in all cases the best W occurs within the range of $\sqrt{N} < W \le \frac{N}{2}$. Note that within this range the best $\alpha = \frac{N}{W}$ and $C = 1$. Moreover, we are interested in knowing what is the best α we should choose for a given τ. Then W and C can be determined too. This relationship can be derived by plugging $W = \frac{N}{\alpha}$ into (13). And then let $\frac{\partial}{\partial \alpha} L_l = 0$ to solve τ in terms of α and N. The function, denoted as $f_N(\alpha) = \tau$, will return a τ value which is most suitable for a given α. In other words, $f_N^{-1}(\tau)$ will return the best α for a given τ. Then the best $W = \frac{N}{f_N^{-1}(\tau)}$ and $C = 1$. For example, we show the curves of $f_N(\cdot)$ for N=500, 1000, 5000 and 10000 in Figure 8. As we predicted before, for small τ, the cycle length Δ is playing an important role, so we should increase W as large as possible. This will limit α to a small number. Note that this design strategy agrees with that of maximizing $Thpt_u$. On the other hand, as τ increases, the average distance $\overline{H_{RG}}$ becomes crucial, so W should be decreased and α should be increased. As can be seen that when τ is small, α's are pretty much the same for different sizes of network, but when τ becomes larger the differences of α become more significant.

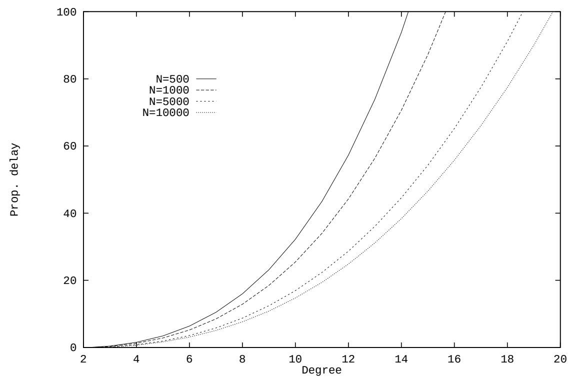


Figure 8: The curves of τ vs. the best α for different network sizes.

Note that for ShuffleNet[9], it was pointed out that for long propagation delay (e.g., τ=50 slots) 2-stage design is preferred, since the diameter is bounded by 3. However, the effect of cycle length is not considered. In Figure 8, we show that the best design may not necessarily occur at a short diameter (i.e., a large α value), instead at some moderate number. Clearly, $f_N(\cdot)$ is related to the topologies of the virtual graph as well as the routing algorithm. The one shown here is for the purpose of demonstrating the trade-off between parameters. In reality, $f_N(\cdot)$ can be derived for a specific topology like ShuffleNet. In summary, we outline the design strategies for different W_{avl} ranges and the corresponding L_l's in Table 7. In general, in order to minimize L_l for given $\tau(\leq 100)$, W_{avl} and N we should obey the following rules:

- First, figure out the best α corresponding to the given τ. That is, to compute $f_N^{-1}(\tau)$.
- Second, exploit as many wavelengths as possible as long as it is no greater than $\frac{N}{f_N^{-1}(\tau)}$.
- Third, choose α as large as possible. This also indicates that C should be as small as possible.

5.3 Examples of Regular Virtual Graphs

We have discussed the strategies for designing WTDM networks from a general virtual graph. In practice, there may exist no regular graph corresponding to the best choice of

Range of W_{avl}	$3 \leq W_{avl} \leq \sqrt{N}$	$W_{avl} > \sqrt{N}$
Best choice	$C = \frac{N}{W_{avl}(W_{avl}-1)}$, $\alpha = W_{avl} - 1$, $W = W_{avl}$	$C = 1$, $\alpha = \frac{N}{\min\{f_N^{-1}(\tau), W_{avl}\}}$, $W = \min\{f_N^{-1}(\tau), W_{avl}\}$
L_l	$O(\frac{N}{W_{avl}} + \tau)$	$O(\tau \cdot \frac{\log_\alpha N}{\alpha} + \log_\alpha N + \alpha)$

Table 7: The optimal design choices for minimizing L_l.

W and α. Different types of graphs may have various limitations on these parameters. However, we believe that the design principles described in the previous subsection can still be applied to special types of regular graphs. In this subsection, we choose two well-known types of regular graph, *m-ary n-cube* and *Shuffle- exchange* (Shuffle-exchange has also been referred to as de Bruijn graph in [24]), as examples to demonstrate this. These two types of regular graph have several desirable features, such as simple routing, small diameter, and flexibility of expanding.

The address i of a node in an *m-ary n-cube* can be represented by a radix-m n-digit number. That is, i can be denoted by $(i_{n-1}i_{n-2}...i_1i_0)_m$. Thus, an *m-ary n- cube* has m^n nodes and nodes are connected according to the following rule. Any pair of nodes are directly connected if their addresses differ in only one digit. The node addresses along a path can be viewed as a sequence of addresses in which two consecutive addresses differ in only one digit. If the addresses of two nodes differ in k digits, a shortest path connecting these two nodes can be obtained by going from one node to the next (starting from the source node) and each move reduces the number of distinct positions between the addresses of the current and destination nodes by one. Since there are k distinct positions between the addresses of the source and destination nodes, after k moves the destination node should be reached. There are k different nodes that can be moved to from the current node if the number of distinct positions between the addresses of the current node and the destination node is k. We have $k!$ alternative shortest paths in total. Also the addresses of any two nodes may differ at most in n digits, so the diameter is n. Moreover, consider one digit of the address of a node. There are $(m-1)$ other nodes and their addresses differ in that digit with the address of the node. Since there are n digits in total, the degree of each node α is equal to $(m-1) \times n$. The corresponding receiving graph has $C \times (m-1) \times n$ physical nodes in a receiving node and the total number of physical nodes N is

$$N = C \times \alpha \times W = C \times (m-1) \times n \times m^n. \quad (15)$$

Assuming N is of moderate size, N wavelengths are available and our design goal is to maximize $Thpt_u$, according to the strategy mentioned in the previous subsection, we should first exploit the number of wavelengths W as close to $\frac{N}{2}$ as possible. It can be proven that the maximum W (i.e., m^n) occurs when $m = 2$ and $C = 1$. In this case,

$\alpha = (m-1) \times n = n$, $W = m^n = 2^n$, and $N = n \times 2^n$. Clearly, 2^n is much smaller than $\frac{N}{2}$. From another point of view, the maximum $Thpt_u$ occurs at $\alpha = 2$. However, α can be at best reduced to n for an m- ary n-cube. Therefore, the best throughput that a WTDM based on an m-ary n-cube can achieve falls below the theoretical upper bound.

Likewise, the address of a node in a *Shuffle-exchange* can be represented by a radix-m n-digit number, and in total there are m^n nodes. However, node $a{=}(a_{n-1}a_{n-2}...a_1a_0)_m$, has a direct link to node $b{=}(b_{n-1}b_{n-2}...b_1b_0)_m$, if and only if $a_{n-2}a_{n-3}...a_0$ equals $b_{n-1}b_{n-2}...b_1$. Thus, every node has m outgoing degree. Essentially, a path is corresponding to a sequence of node addresses in which a node address (except the source node) comes from shifting out the leftmost digit of the predecessor's address and padding in an appropriate digit value to the right. To find the shortest path is more complicate than that of m-ary n-cube, but still can be performed in $O(n)$ (see [24]). It can be proven that there is only one unique shortest path between any pair of nodes. In the worst case, two node addresses differ in n digits. Thus, diameter is also n. Since each node has m outgoing degree, the corresponding receiving graph will have $C \times m$ physical nodes in each receiving node and the total number of physical nodes N is

$$N = C \times \alpha \times W = C \times m \times m^n. \tag{16}$$

Again, assuming N is of moderate size, $W_{avl} = N$ and the design goal is to maximize $Thpt_u$. The largest W we can obtain is when $C = 1$ and $m = 2$. In that case, $W = m^n = 2^n$, $\alpha = m = 2$ and $N = 2^{n+1}$. In fact, this is the best design to maximize $Thpt_u$. For this reason, we conclude that Shuffle-exchange virtual graph is more preferred than m- ary n-cube virtual graph in terms of design flexibility. However, if the number of available wavelengths is much less than $\frac{N}{2}$, we might consider using m-ary n-cube because of its easier routing algorithm and provision of multiple alternative shortest paths.

6 Conclusion and Discussion

In this paper we have focused on the optical networks which are based on passive star couplers. In order to reduce the hardware interface cost and to exploit higher bandwidth, each station is assumed to have only one fixed wavelength transmitter and one fixed wavelength receiver. We have demonstrated the tradeoffs between different design parameters and also proposed a systematic way to design optimal WTDM networks. In the following we will briefly discuss two other design issues related to this type of optical network: 1) how to reduce propagation delay for multi-hop optical networks, and 2) how to perform dynamic bandwidth allocation.

In any network environment, propagation delay is an important factor which influences network performance. With increasing channel bandwidth and higher operational speed of interface devices, the communication delay between a pair of nodes will be dictated by propagation delay. This delay may become unacceptable in multihop optical networks since the traversal of several hops may be required before a packet reaches to its final destination [19].

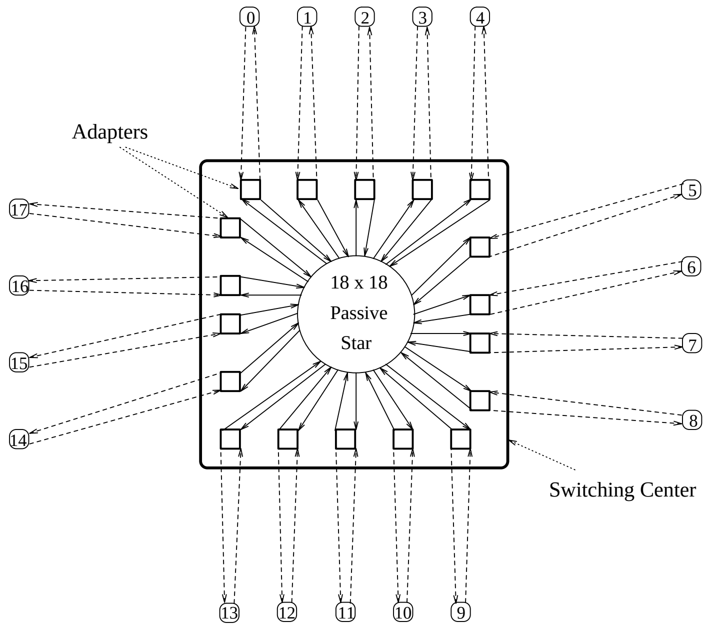


Figure 9: An 18-node local area network with a central passive star switch.

Furthermore, with a passive star network, each node is located at a different distance from the center of the star, so the propagation delay from each node to the star is different. In order to properly operate WTDM media access protocol, a method of signal synchronization, which guarantees non-conflicting arrivals of several signals at the optical star, is needed. Nevertheless, when the network covers a wide area and involves thousands of nodes, such global synchronization is a formidable task [19].

Because of these reasons, we can take the *Centralized Passive Star Switch Approach* proposed in [19]. The architecture of a centralized passive star switch is shown in Figure 9. The transmitters and receivers are separated from nodes and placed within a *switching center*. Dedicated wires (either electronic or optical) are installed between the nodes and the switching center. There is an adapter connected to each wire. An adapter is the interface between a node and the passive star which is resided in the switching center. It also performs all routing decisions within the switching center. In this way, a packet is relayed within the switching center without going through the nodes, so the packet delay caused by signal propagation can be greatly reduced. Moreover, the synchronization of time slots becomes much easier since it can be performed within the centralized switching center. According to the design strategy of minimizing L_l, the small propagation delay within the switch center suggests embedding a virtual graph with as many nodes (i.e., wavelengths) as possible. This also improves $Thpt_u$.

Dynamic bandwidth allocation is another important feature. It is especially desirable in a skewed traffic load situation. Multihop WTDM networks can provide a certain degree of dynamic bandwidth allocation through exploiting alternative paths. However, if the number of alternative paths is limited, the amount of channel bandwidth that can be reallocated is also limited. Moreover, in some cases (e.g., virtual circuit), it is desirable to ask all packets to follow the same path to avoid resequencing overhead at the destination node. Therefore, another way of exploiting higher bandwidth for a node is needed.

One possible way (which requires one extra fixed wavelength receiver in each node) is to allow dynamic bandwidth sharing among the nodes in the same transmitting group. As long as a slot is unused by a node, other nodes in the same transmitting group can use it on a contention basis. In order to sense the status (including un-used or collision) of a slot, *one extra fixed wavelength receiver* tuned to the transmitter's wavelength is required in each node. Several dynamic bandwidth allocation Time-Division Multiplexed protocols based on a single broadcasting bus have been proposed [25] and they can be adopted in this type of WTDM networks directly.

References

[1] A.S. Acampora, M. J. Karol, and M. G. Hluchj, "Multihop Lightwave Networks: A New Approach to Achieve Terabit Capabilities," *Pro. ICC'88*, Vol.1, pp.1478-1484, 1988

[2] W.E. Burr, "The FDDI Optical Data Link," *IEEE Comm Magazine*, Vol.24, No.5, May 1986.

[3] *IEEE P802.6/D12*, "Distributed Queue Dual Bus (DQDB) subnetwork of a Metropolitan Area Network (MAN)," pp. 867-876, Feb. 7,1990.

[4] R. Ballart, Y.C. Ching, "SONET: Now It's the Standard Optical Network," *IEEE Comm. Magazine*, pp. 8-15, March 1989.

[5] C.A. Brackett, "Dense Wavelength Division Multiplexing Networks: Principles and Applications," *IEEE Journal on Selected Areas in Comm.*, Vol. 8, no. 6, August 1990.

[6] N. R. Dono et al., "A Wavelength Division Multiple Access Network for Computer Communication," *IEEE Journal on Selected Areas in Comm.*, Vol. 8, no. 6, pp. 983-994, August 1990.

[7] E. Arthurs et al. , "Multiwavelength Optical Crossconnect for Parallel Processing Computers," *Electron. Lett.*, Vol 24, pp. 119-120, 1988.

[8] E. Arthurs, M.S. Goodman, H. Kobrinski, and M.P. Vecchi, " HYPASS: An Optoelectronic Hybrid Packet-Switching System," *IEEE Journal on Selected Areas in Comm.*, Vol. 6, pp. 1500-1510, 1988.

[9] M. G. Hluchyj and Mark J. Karol, "ShuffleNet: An Application of Generalized Perfect Shuffle to Multihop Lightwave Networks," *IEEE Journal of Lightwave Technology*, vol.9, no. 10, pp. 1386-1396, Oct 1991.

[10] Patrick W. Dowd, "Random Access Protocol for High-Speed Interprocessor Communication Based on an Optical Passive Star Topology," *IEEE Journal of Lightwave Technology*, Vol. 9, no. 6, pp. 799 - 808, June 1991.

[11] Imrich Chlamtac and Aura Ganz, "Toward Alternative High- Speed Network Concepts: The SWIFT Architecture," *IEEE Trans. on Comm.*, Vol. 38, no. 4, pp. 431 - 439, April 1990.

[12] Imrich Chlamtac and Aura Ganz, " Channel Allocation Protocols in Frequency- Time Controlled High Speed Networks," *IEEE Trans. on Comm.*, Vol. 36, no. 4, pp. 430 - 440, April 1988.

[13] Aura Ganz and Zahava Koren, "WDM Passive Star - Protocol and Performance An analysis," *IEEE Infocom.*, Vol. 2, pp. 9A.2.1 - 9A.2.10, 1990.

[14] M.S. Chen, N. R. Dono, and R. Ramaswami, "A Media-Access Protocol for Packet-Switched Wavelength Division Multiaccess Metropolitan Area Networks," *IEEE Journal on Selected Areas in Comm.*, Vol. 8, no. 6, pp. 1048 - 1057, August 1990.

[15] G.N.M. Sudhakar, N. D. Georganas, and M. Kavehard, " Slotted Aloha and Reservation Aloha Protocols for Very High-Speed Optical Fiber Local Area Networks Using Passive Star Topology," *IEEE Journal of Lightwave Technology*, Vol. 9, no. 10, pp. 1411 - 1422, Oct. 1991.

[16] N. Mehravari, "Performance and Protocol Improvements for Very High Speed Optical Fiber Local Area Networks Using a Passive Star Topology," *Journal of Lightwave Technology*, Vol. LT-8, no. 4, April 1990.

[17] I. M. I. Habbab, M kavehrad, and C.E.W. Sundberg, " Protocol for Very High Speed Optical Fiber Local Area Networks Using a Passive Star Topology," *Journal of Lightwave Technology*, Vol. LT-5, no. 12, Dec 1987.

[18] T. Q. Dam, K.A. Williams, and D.H.C. Du, "A Media-Access Protocol for Time and Wavelength Division Multiplexed Passive Star Networks," to appear in *IEEE Journal on Selected Areas in Communications.*

[19] S. Bhattacharya, D.H.C. Du, A. Pavan, S.R. Tong, R. Vetter, and K. Williams, "A Network Architecture for Distributed High Performance Heterogeneous Computing," *Technical Report, U. of Minnesota*, 1992, Submitted to IEEE Computer.

[20] M. Goodman *et al.*, "The LAMDANET Multiwavelength Network: Architectures, Applications, and Demonstrations," *IEEE Joural on Selected Areas in Communications*, Vol. 8, No. 6, August 1990.

[21] M. Goodman, "Multiwavelength Networks and New Approaches to Packet Switching," *IEEE Communication Magazine*, Oct. 1989.

[22] T. Lee, M. Goodman, and E. Arthurs, "A Broadband Optical Multicast Switch," *ISS'90*, 1990.

[23] K. Eng, "A Photonic Knockout Switch for High-Speed Packet Networks," *Globecom'87*, pp.1861-1865. 1987.

[24] K. Sivarajan and R. Ramaswami, "Multihop Lightware Network Based on de Bruijn Graph," *IEEE INFOCOM 1990*, Vol.2, June 1990.

[25] Andrew S. Tanenbaum, *Computer Networks*, 2nd Edition, Prentice-Hall, pp. 179-182, 1989.

[26] T. Cox, F. Dix, C. Hemrick, J. McRoberts, "SMDS: The Beginning of WAN Superhighways." *Data Communications*, April 1991.