COMPARISON OF VARIOUS PHOTONIC PACKET SWITCH ARCHITECTUES UNDER RANDOM AND BURSTY TRAFFIC

by

AMIT SINGHVI

DEPARTMENT OF ELECTRICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY, KANPUR

June, 2006

COMPARISON OF VARIOUS PHOTONIC PACKET SWITCH ARCHITECTUES UNDER RANDOM AND BURSTY TRAFFIC

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Technology

by

AMIT SINGHVI

to the

DEPARTMENT OF ELECTRICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY, KANPUR

June, 2006

CERTIFICATE

It is certified that the work contained in the thesis entitled **"Comparison of various photonic packet switch architectures under random and bursty traffic"** has been carried out by **Mr. Amit Singhvi (Roll No. Y4104006)** under my supervision and the work has not been submitted elsewhere for a degree.

Dr. Y.N. Singh
Thesis Supervisor
Electrical Engineering
I.I.T. Kanpur

June, 2006

Abstract

Many different photonic packet switch architectures have been proposed to facilitate the widespread deployment of photonic packet switched networks. This thesis discusses a selected number of switch architectures with traveling type buffers and makes a comparison between them in terms of the number of components required, packet loss probability, and latency. Simulations have been done to compare the packet loss probability and latency of different switch architectures.

Except for the staggering switch, packet loss probability has been simulated using random traffic model in all switch architectures. In this model, traffic on each input of the switch is assumed to be uncorrelated in time for simplicity although it is unrealistic for high speed services. Later on, time correlation of traffic on each input of the switch has been taken into account by a more realistic bursty traffic model. The simulations have been carried out to study the effect of burstiness on the switch performance.

Finally, the scheduling algorithm may have a considerable effect on the performance of the switch, both in the probability of loss and latency. Simulations show the performance improvement in the staggering switch with an optimal scheduling algorithm.

ACKOWLEDGEMENT

This thesis is dedicated to my parents and my brother.

I would like to express deep gratitude to my supervisor Dr. Y.N. Singh for his invaluable guidance and encouragement which inspired me throughout. He continuously encouraged me through his valuable suggestions and constructive criticism.

I would also like to thank Rajiv Srivastava and Rajat K. Singh for their help and motivation in this thesis.

Last but not the least, I would like to thank my friends Aamir, Vipin, JP, Prem, Sunil and friends of Hall-VII, who helped me in times of crisis and supported me in the times of need and made my stay at IIT Kanpur a memorable one. I wish them all a great future.

June, 2006 Amit Singhvi

Contents

List of figures

List of tables

Chapter 1

Introduction

1.1 Introduction

The growth of existing and new broadband services will continue to increase the traffic flow in the telecommunication networks and thus push for larger and larger bandwidth demand. The telecommunication networks are generally build using various communication media like twisted pair copper wire, coaxial cable, wireless and optical fiber. The optical technology for the telecom, at the moment, primarily runs on optical fiber. The optical fiber has a very large bandwidth of the order of 40 THz. This bandwidth is generally a small fraction of carrier frequency used in optical networks which is of the order of 10^{15} Hz. Optical fiber has low loss in 1.3-1.55 μm band. Currently, this window is used for data transmission in telecom network. Some other advantages of fiber are large bandwidth-distance product, immunity to noise and interference, very low cost per unit bandwidth. Due to these advantages, optical fiber network can be used for providing high bandwidth services.

1.2 Telecommunication networks

Telecommunication networks basically consist of two portions, access network and backbone network (Fig. 1.1). The backbone network connects one access network to another. It consists of routers or switches interconnected by long distance links. These links can be satellite, microwave or optical fiber based. As shown in Figure 1.1, the nodes A and B can communicate with each other through switch or router.

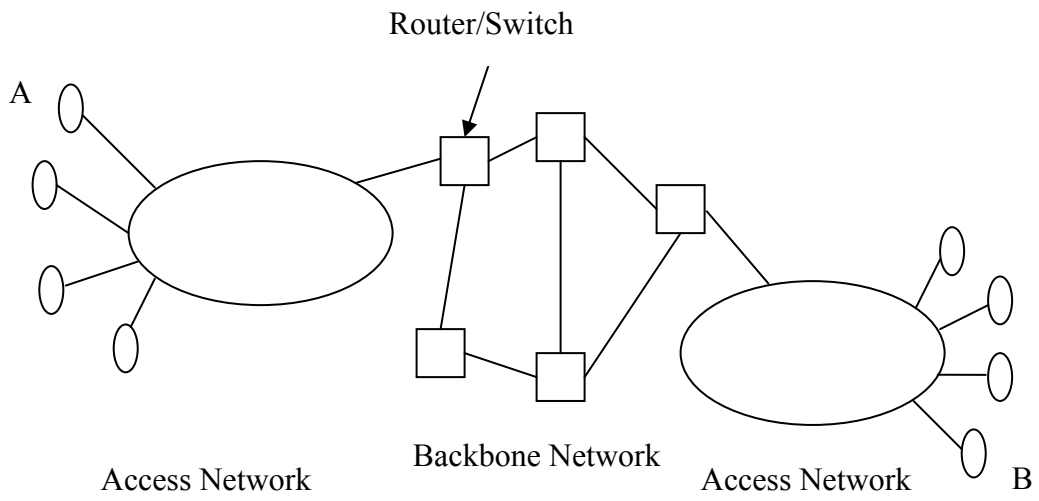

Figure 1.1 Telecommunication Network

1.3 Packet switching

Packet switching is a technique used to send data from one place to another. In packet switching the message is broken into data packets of fixed size with some header bytes added before transmission. Header contains information such as destination address, sender address, size of payload etc. The data packets of message are transmitted from one node to another till it reaches its destination. The packet switching does not require a dedicated link and reserved bandwidth. Depending on the network and available bandwidth on different link, any packet may take several possible routes to reach the destination. In such a case, a sequence number is included in the packets to ensure that packets will not be misinterpreted if they arrive out of order.

1.4 Optical packet switching

When a network is made completely of optical fiber links, it is called optical network. In these networks, the switches or routers are generally electronic devices. At the switch/router input, data has to be converted from optical to electric (O/E) form, and at the output, the data is converted back from electrical to optical form. These O/E and E/O conversions constrain the speed of the network. To overcome this constraint, one possibility is to use an all-optical router/switch. All-optical switching will require an optical

implementation of all the switch functions. However, the control and processing logic implementations are not optically feasible.

In near future, photonic packet switches will also be used, where packet switching and buffering is done optically while controlling is done in electronic domain. This approach will provide very large throughput and transparency of photonic devices combined with the functionality and processing power of electronic control circuit. At each photonic packet switch node in the network, the optical packet headers will be received and electronically processed to provide routing information to the control electronics. The control will then properly configure the node resources to switch packet payloads directly in the optical domain without undergoing any O/E or E/O conversions. In such a network there will be no electronic processing in the data path and the data can ideally pass through the switch without any limitation on bit rate.

1.4.1 Optical packet format

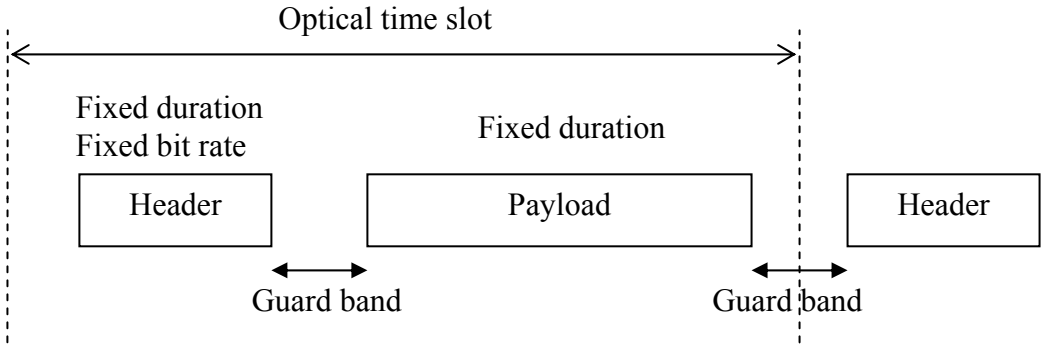

Figure 1.2 Format for packet

The optical packets are placed into a fixed duration time slot and consist of a header and a payload (Fig. 1.2). The header contains routing information and other control information. As the header has to be processed at each switch node, the header has a relatively low fixed bit rate suitable for electronic processing (say, less than 10 Gbps), while the payload could have a variety of bit rates ranging from 10 Gbps to 100's of Gbps. The use of fixed length packets can significantly simplify the implementation of packet contention resolution and buffering, packet routing as well as packet synchronization.

Guard bands allow the setup of the photonic devices (for synchronization and switching) and also reduce timing jitter.

1.5 Thesis organization

The first chapter introduced optical switching. Second chapter describes photonic packet switch and photonic components. Third chapter describes the various switch architectures and compare them on the basis of packet loss probability, latency and their component counts. The bursty traffic performance of different switch architectures is also discussed. Fourth chapter shows the performance improvement in staggering switch with an optimum scheduling algorithm. Fifth chapter presents the simulations and results. In chapter 6, conclusions and future work are given.

Chapter 2

Overview of photonic packet switch

2.1 General structure of photonic packet switch

Figure 2.1 shows the schematic diagram of a photonic packet switch. The key functions [1] affecting its operation and implementation are

1) Packet synchronization, 2) Contention resolution, 3) Packet buffering 4) Packet routing, 5) Control, and 6) Packet header replacement

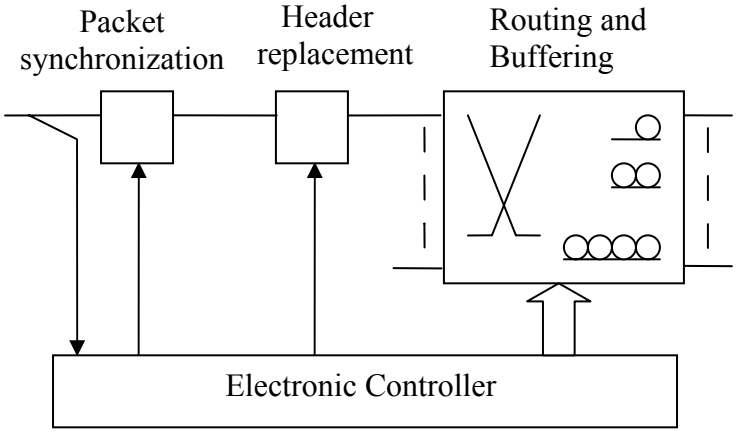

Figure 2.1: Schematic of a generic photonic packet switch

2.1.1 Synchronization

Photonic packet-switched networks can be divided into two categories: slotted (synchronous) and unslotted (asynchronous). When individual photonic switches are combined to form a network, packets can arrive at different times at the input ports of each node. In a slotted network, all the input packets arriving at the input ports should have the same size and slots should be aligned in phase. In an unslotted network, packets may or may not have the same size and packets may arrive and enter the switch without being aligned.

Slotted operation simplifies the implementation of packet contention resolution, and subsequent buffering and routing. A general implementation of packet synchronization [2] required for slotted operation is shown in Figure 2.2. It consists of a packet start recognizer and a programmable delay line module. The programmable delay line is a series of 2 × 2 optical switches interconnected by fiber delay lines of different length. The function of the packet start recognizer is to identify the packet start. Once the packet header has been recognized and packet delineation carried out, the packet start time is identified and the control electronics can calculate the necessary delay and configure the correct path of the packet through these delay lines. The lengths of delay line are arranged in an exponential sequence. The first delay line is equal to 1/2 time slot duration, the second delay line is equal to 1/4 slot duration, and so on. This scheme has a resolution $1/2^N$, where T is the time slot duration and N is the number of delay lines stages.

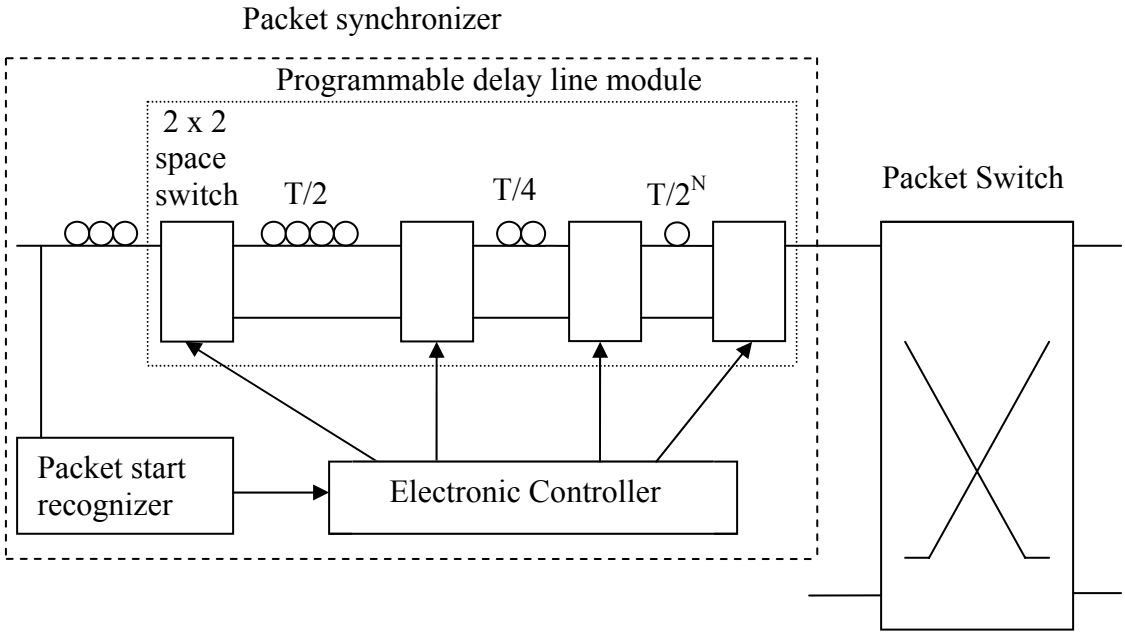

Figure 2.2: Schematic of packet synchronization

2.1.2 Contention resolution

At the input of a switch, it is possible to receive more than one packet in a single time slot, which needs to be forwarded to same output port. Since only one packet can be transmitted to an output port at a given time, additional packets need to be buffered for resolving the contention. Thus, high throughput and switching efficiency in the switching

network can be achieved. In addition to buffering, switches can block, drop or deflect the packet to control traffic flow and resolve contention [3].

2.1.3 Buffering

As discussed above, buffering of photonic packets somewhere within the switch fabric is essential to prevent packet contention. Advanced electronic RAM's can be used for packet buffering. However, electronic RAM's have a limited access speed, which will eventually constrain the speed and capacity of photonic packet switching system. In addition, this approach requires optical to electrical (O/E) conversion and vice versa when packets are written into and read out of electronic RAM's and hence adds to the complexity. All-optical RAM suitable for photonic packet switching has not been yet developed. The alternative is to use optical fiber delay-lines [4] which are simply a suitable length of fiber. In general, these optical fiber delay lines based buffers can be classified into two basic categories: traveling type and recirculating type. Figure 2.3 shows the various types of fiber delay-lines buffer.

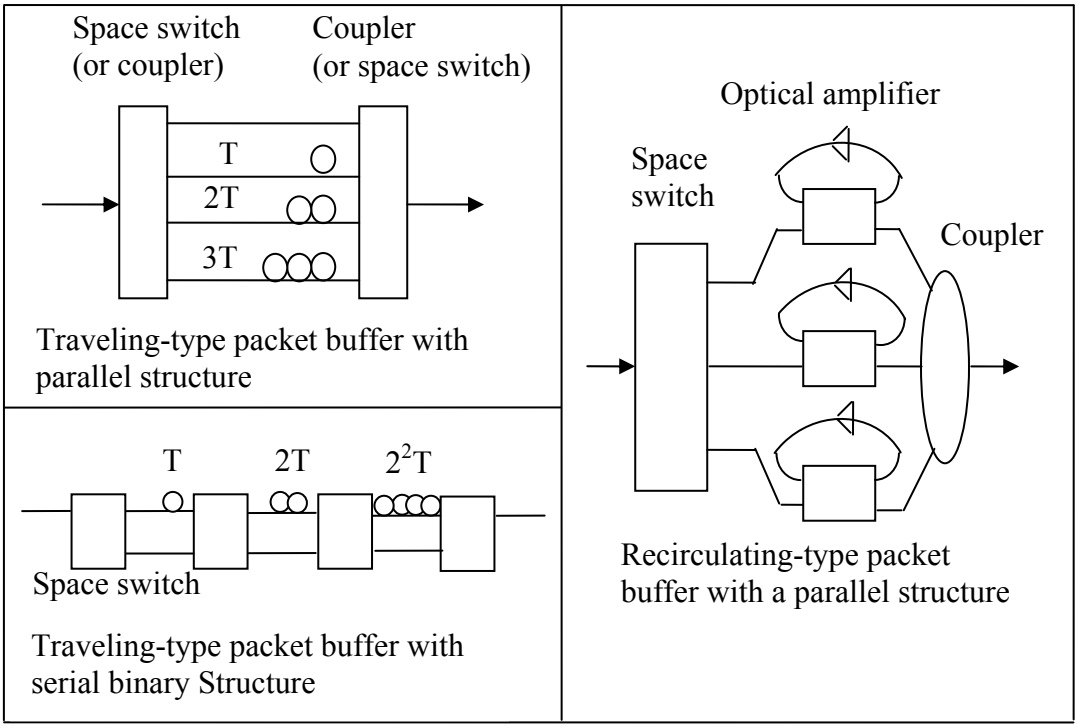

Fig. 2.3: Various types of buffers

A traveling type buffer generally consists of multiple optical fiber delay lines whose lengths are equivalent to multiples of packet duration T, and optical switches to select delay lines. The recirculating type buffer is more flexible than the traveling type buffer because the packet storage time is adjustable by changing the number of circulations. However, a problem with recirculating type buffer is that the signal has to be amplified during each circulation to compensate for the power loss. This results in accumulated amplified spontaneous emission (ASE) noise from the optical signal-level fluctuation, which limits the maximum buffering time. Therefore, the traveling type buffer is superior over the recirculating type buffer in terms of signal quality.

The fiber length required to delay the packet by one time slot can be illustrated by the following example

Effective Refractive index in fiber, $n \sim 1.5$

Speed of light in fiber = $c / n \sim 2\text{x}10^8$ m/s

Let number of bytes, b in a packet = 1024 bytes = 8192 bits ~ 9000 bits (overhead bits, synchronization bits etc.) and transmission rate, T = 1 Gbps,

Fiber length required to delay the packet by one slot, l

$$l = \frac{c}{n} \times \frac{b}{T} = 1.8 \text{ km}$$

and slot period = 9 μs

2.1.4 Routing

The packets are routed from switch inputs to switch outputs in a distributed network. Packet headers are carried with individual packets, separated from the data at each switch, and processed to set the correct switch state.

2.1.5 Control

Here, header will be analyzed to search the routing database to determine the output port to which packet will be forwarded. The main function of control unit is to search routing database, decide on outgoing port and then configure various components so that packet is switched to the designated output port. The other tasks assigned to control unit are tracking buffer status, input/output port status, synchronized operation of switch

components, synchronization with master clock, communication with other switching nodes, and packet-scheduling algorithm etc.

2.1.6 Packet header replacement

Packet header replacement is an important function in networks when virtual circuits are used. Here, the routing information in the packet header is contained as virtual circuit identifier (VCI). When packets traverse a switch node, the VCI is used to identify the output port of the switch by looking up a local forwarding table. To make the forwarding table as small as possible and the packet header as short as possible, the VCI contained in the packet header should only have a local meaning. This means that the packet header has to be replaced by a new one before a packet is delivered to the next node.

In the context of photonic packet switching, optical replacement of packet header is sometimes necessary. Figure 2.4 shows an implementation of packet header replacement [5].

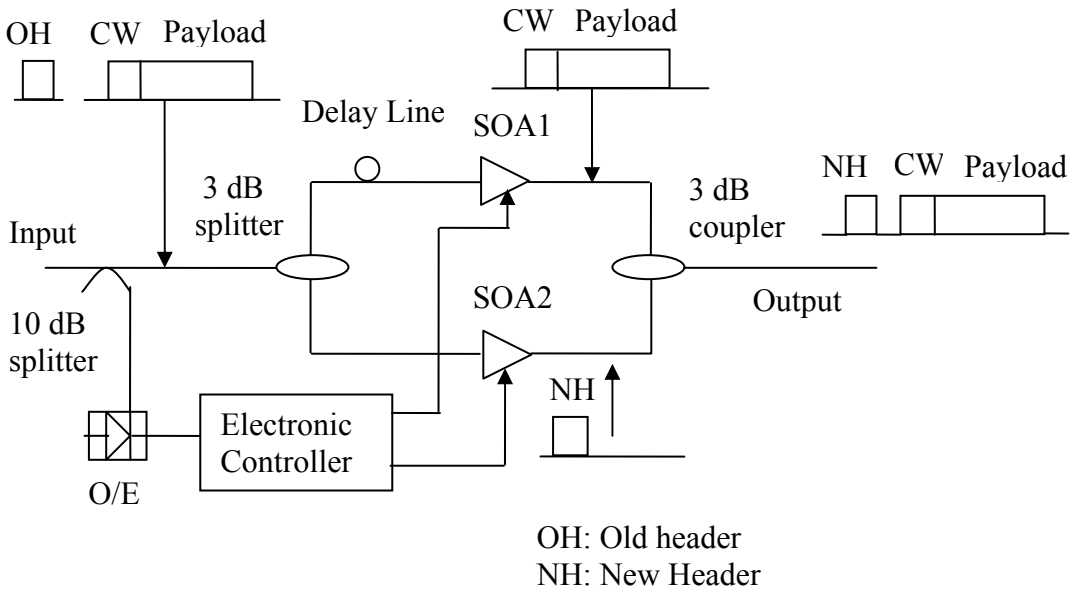

Fig 2.4: Technique for packet header replacement

This technique introduces a continuous wave (CW) portion between the header and payload. During the packet header period, SOA1 is turned off so as to remove the old packet header, while SOA2 is used to modulate a copy of the CW portion with the new header data and to block the other portions of the data. An optical delay line of a length

equal to the packet header time is inserted before SOA1 to adjust the timing so that the new packet header exiting SOA2 is exactly in place of the old packet header.

2.2 Basic photonic components needed in the architectures under consideration

2.2.1 Couplers [6,7]

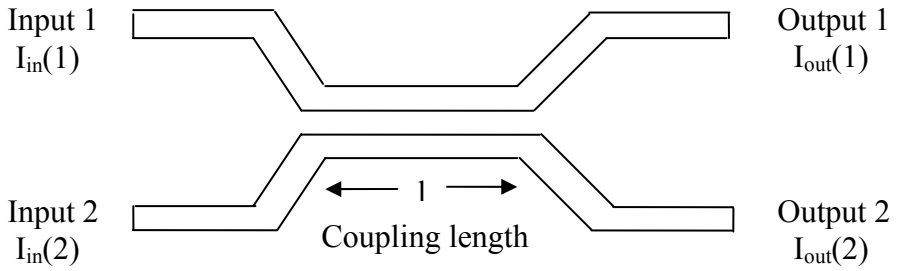

Figure 2.5: Schematic of coupler

A directional coupler is used to combine and split signals in an optical network. It can be constructed by fusing two fibers together in the middle (Fig. 2.5). Such a coupler takes a fraction α of the power from input 1 and places it on output 1 and the remaining fraction 1-α on output 2. Likewise, a fraction 1-α of the power from input 2 is distributed to output 1 and the remaining power to output 2. Thus, the relationship between the input power and output power can be given by the following equations

$$I_{out}(1) = \alpha I_{in}(1) + (1-\alpha) I_{in}(2), \tag{2.1}$$

$$I_{out}(2) = \alpha I_{in}(2) + (1-\alpha) I_{in}(1). \tag{2.2}$$

The power coupled from one fiber to the other can be varied by changing the coupling length, l. In 3 dB couplers, α is 1/2 so that half the power from each input appears at each output.

An $N \times N$ star coupler is an N-input, N-output coupler in which power from each input is divided equally among all the outputs. It can be constructed by suitably interconnecting a number of 3-dB couplers.

Couplers are the building blocks used to construct a variety of optical devices such as optical switches, optical filters and multiplexers/demultiplexers.

2.2.2 Multiplexers and Demultiplexers [6]

A WDM Mux combines different wavelengths into one fiber and a WDM Demux does the opposite. Based on inter-channel spacing, WDM Mux/Demux are classified as WDM and Dense WDM. WDM Mux/Demux is generally based on array waveguide grating or interferometric principles. A Mux/Demux is a wavelength specific device, such that it is designed for specific input/output wavelengths at each port. For example, let a Mux be designed for multiplexing N wavelengths λ_1 to λ_n at N input ports. This device would multiplex wavelengths from the specified group only when they are available at the designated input ports. Similarly in WDM Demux, individual wavelengths and their output ports are specified.

2.2.3 Semiconductor Optical Amplifier (SOA) switch [6]

The SOA can be used as an on-off switch by varying the bias voltage to the device. If the bias voltage is reduced, no population inversion is achieved, and the device absorbs input signals. If the bias voltage is present, it amplifies the input signals. The combination of amplification in the on-state and absorption in the off-state makes the device capable of achieving very large extinction ratios which is the ratio of output power in on-state to the output power in off state. The switching speed of the SOA switch is of the order of 1 ns.

2.2.4 Tunable Wavelength Converter (TWC) [6,8]

Wavelength conversion is an important function in optical networks. This wavelength conversion can be done either in electrical or optical form. There are many optical techniques for the conversion of the wavelength. Here, Four Wave Mixing (FWM) based TWC is considered (Fig. 2.6).

FWM is a good candidate for fully transparent wavelength conversion as it preserves both phase and amplitude. It also allows simultaneous conversion of a set of multiple input wavelengths to multiple output wavelengths by shifting all of them in frequency space.

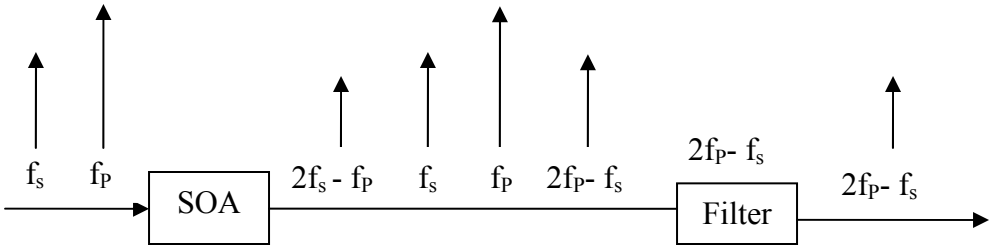

Figure 2.6: Schematic of F.W.M. wavelength conversion

FWM occurs due to nonlinearities in the transmission medium. It causes three waves at frequencies f_1, f_2 and f_3 to produce a fourth wave at frequency $f_1 + f_2 - f_3$. If $f_1 = f_2$, a wave at frequency $2f_1 - f_3$ is obtained. In optical fibers, the generated FWM power is quite small. For the purpose of wavelength conversion, the FWM power can be enhanced by an SOA because of higher intensities within the device. Thus, if we have an input signal at frequency f_s and a probe at frequency f_p, then FWM will produce signals at frequency $2f_p - f_s$ and $2f_s - f_p$, if these frequencies lie within the amplifier bandwidth. The other waves must be filtered out at the SOA output.

2.2.5 Arrayed Waveguide Grating (AWG) [6,9,10]

This device consists of two star couplers, interconnected by an array of waveguides (Fig. 2.7). Each grating waveguide has a precise path difference with respect to its neighbors, ΔX, and is characterized by a refractive index of value n_W.

Once the signal enters the AWG from an incoming fiber, the input star coupler divides the power among all waveguides in the grating array. As a consequence of the difference of the guides lengths, light traveling through each couple of adjacent waveguides emerges with a phase delay difference given by

$$\Delta\phi = 2\pi n_w \frac{\Delta X}{\lambda}, \qquad (2.3)$$

where λ is the incoming signal central wavelength. As all the beams emerge from the grating array they interfere constructively onto the focal point in the star output coupler, in a way that allows to couple each interference maximum with a particular output fiber, depending only on the input signal central wavelength.

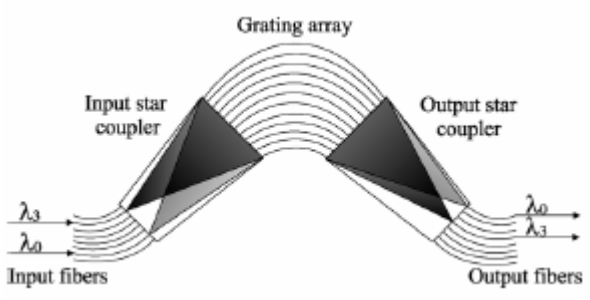

Figure 2.7: Schematic of Arrayed Waveguide Grating

As an example shown in Fig. 2.7, two signals of wavelength λ_3 and λ_0 entering an 8 × 8 AWG, from input fibers 2 and 7, respectively emerge out from the fibers 1 and 4. This permutation of the signals is dependent only on the wavelength and input port of the signals.

An AWG can be used as an *N* × *1* wavelength multiplexer, *1* × *N* wavelength demultiplexer and static router. A possible routing pattern of a static router constructed from an AWG is shown in the Figure 2.8.

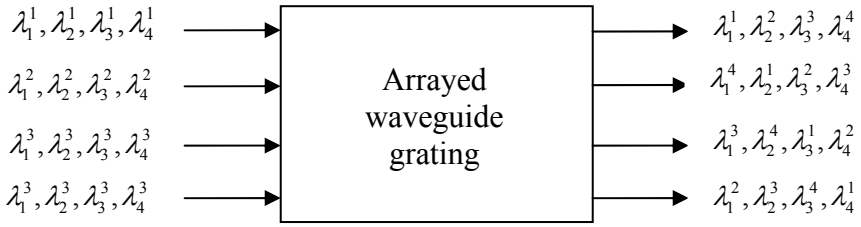

Figure 2.8: Routing pattern of a static router constructed from an AWG

The above figure shows a 4 × 4 AWG router using four wavelengths with each wavelength routed from the inputs to each of the four outputs.

2.2.6 Optical space switch [6,11,12]

Many different technologies are available to realize optical space switches. SOA based gate switches can provide high switching speed and large extinction ratios. Thus, optical space switches made of SOA gates are most promising in the context of photonic packet switching. SOA based optical space switch are constructed using couplers and SOA switches (Fig. 2.9). SOA are used as gates that let the signal go through or stop, to achieve

the required switching of optical signals. SOA switch also allows the amplification of signals, making it possible, besides switching functionalities, to restore a required given signal level.

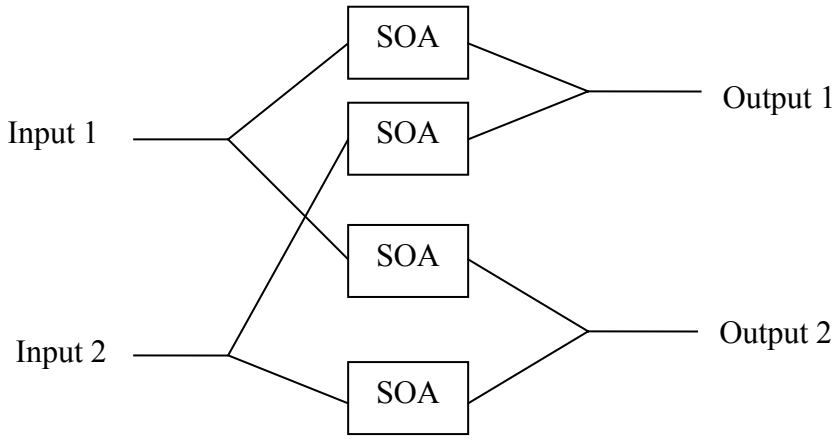

Figure 2.9: 2 x 2 SOA space switch

Another feature of this SOA space switch is that a packet can be broadcast to all the outputs or multicast to several outputs by appropriately turning SOA gates on or off. Also, multiple packets can appear at the same output, provided that they have different wavelengths.

Chapter 3

Comparison of different photonic packet switch architectures

3.1 Different photonic packet switch architectures

Many different photonic packet switch architectures have been proposed in literature [1,13,14] to facilitate the widespread deployment of photonic packet switched networks. In this chapter, a selected number of switch architectures with traveling type buffers have been discussed and a comparison is done between them. In all these switches the following are assumed

- All the packets arriving at the switch are of fixed size.
- Time is divided into equal timeslots, each containing one packet.
- The arrival of all the packets at the input of the switch is synchronized.

In all the described switch architectures below, initially a small portion of optical power is tapped at each input and passed to the detector where it gets converted to an electrical signal which is forwarded to the electronic control (Figs. 3.1, 3.2, 3.3, 3.4, 3.5). The control reads the header bits to determine the required routing for the packet and drives the switch components accordingly.

3.1.1 Wavelength routed photonic packet switch [15, 16]

Figure 3.1 shows the schematic diagram of a wavelength routed photonic packet switch. This packet switch consists of packet encoding block, buffering block and a packet demultiplexing block. The packet encoding block is composed of N TWC's. Each TWC converts a packet to a new wavelength corresponding to its desired output, i.e. a packet addressed to the i^{th} output is assigned a wavelength λ_i. The buffering block consists of an $N \times M$ SOA gate switch matrix followed by a set of M optical delay lines whose length range from *0* to *M-1* packet duration. The SOA-gate switch matrix provides wavelength-encoded

packets access to appropriate delay-lines so that each packet can receive any amount of delay ranging from *0* to *M-1* packet duration. The demultiplexing block consists of an $M \times N$ star coupler followed by a set of N bandpass filters, one at each output to select packets whose wavelengths are matched with its passband.

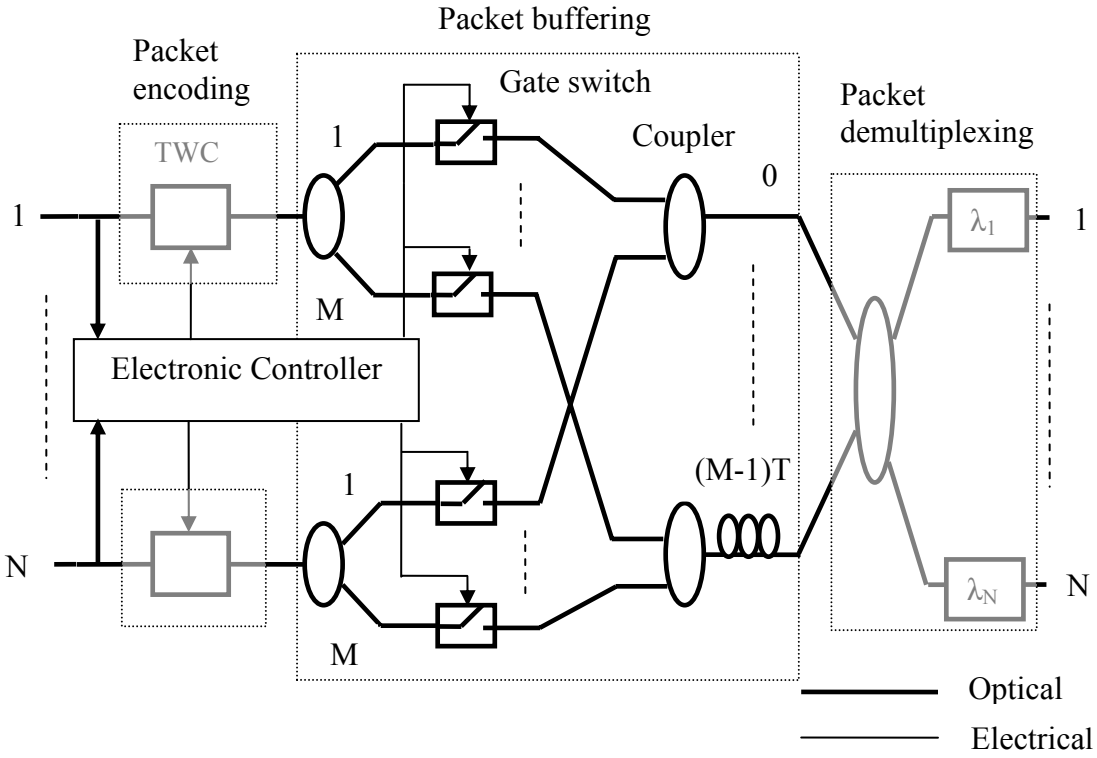

Figure 3.1: Wavelength routed photonic packet switch

As packets are wavelength encoded from λ_1 to λ_N, the M delay lines function as N first-in first-out (FIFO) buffers, each corresponding to one output and capable of storing up to M packets. In these FIFO buffers, packets arriving on the same input destined for the same output are received at the output in the same order in which they arrived.

In each timeslot, zero or more packets can arrive at the input, destined for a particular output and hence a particular output buffer. All these packets are placed in the appropriate buffer simultaneously. However, if the buffer is full, the packet is dropped and packet loss occurs. Thus, this switch emulates an output buffered switch in which M packets can be buffered for every output and the following scheduling algorithm can be considered (Table 3.1).

```
for output i=1 to N  do
      counter[i]=0;
end
while() do
     for input i=1 to N do
          if (packet is present on input i for output j)
              if (counter[j]<M)
                  convert the packet to wavelength λj and route it to buffer
                  with delay counter[j]-1;
                  counter[j]=counter[j]+1;
             else
                  drop the packet;
             end
          end
     end
     for output j=1 to N do
          if (counter[j]!=0)
              packet at wavelength λj is picked by output j;
              counter[j]=counter[j]-1;
          end
     end
end
```

Table 3.1: Scheduling algorithm for wavelength routed photonic packet switch

3.1.2 Broadcast and select packet switch [17]

Figure 3.2 shows the schematic diagram of a broadcast and select packet switch. In this switch, packets from different inputs are encoded on a different wavelength. These packets are then multiplexed and these combined WDM packets are broadcasted to a set of *M* optical delay lines. This set of delay lines is shared by all the packets. After propagating through the set of *M* delay lines, each packet can receive any amount of delay ranging from *0* to *M-1* packet duration.

Packets emerging at each delay line are broadcasted to all the output ports. At each output, two sets of optical gate switches are used for packet selection. The first set of gate switches selects a particular delay line through which a desired packet arrives; the second set of gate switches in conjunction with a WDM Demux-Mux pair chooses a wavelength at which the desired packet is encoded. Similar to the wavelength routed switch, this switch also emulates an output buffered switch.

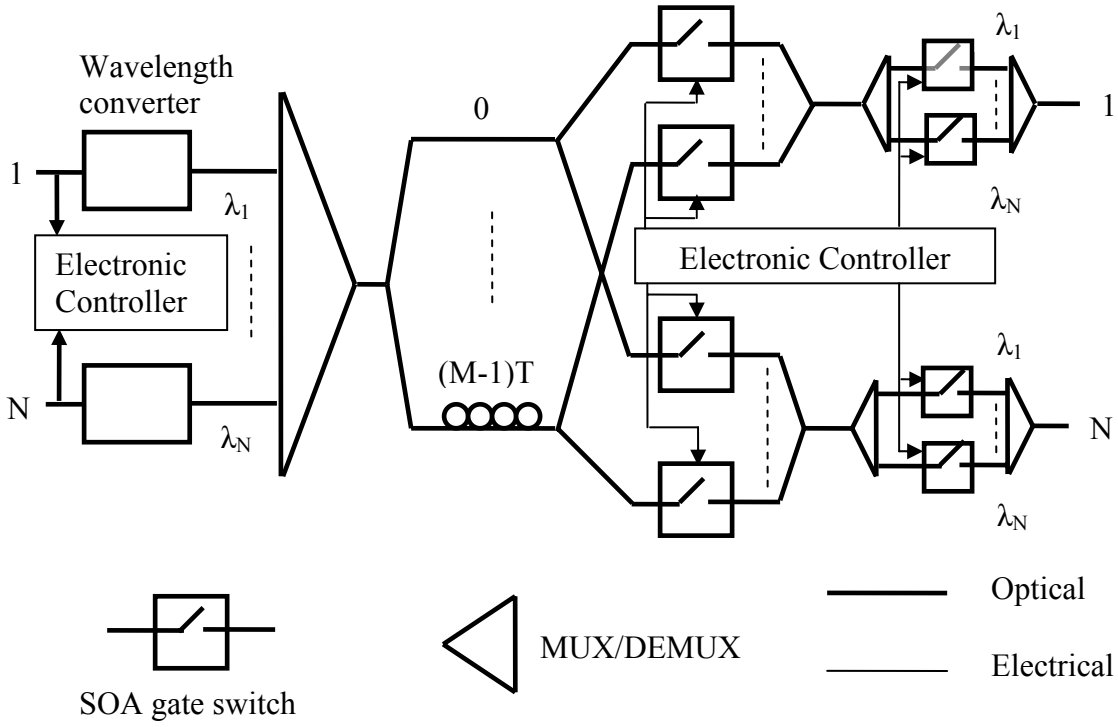

Figure 3.2: Broadcast and select packet switch

3.1.3 Input buffered wavelength routed switch [18]

Figure 3.3 shows the schematic diagram of a input buffered wavelength routed switch. This switch consists of a scheduling section and a switching section with an electronic controller to control the operation of the switch. The scheduling section has *N* TWC's, one at every input of an *N*-input *N*-output wavelength routing based buffer. This buffer has a pair of $K \times K$ AWGM's connected by a set of *M* optical fiber delay-lines where $K=\max(N, M)$. In this buffer, multiple packet streams can share a single pool of fiber delay lines through WDM by means of a pair of AWGM's. The relationship between the inputs,

outputs and the wavelengths within a 4 × 4 AWGM is shown in Table 3.2. It can be seen that packets entering the buffer at the i^{th} input port will be routed to different outputs of the first AWGM, according to their wavelengths. After passing through different lengths of delay lines, these packets are multiplexed by the second AWGM, and then leave the buffer from the i^{th} output port.

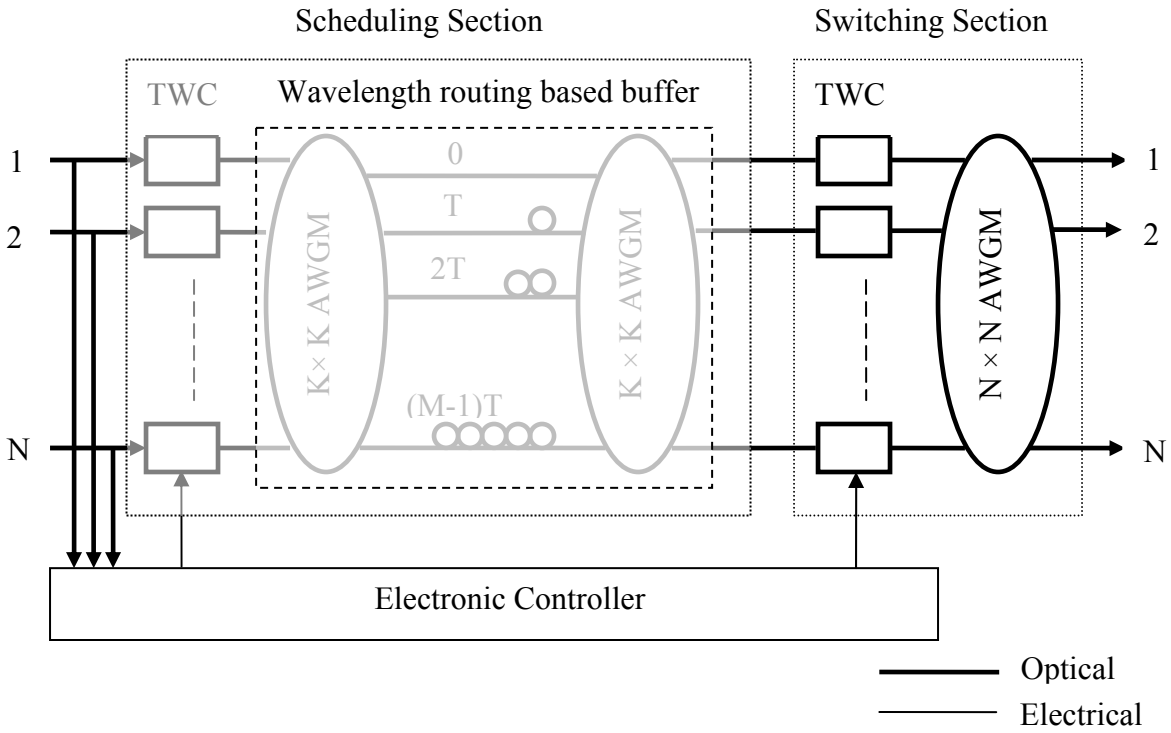

Figure 3.3: Input buffered wavelength routed switch

Output

Input	1	2	3	4
1	λ_1	λ_2	λ_3	λ_4
2	λ_2	λ_3	λ_4	λ_1
3	λ_3	λ_4	λ_1	λ_2
4	λ_4	λ_1	λ_2	λ_3

Table 3.2: Wavelength routing model of a 4 × 4 AWGM

The switching section is composed of N TWCs and an $N \times N$ AWGM. The TWC converts the packet to appropriate wavelength so that AWGM can route them to their desired output.

The *M* delay lines can provide a delay of *0* to *(M-1)*-packet times. In other words, there are a maximum of *M* possible time slots that can be assigned to a new packet for each input. Thus, each input logically has its own buffer, though the delay lines are physically shared by all the inputs. The scheduling algorithm should allocate a minimum time delay to each packet subject to the following two conditions in any time slot

- Only one packet at the most will be routed to same output of the switching section simultaneously.
- Only one packet at the most will appear at each input of the switching section.

Thus, the following scheduling algorithm can be considered for IBWR switch for each time slot (Table 3.3).

```
for input i= 1 to N do
    if (packet is present on input i) then
      packet_placed=0;
      for timeslot j = 0 to M-1 do
          if (timeslot j is idle and no other packet to the same destination) then
            convert the packet to wavelength depending on input and delay line output j;
            allocate timeslot j to the packet;
            packet_placed=1;
           break;
          end
      end
      if(packet_placed=0) drop the packet;
   end
end
for input i=1 to N of the switching section
   if (packet is present on input i) then route the packet to desired output by
      converting it to appropriate wavelength;
end
```

Table 3.3: Scheduling algorithm for input buffered wavelength routed switch

3.1.4 Output buffered wavelength routed switch [18]

Figure 3.4 shows the schematic diagram of an output buffered wavelength routed switch. In this switch architecture, switching of packets to its desired outputs is carried out before packet buffering. It consists of a set of N TWC's, an $N \times N$ nonblocking optical space switch, and an $N \times N$ wavelength routing based buffer (Fig. 3.4). The $N \times N$ optical space switch performs the switching of packets to their desired outputs.

In each time slot, the packets that are destined for the same output will be shifted to different wavelengths and then routed to the desired outputs by the optical space switch. Since these packets have been given different wavelengths by the TWC's at the input of the space switch, they will receive different packet delays at the wavelength routing based buffer so that only one packet will emerge at a given output of the switch in any time slot and hence packet contention will be resolved. This switch also effectively operates as an output buffered switch.

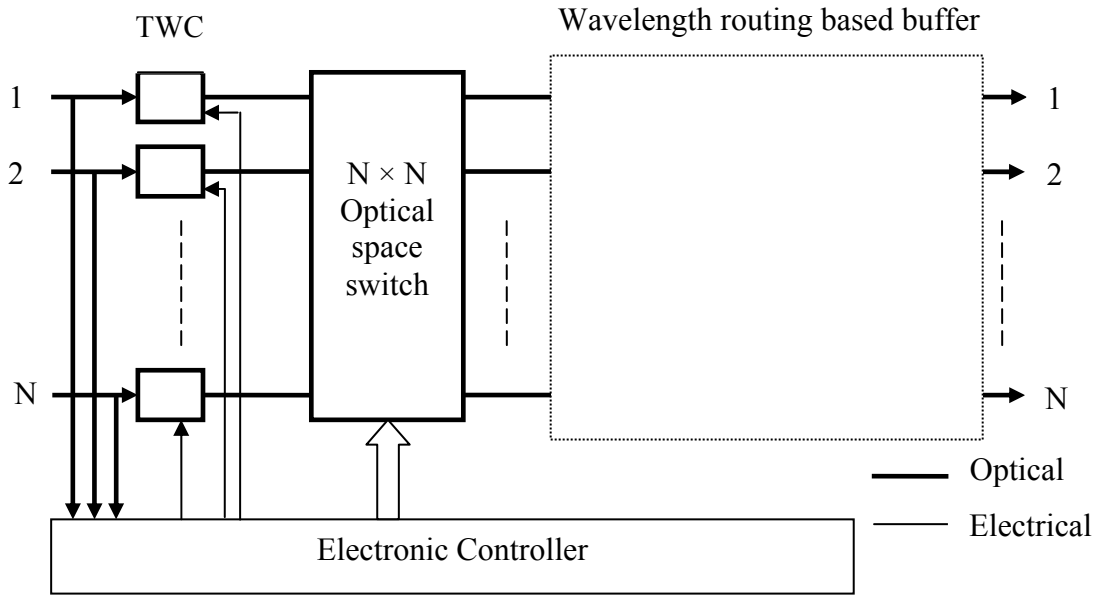

Figure 3.4: Output buffered wavelength routed switch

3.1.5 Staggering Switch [19]

This architecture is based on two stages: the scheduling stage and the switching stage (Fig. 3.5). Each stage uses a non-blocking space switch. The scheduling stage is $N \times M$ and the switching stage is $M \times N$ where $M \geq N$. The scheduling stage is connected to the

switching stage by M delay lines, d_i, i = 0 to M-1. The delay of the d_i delay line equals i packets.

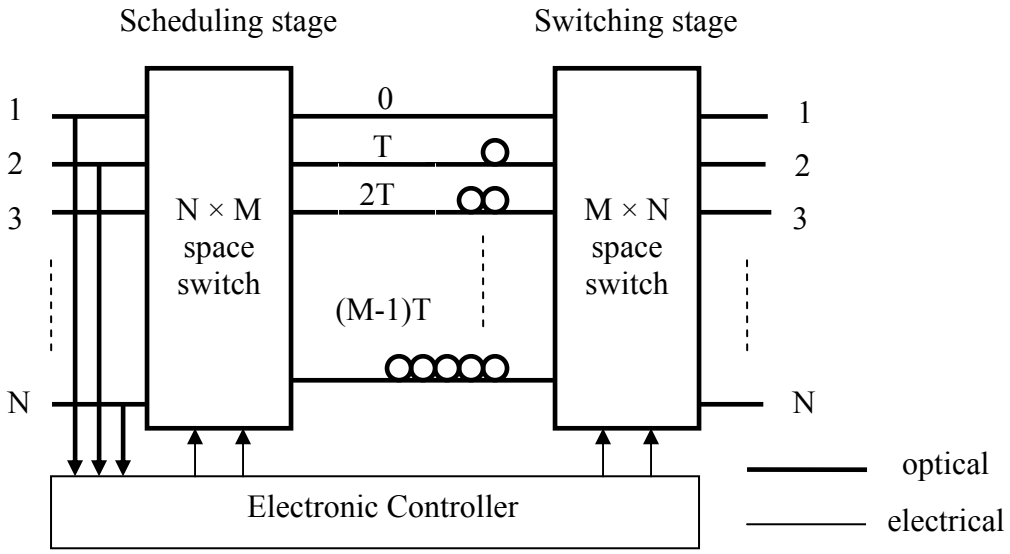

Figure 3.5: Staggering switch

The control receives the header information from all the arriving packets and attempts to allocate as many packets as possible into the lowest possible delay lines, d_i subject to the following conditions in any time slot

- No previous packet was inserted in the delay line in this time slot
- No other packet to the same destination exists in the column in which the packet is to be inserted (Fig. 3.6).

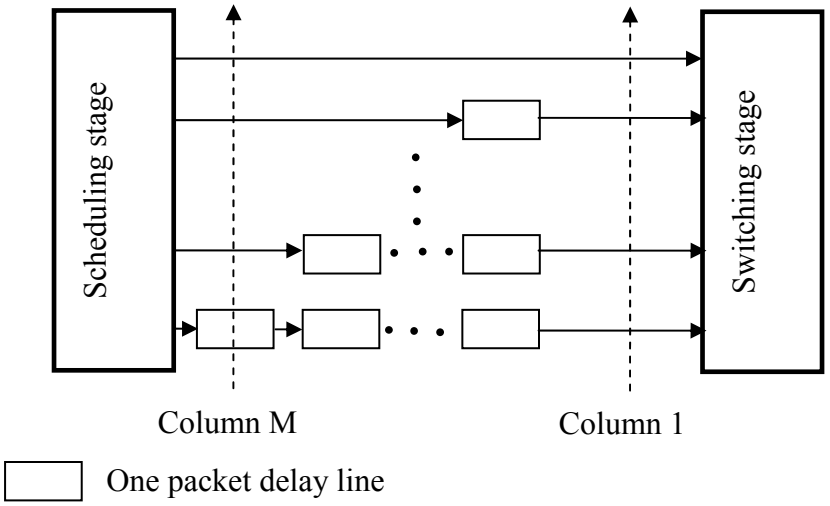

Figure 3.6: Scheduling in Staggering switch

Using Fig. 3.6, the scheduling is done according to the following algorithm called sequential scheduling algorithm (Table 3.4).

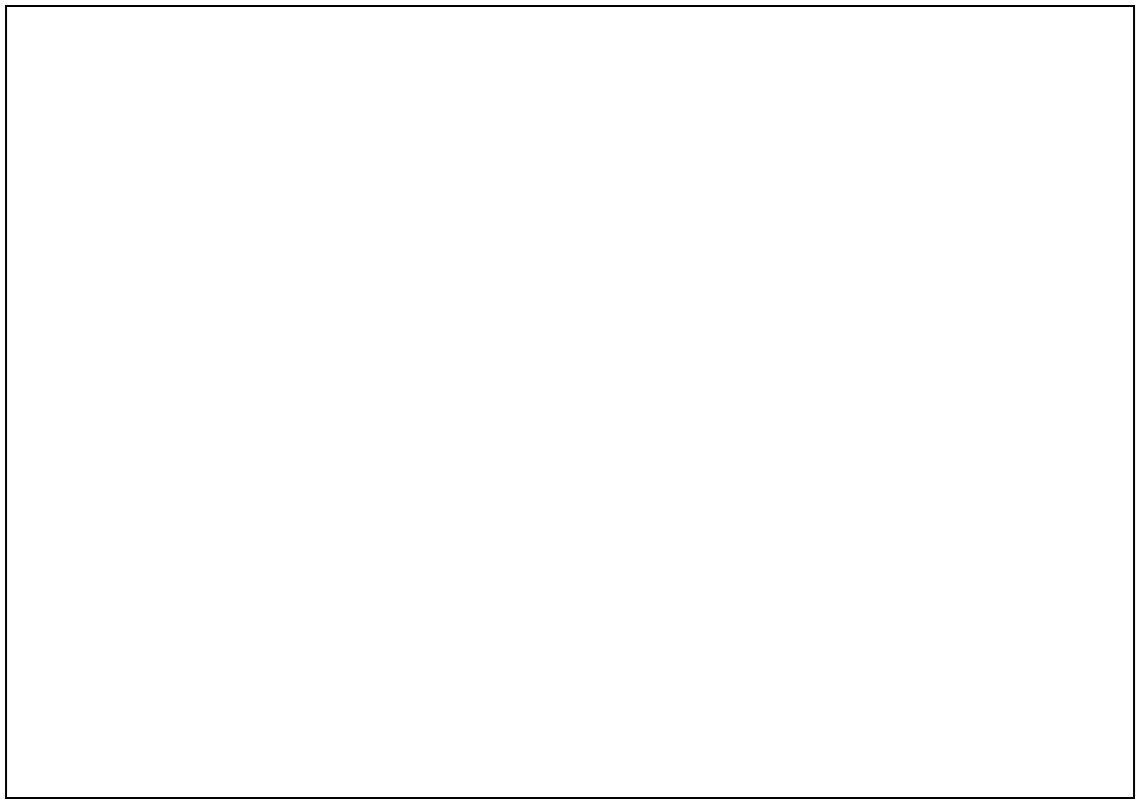

Table 3.4: Scheduling algorithm for staggering switch

3.2 Comparison of photonic packet switches

In this thesis, the switch architectures described above are compared. These switches are wavelength routed packet switch, broadcast-and-select (B&S) packet switch, input buffered wavelength routed (IBWR) switch, output buffered wavelength routed (OBWR) switch, and the staggering switch. The comparison is done on the basis of following metrics

3.2.1 Number of Components

The component count of different architectures is given in the following Table 3.5. Here the switch size is N and the number of delay lines is M. It is assumed that optical space switches are made from SOA gates using crossbar architecture [11].

	Wavelength routed packet switch	B&S packet switch	IBWR Switch	OBWR Switch	Staggering switch
Tunable WC	N	0	$2N$	N	0
Fixed WC	0	N	0	0	0
Optical gate switches	NM	$N(N+M)$	0	N^2	$2NM$
AWGM	0	0	1 ($N \times N$) 2 ($K \times K$)	1 ($N \times N$)	0
BPF	N	0	0	0	0
MUX/ DEMUX	0	$2N+1$	0	0	0
Coupler	N (1 × M) M (N × 1) 1 (M × N)	M (1 × N) N (M × 1) 1 (1 × M)	0	N (1 × N) N (N × 1)	$2N$ (1 × M) $2M$ (N × 1)

Table 3.5: Component count of different architectures

From Table 3.5, it can be seen that the number of SOA gate switch in all switch architectures other than IBWR switch increases rapidly with an increase in number of delay lines and/or the switch size. IBWR switch uses AWGM along with TWC to switch the signals to their desired output ports. But, IBWR switch requires double the number of TWC than OBWR switch and wavelength routed packet switch.

3.2.2 Packet loss probability

In the above switch architectures, some of the packets cannot be accommodated by the scheduling algorithm and will be lost. This fraction of lost packets is characterized by probability of packet loss. It is given by

$$\Pr(Packet\,lost) = \frac{Number\,of\ packet\,lost}{Number\,of\ packet\,transmitted}\ . \qquad (3.1)$$

To simulate the packet loss probability of different switches, random traffic model for the arrival of packets to the input can be used which makes the following assumptions

- Each input has an identical and independent arrival of packets with a probability ρ in any time slot.
- Each input packet is equally likely destined for any of the outputs.

The probability ρ can also be called as offered traffic load at each input. This offered traffic load is the probability that an input slot contains a packet. In any time slot, the probability of i new packets addressed to a given output of the switch can be given by

$$\Pr(i) = \binom{N}{i}\left(\frac{\rho}{N}\right)^{i}\left(1-\frac{\rho}{N}\right)^{N-i}, \qquad (3.2)$$

where N is the switch size or the number of input / output ports.

The simulation results for packet loss probability of different switches are discussed in Chapter 5.

3.2.3 Latency

The switch latency is defined as the average time in slots that a packet is delayed in the switch. The maximum and minimum switch latencies are *0* and *M-1* respectively. Latency is averaged only for those successful packets which are not dropped. The latency of different switches is discussed in Chapter 5.

3.2.4 Packet ordering and packet priority

In staggering switch, packets arriving on the same input destined for the same output may be received at the output in the reverse order, because the packet arriving first may be placed in a longer delay line than the packet arriving later. In other switches, inputs or outputs have logically their own buffer and hence packet order is preserved. Thus, except the staggering switch, the packet ordering in all the switches is preserved.

Implementation of packet priority means whether a low priority packet can be preempted by a packet with higher priority. This feature is possible only in broadcast and select packet switch because in this switch, each packet is broadcasted to all the delay lines. So, a packet can be received with all possible delays. Thus, if a high priority packet arrives at the switch, a low priority packet can be preempted and received at a higher delay. This is

not possible in other switch architectures because a packet already in the buffer cannot be preempted by a high priority packet.

3.3 Performance of photonic packet switch architectures under bursty traffic conditions: Simulation study

Except for the staggering switch, packet loss probability has been simulated using random traffic model in all switch architectures. In this model, traffic on each input is assumed to be uniformly distributed and uncorrelated in time. However, the assumption of uncorrelated destinations is not realistic in high speed services such as host to host data and video communication, where the source data rate may be comparable to the link rate. In this case strong correlation can exist between adjacent time slots since packets from these high speed sources may be transmitted consecutively on the link. Packets from such high speed sources may dominate the traffic on the links.

Thus, as a function of time, traffic on each input will be composed of bursts of packets destined to the same output (Fig. 3.7). These bursts are followed by an idle period that can be of length zero i.e. two back to back bursts.

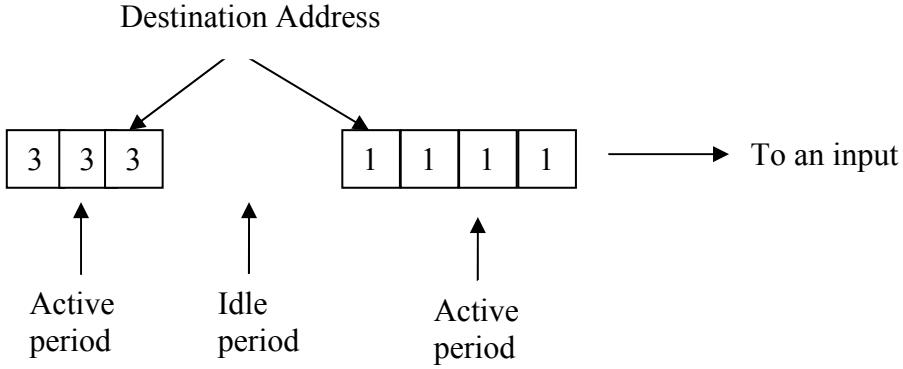

Figure 3.7: Packet arrival to an input under bursty traffic

This bursty traffic can be characterized by two parameters

- Average offered traffic load per input, ρ
- Average burst length, l

Time correlation of traffic on each input can be modeled as the markov chain [19] shown in Figure 3.8. The chain is composed of two states: idle state (0) or burst state (b).

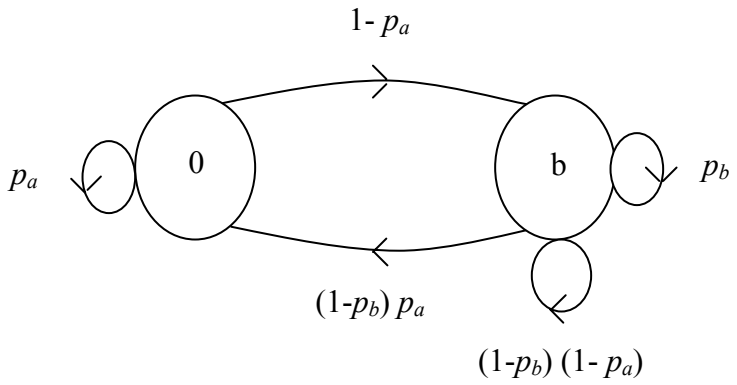

Figure 3.8: Markov model for bursty traffic

The system is in the idle state (0) when no packet arrives in the current slot. With probability p_a, no packet will also arrive in the next slot, while with probability $1\text{-}p_a$, a new burst will begin and the system will transfer to the burst state (*b*). In this state, one of the values of output destinations is chosen with equal probability. Being in this bursty state, the next arrival will be part of the burst with probability p_b (i.e., destined to the same destination), or the burst will terminate with probability p_a. The termination of the burst can occur in two ways: by starting a new burst (i.e. to another destination), in which case the system will transfer to a different output destination than the current one with equal probability, or by going to the idle state. The probability of the first case is $(1\text{-}p_b).(1\text{-}p_a)$, and the probability of the second case is $(1\text{-}p_b).p_a$. Equation (3.3) corresponds to the steady state solution of the above markov chain

$$\begin{pmatrix} p_a & p_a.(1-p_b) \\ (1-p_a) & (1-p_a).(1-p_b)+p_b \end{pmatrix} \begin{pmatrix} \pi_0 \\ \pi_b \end{pmatrix} = \begin{pmatrix} \pi_0 \\ \pi_b \end{pmatrix}, \tag{3.3}$$

where π_0 is the steady state probability of the system being in idle state (0) and π_b is the steady state probability of system being in burst state (*b*). Here, π_b is the sum of steady state probability of system being in any of the output destination burst, $j = 1$ to N.

$$\pi_b = \sum_{j=1}^{N} \pi_j \,, \tag{3.4}$$

where π_j is the steady state probability of system being in a particular output destination burst. Also,

$$\pi_0 + \pi_b = 1 \,. \tag{3.5}$$

Solving (3.3) and (3.5) results in

$$\pi_0 = \frac{p_a.(1-p_b)}{1-p_a.p_b}, \tag{3.6}$$

$$\pi_b = \frac{1-p_a}{1-p_a.p_b}\ . \tag{3.7}$$

Now, we calculate the average offered traffic load ρ, which equals the fraction of the time that the system is not in the idle state

$$\rho = 1-\pi_0 = \frac{1-p_a}{1-p_{a.}p_b}, \tag{3.8}$$

The probability of burst length, k is

$$\Pr(k) = (1-p_b).p_b^{\,k-1}, k \geq 1\,. \tag{3.9}$$

Then the average burst length is

$$l = \sum_{k=1}^{\infty} k.\Pr(k) = \frac{1}{1-p_b}\,. \tag{3.10}$$

For a given offered traffic load, ρ and average burst length, l, p_a and p_b can be calculated from (3.8) and (3.10). These values of p_a and p_b can be used to generate the input traffic for the switch using markov bursty traffic model (Fig. 3.8) and the performance of different switch architectures under bursty traffic conditions can be simulated. The results are shown in Chapter 5.

Chapter 4

Maximum matching characterization of staggering switch

The scheduling algorithm may have a considerable effect on the performance of the switch, both in the probability of loss and latency. In staggering switch, the scheduling algorithm is referred to as the sequential algorithm. In this algorithm, the input ports are scanned sequentially and for each input packet present, the algorithm tries to insert the packet in the lowest possible delay line, subject to the following two conditions

- No previous packet was inserted in the delay line in this time slot.
- No other packet to the same destination exists in the column in which the packet is to be inserted (Fig. 3.6).

The sequential algorithm is not an optimum algorithm for achieving the lowest packet loss probability. The problem of maximizing the number of packets that can be accommodated in a given slot can be viewed as the maximum matching on the bipartite graph as follows (Fig. 4.1)

- Nodes in the left side represent input ports with packets destined to the target output port.
- Nodes in the right side represent the delay lines without output contention.
- An edge between input *i* and delay line *j* indicates that a packet from input *i* can be accommodated in delay line *j*.

A matching for this bipartite graph is a valid schedule for the packets destined to the target output port, where each input packet is assigned at most one delay line, and each delay line has at most one input packet associated. This is our criteria of interest since in each time slot, non selected input packets are directly lost. This can be illustrated by the following example (Fig. 4.1). In a given time slot, input ports 1, 2, 3 have packets and they can be accommodated in the delay lines as shown in Figure 4.1.

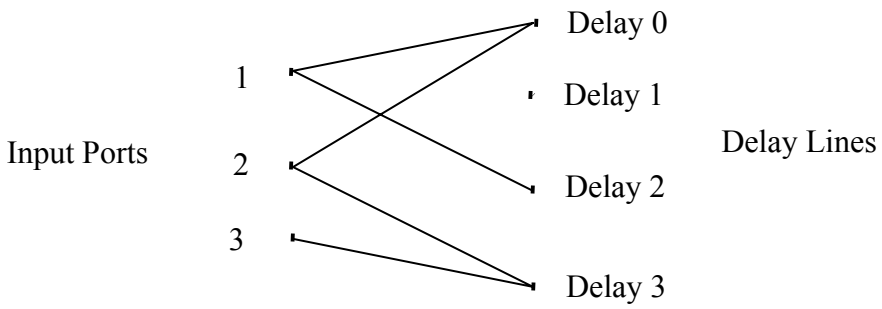

Figure 4.1: Problem Graph

Thus, in this time-slot the sequential algorithm will scan the input ports sequentially and will assign packets at input ports 1 and 2 to delay lines 0 and 3 respectively. The packet at input port 3 will have to be dropped (Fig. 4.2).

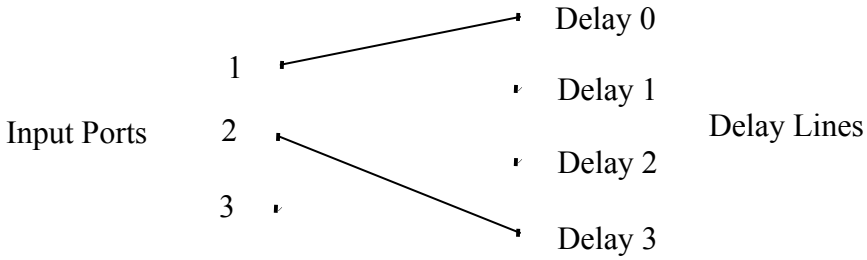

Figure 4.2: Sequential Match

Maximizing the size of matching on this bipartite graph (Fig. 4.1) corresponds to the maximum number of packets that can be accommodated in this slot. Thus, the maximum size matching will provide a better solution as all three input packets can be accommodated in the delay lines (Fig 4.3).

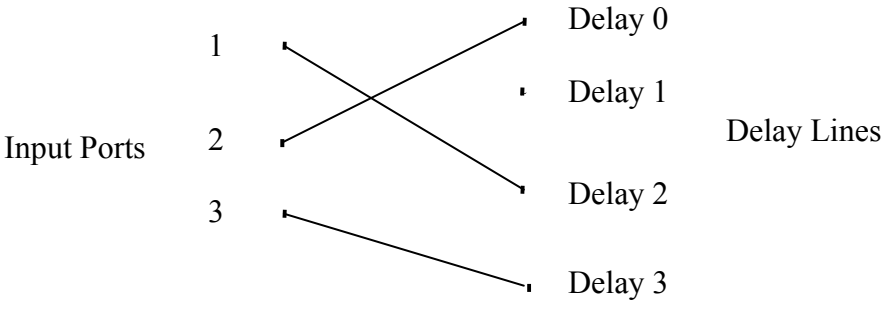

Figure 4.3: Maximum Match

However, if more than one option exists for maximum sized matching, then a matching with overall lower numbered delay lines should be chosen.

This can be accomplished by assigning weights to the edges in the bipartite graph. In this edge-weighted bipartite graph, each edge between left hand side node *i* and right side node *j* is assigned a weight x_{ij} proportional to delay (Fig. 4.4). Thus, if a maximum match with the least sum of weights of the matched edges is chosen then an optimum solution per time slot is obtained which maximizes the switch throughput and minimizes latency (Fig. 4.5 and Fig.4.6).

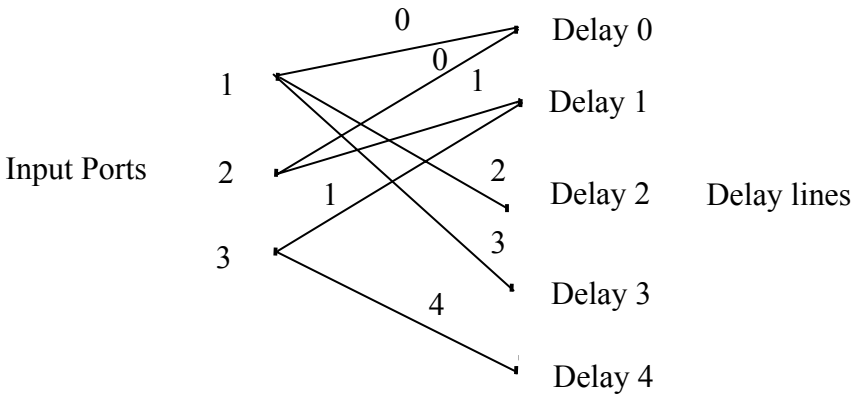

Figure 4.4: Problem Graph

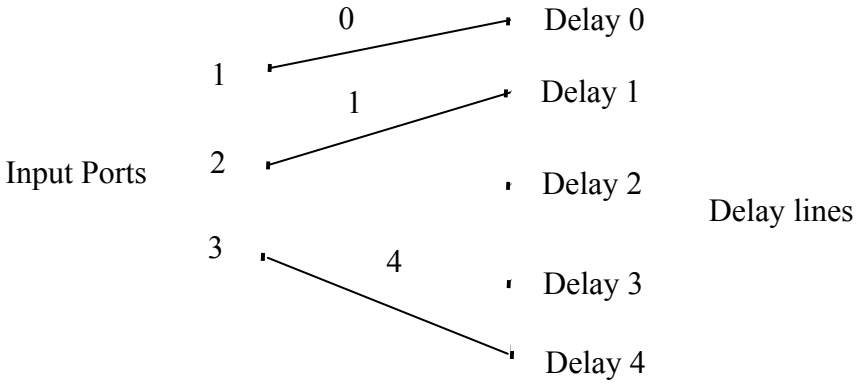

Figure 4.5: Sub Optimum Match

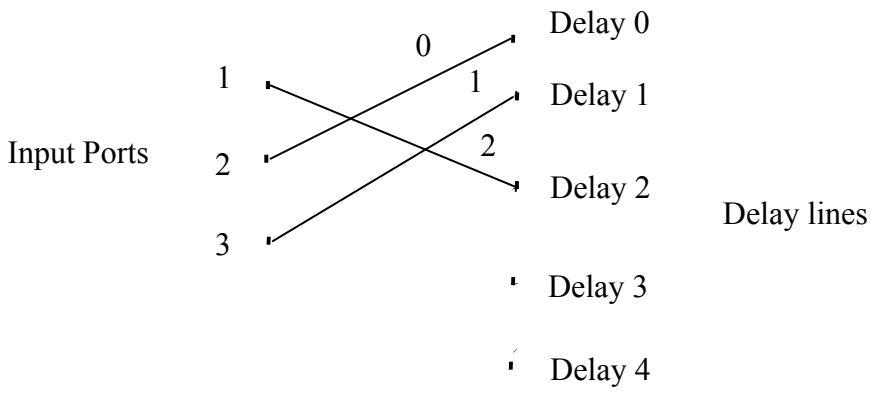

Figure 4.6: Optimum Match

As shown in the figures 4.5 and 4.6, the sum of the delays in the sub-optimum matching is (0+1+4) =5 whereas in optimum matching it is (0+1+2) =3.

To solve this assignment of packets to delay lines problem, Hungarian algorithm [20,21] can be used. In this algorithm, we will replace the edge-weighted bipartite graph with a matrix where the rows represent inputs and columns represent delay lines and the entry in row *i* and column *j* is equal to *j*. Those entry in the matrix for which there is no edge in the bipartite graph is replaced by a large number ∞. Also, the matrix is made square by adding extra rows or columns of zeros if the matrix is non-square. Let the matrix size be *n* x *n*. The algorithm is described as follows

The Hungarian Algorithm

Step 1 Subtract the smallest entry in each row from all the entries in that row. After Step 1, each row has at least one zero entry and all other entries are positive.

Step 2 Subtract the smallest entry in each column from all the entries in that column. After this step, each row and column have at least one zero entry.

Step 3 Draw lines through rows and columns so that all the zero entries of the matrix are covered and a minimum number of such lines have been used.

Step 4 Test for Optimality

(i) If the minimum number of covering lines is *n*, an optimal assignment of zeros is possible.

(ii) If the minimum number of covering lines is less than n, an optimal assignment of zeros is not possible. Go to step 5.

Step 5 Determine the smallest entry not covered by any line. Subtract this entry from all uncovered entries and then add it to all entries covered by both a horizontal and vertical line. Return to Step 3.

The following example shows the application of Hungarian algorithm to the problem graph of Figure 4.4. The matrix corresponding to this problem graph is given by

$$\begin{pmatrix} 0 & \infty & 2 & 3 & \infty \\ 0 & 1 & \infty & \infty & \infty \\ \infty & 1 & \infty & \infty & 4 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Now, applying Hungarian algorithm on this matrix, we have

Step 1 After applying this step we get the following matrix

$$\begin{pmatrix} 0 & \infty & 2 & 3 & \infty \\ 0 & 1 & \infty & \infty & \infty \\ \infty & 0 & \infty & \infty & 3 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Now, we have at least one zero in each row.

Step 2 As the smallest entry in each column is 0 we will get the same matrix after this step as step 1.

Step 3 Drawing minimum number of lines through rows and columns such that all zero entries are covered we get

$$\begin{pmatrix} 0 & \infty & 2 & 3 & \infty \\ 0 & 1 & \infty & \infty & \infty \\ \infty & 0 & \infty & \infty & 3 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Step 4 As the minimum number of covering lines needed is 4 which are less than 5, an optimal assignment of zeros is not possible. Hence, go to Step 5.

Step 5 The smallest entry not covered by any line is 2. Thus, subtracting 2 from all uncovered entries and adding it to all entries covered by both horizontal and vertical lines we get the following matrix

$$\begin{pmatrix} 0 & \infty & 0 & 1 & \infty \\ 0 & 1 & \infty & \infty & \infty \\ \infty & 0 & \infty & \infty & 1 \\ 2 & 2 & 0 & 0 & 0 \\ 2 & 2 & 0 & 0 & 0 \end{pmatrix}$$

Return to step 3.

Step 3

$$\begin{pmatrix} \cancel{0} & \cancel{\infty} & \cancel{0} & \cancel{1} & \cancel{\infty} \\ \cancel{0} & \cancel{1} & \cancel{\infty} & \cancel{\infty} & \cancel{\infty} \\ \cancel{\infty} & \cancel{0} & \cancel{\infty} & \cancel{\infty} & \cancel{1} \\ \cancel{2} & \cancel{2} & \cancel{0} & \cancel{0} & \cancel{0} \\ \cancel{2} & \cancel{2} & \cancel{0} & \cancel{0} & \cancel{0} \end{pmatrix}$$

Step 4 As the minimum number of covering lines is equal to 5, an optimal assignment of zeros is possible.

The optimal assignment can be seen in the matrix as

$$\begin{pmatrix} 0 & \infty & \bigcirc & 1 & \infty \\ \bigcirc & 1 & \infty & \infty & \infty \\ \infty & \bigcirc & \infty & \infty & 1 \\ 2 & 2 & 0 & \bigcirc & 0 \\ 2 & 2 & 0 & 0 & \bigcirc \end{pmatrix}$$

Thus, using this optimal assignment we can assign packets at input ports 1, 2, 3 to delay lines 0, 1, 2 respectively (Fig. 4.6). The result of applying this maximum matching scheduling algorithm is shown in Chapter 5.

Chapter 5

Simulation and Results

5.1 Comparison of photonic packet switches

Figure 5.1 and 5.2 plots the packet loss probability and latency respectively for different switch architectures with switch size, N = 16 and the number of delay lines, M = 16. Amongst the different switch architectures, broadcast and select packet switch, wavelength routed packet switch and output buffered wavelength routed switch operate as output buffered switch. Thus, they will have the same packet loss probability and latency performance.

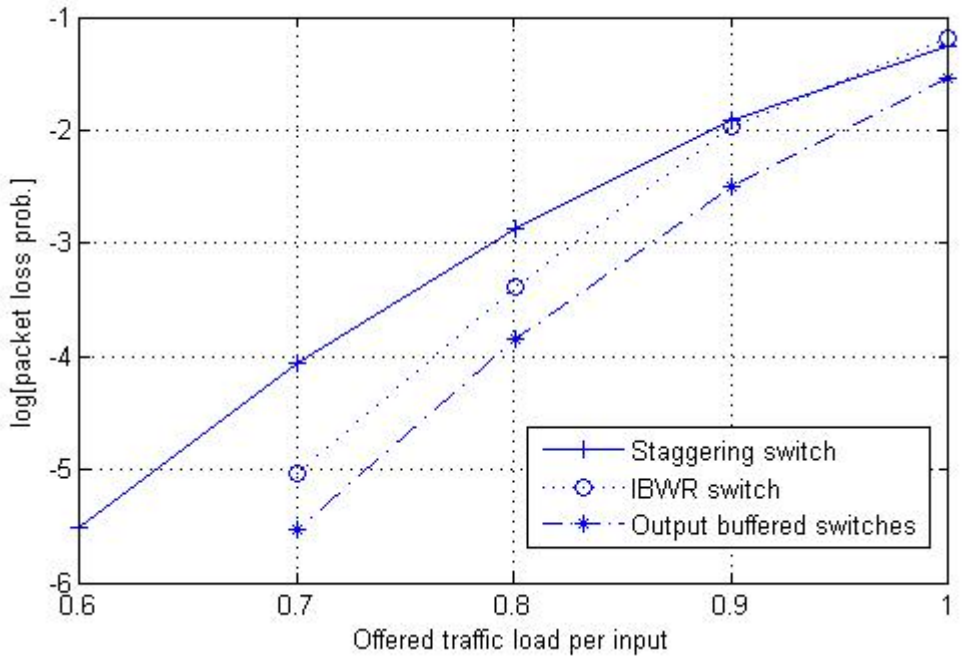

Figure 5.1: Packet loss probability vs. offered traffic load per input for different switch architectures of size, N=16 and number of delay lines, M=16

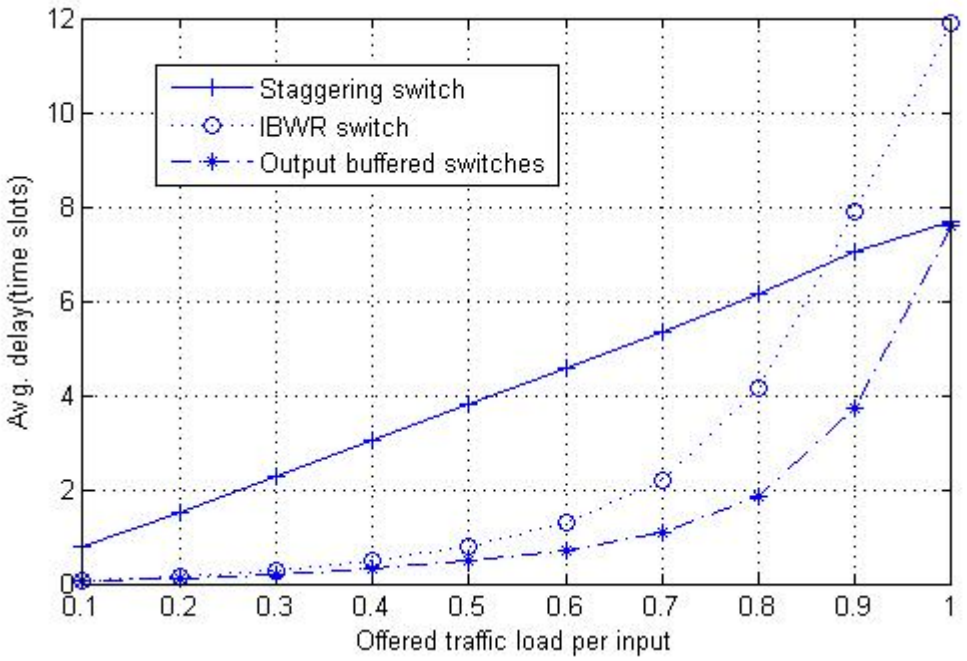

Figure 5.2: Latency vs. offered traffic load per input for different switch architectures of size N=16 and number of delay lines, M=16

It can be seen from Figure 5.1 and 5.2 that

- The packet loss probability and latency is lowest for output buffered switches at different offered load.
- The input buffered wavelength routed (IBWR) switch has a lower packet loss probability and latency than staggering switch for offered load less than 0.9.

For example, the packet loss probability at offered load 0.8 is 1.35×10^{-3} for staggering switch, 4.17×10^{-4} for IBWR switch and 1.45×10^{-4} for output buffered switches. Similarly, the latency at load 0.8 is 6.17 for staggering switch, 4.13 for IBWR switch and 1.87 for output buffered switches.

The better performance of output buffered switches and IBWR switch than staggering switch is because of increased efficiency of fiber delay lines due to wavelength division multiplexing. Output buffered switches perform better than IBWR switch because arriving packets destined for one output do not cause contention to packets going to different outputs unlike in IBWR switch.

Figure 5.3 plots the packet loss probability of different switch architectures with switch size, N=16 and offered traffic load per input, ρ=0.8 with increasing number of delay lines, M=16 to M=26.

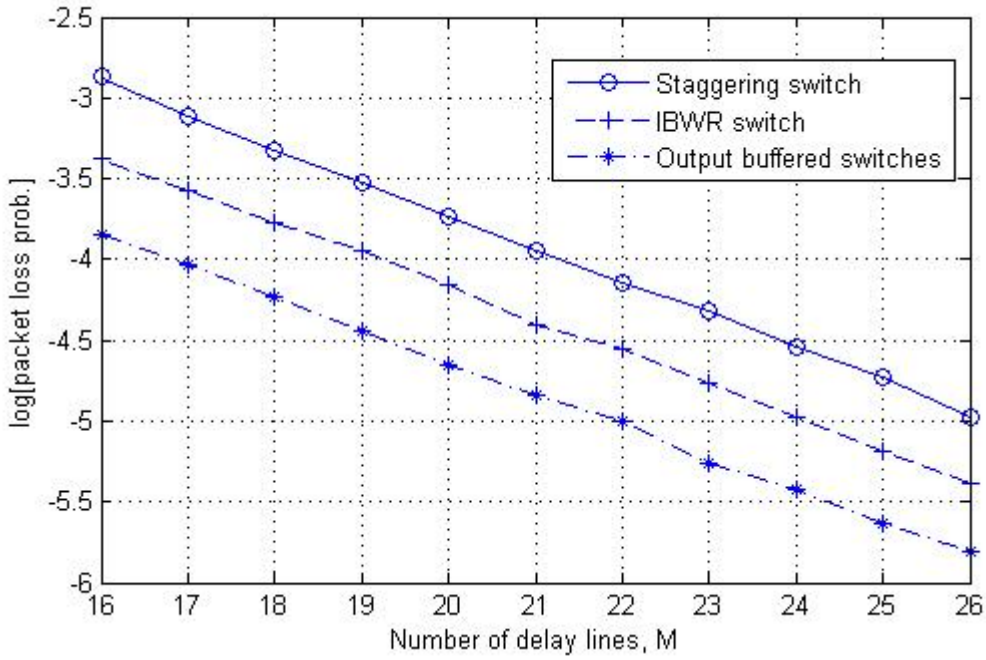

Figure 5.3: Packet loss probability vs. number of delay lines for different switch architectures of size N=16 and offered traffic load per input, ρ=0.8

As can be seen, for same packet loss probability, output buffered switches require the least number of delay lines followed by IBWR switch and staggering switch. For example, to achieve a packet loss probability of 10^{-5} output buffered switches need M=22, IBWR switch needs M=24 and for staggering switch, M=26.

Using these values of M for different switches, a comparison can be made for the component counts of different switch architectures under same packet loss probability at a given offered traffic load per input and a given switch size (Table 3.5). It can be seen that the number of SOA gate switches will increase most rapidly in staggering switch due to two optical space switches used in the architecture and the largest value of number of delay lines, M required for a given packet loss probability. Amongst the output buffered switches, the number of SOA gate switches will increase most rapidly in broadcast and select packet switch.

Figures 5.4 and 5.5 plot the packet loss probability and latency respectively of different switch architectures for a given M=20 and offered traffic load ρ=0.8 with switch size varying from N=8 to N=20.

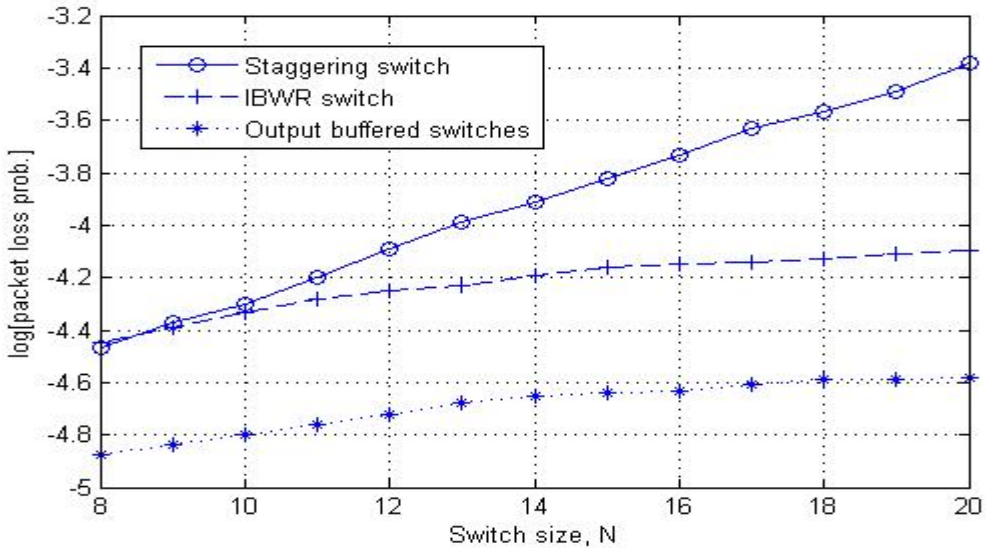

Figure 5.4: Packet loss probability vs. switch size for different switch architectures with number of delay lines, M=20 and offered traffic load per input, ρ=0.8

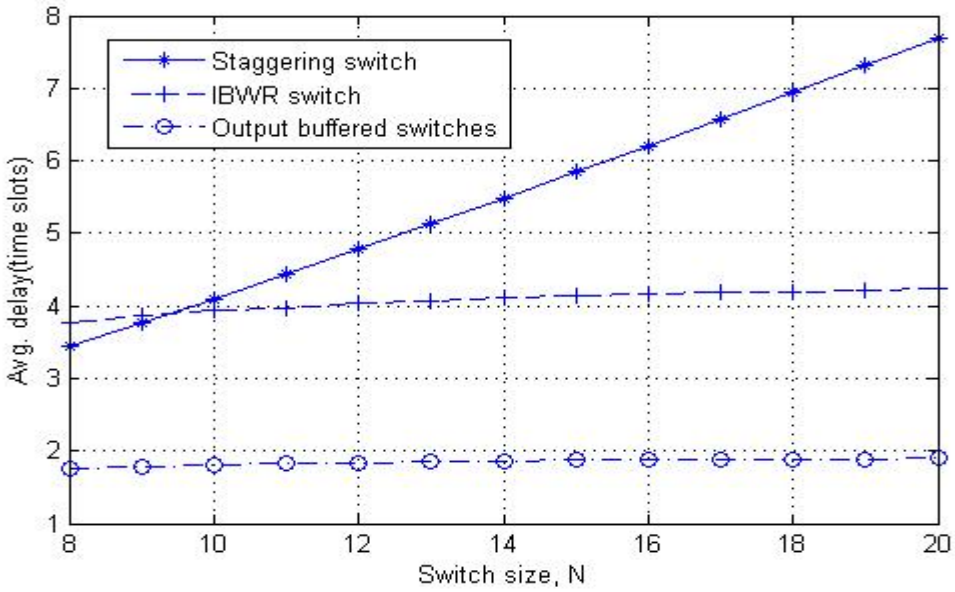

Figure 5.5: Latency vs. switch size for different switch architectures with number of delay lines, M=20 and offered traffic load per input, ρ=0.8

It can be seen from these figures that except for the staggering switch, packet loss probability and latency is not much affected by the increase in switch size for a given number of delay lines in other switch architectures. This happens because in IBWR switch and output buffered switches, each input or output has logically its own buffer whereas in staggering switch, the same number of delay lines are shared between the increased number of input ports.

It again implies that SOA gate switches will increase most rapidly in staggering switch with an increase in switch size because packet loss probability increases significantly with increasing switch size (Table 3.5). Thus, a higher number of delay lines will be required to maintain the same packet loss probability. But, in output buffered switches the packet loss probability is not much affected by the increase in switch size. So, almost the same number of delay lines will be required to achieve the same packet loss probability.

5.2 Bursty traffic performance of different switches

The following graphs (Fig. 5.6, 5.7, 5.8) show the packet loss probability performance of different switch architectures under bursty traffic conditions. The switch size is taken as N=16 and the number of delay lines, M=16. The average burst length is l and l takes values of 1, 2, 5 and 8 in the simulation. Here, l=1 corresponds to the random traffic with no time correlation.

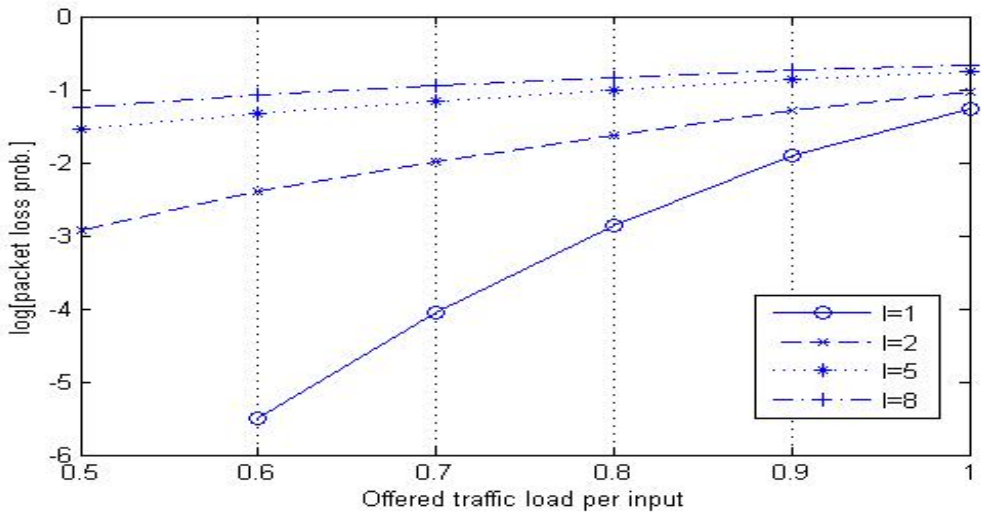

Figure 5.6: Packet loss probability vs. offered traffic load per input for staggering switch under bursty traffic with switch size, N=16 and number of delay lines, M=16

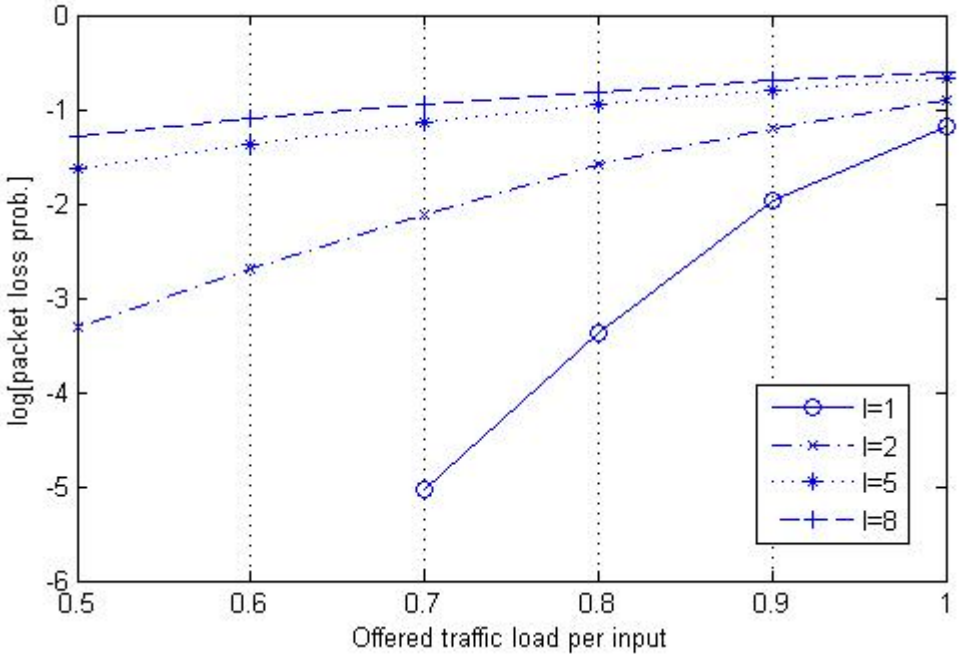

Figure 5.7: Packet loss probability vs. offered traffic load per input for IBWR switch under bursty traffic with switch size, N=16 and number of delay lines, M=16

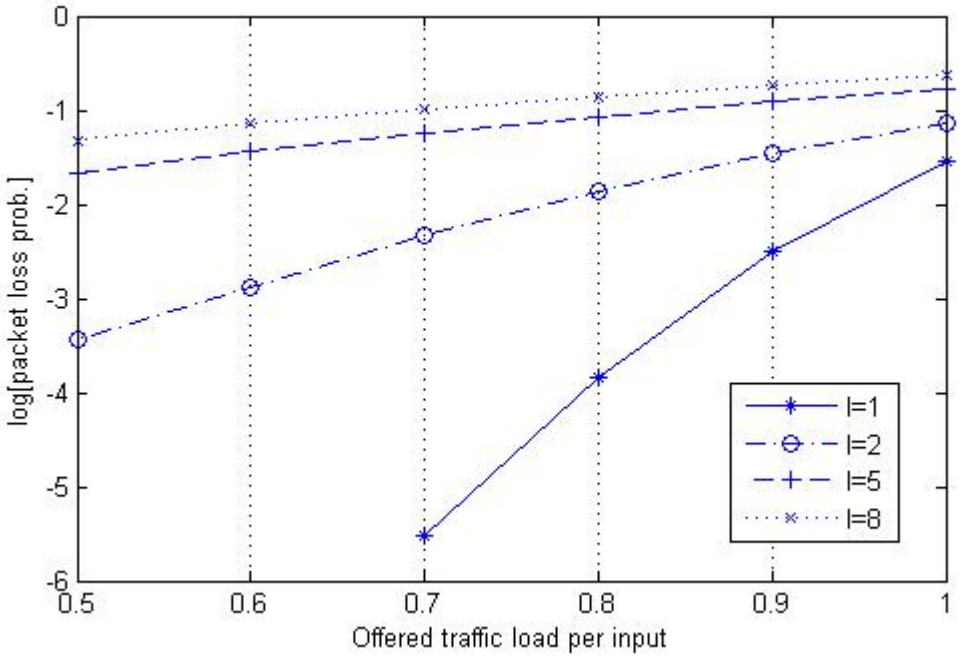

Figure 5.8: Packet loss probability vs. offered traffic load per input for output buffered switches under bursty traffic with switch size, N=16 and number of delay lines, M=16

It can be seen that

- Performance of all the switches degrades rapidly with an increase in the average burst length.
- The slope of the log[packet loss probability] vs. ρ curve becomes less steep with increasing l so that decreasing ρ does not lower loss probability as much as random traffic. This happens because when the average burst length is too long, the buffer capacity will be exceeded even at lower loads.

If we overlay the graphs of Figures 5.6, 5.7 and 5.8 we can see that the packet loss probability performance of different switch architectures under bursty traffic comes close to each other with an increasing average burst length. As an example, the following graph (Fig. 5.9) compares the different switch architectures performance under bursty traffic for an average burst length, l=8.

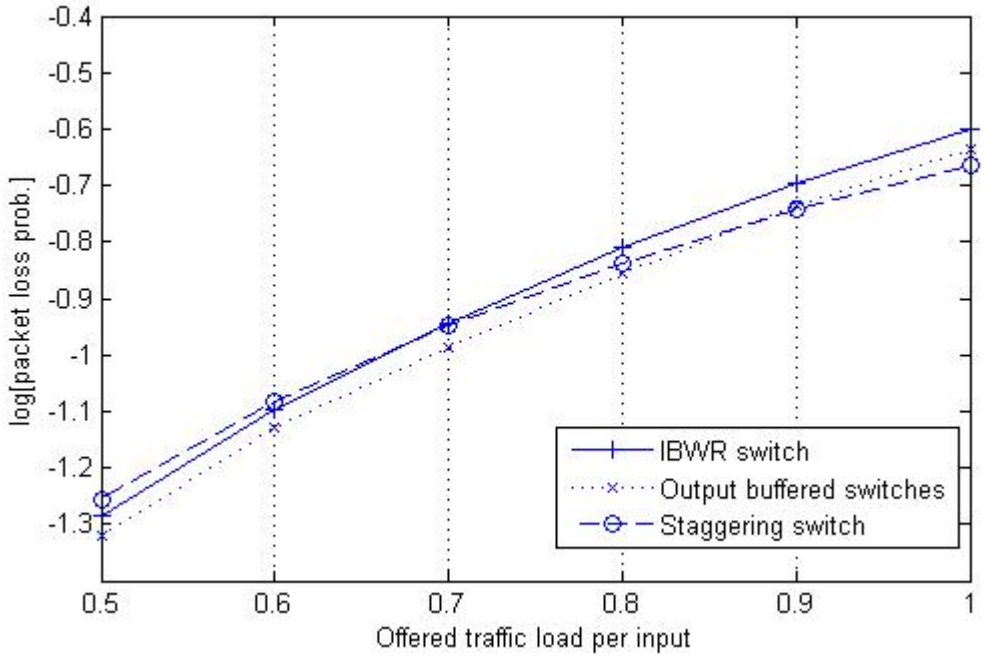

Figure 5.9: Packet loss probability vs. offered traffic load per input under bursty traffic for different switch architectures with switch size, N=16, number of delay lines, M=16 and average burst length, l=8

For example, the packet loss probability at load 0.8 is 1.47×10^{-1} for staggering switch, 1.54×10^{-1} for IBWR switch and 1.38×10^{-1} for output buffered switches.

Thus, under bursty traffic it can not be said that output buffered switches have a lower probability of packet loss as compared to IBWR switch and staggering switch. This happens because in output buffered switches, when multiple bursts arrive for the same output, packets from all these bursts go to the same output buffer and the buffer can overflow easily. This is also true for the shared buffers in staggering switch and input buffers in IBWR switch. Thus, the performance of all the switches is almost the same.

5.3 Maximum matching characterization of staggering switch

The following graph (Fig. 5.10) shows the application of the maximum matching algorithm on 8 × 8 and 16 × 16 staggering switch with 8 and 16 delay lines respectively. Random traffic on input ports is assumed. The performance is compared with the sequential algorithm. It can be seen that the performance of the switch improves in terms of packet loss probability, especially at lower offered loads. Also, latency of the switch which is averaged only for successful packets decreases slightly despite the increase in the number of successful packets. This is due to the placing of packets in the lower numbered delay lines.

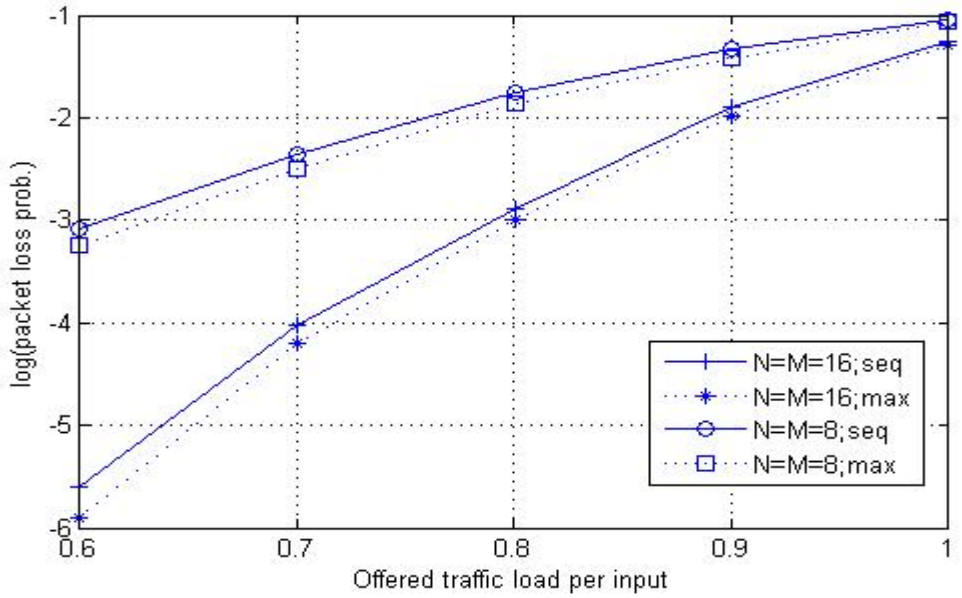

Figure 5.10: Packet loss probability vs. offered traffic load per input of staggering switch for sequential and maximum matching algorithm

Chapter 6

Conclusions and Future Work

6.1 Conclusions

This thesis presents the various photonic packet switch architectures and makes a comparison of them on the basis of number of components, packet loss probability and latency. The following conclusions are made

1. Output buffered switches have the lowest probability of packet loss and latency.
2. IBWR switch has a lower packet loss probability and latency when the offered traffic load is less than 0.9.
3. For a given offered traffic load, the number of delay lines required to achieve a given packet loss probability is least in output buffered switches followed by IBWR switch and staggering switch.
4. For a given offered traffic load and a given number of delay lines, the packet loss probability is almost independent of the switch size in all switches except staggering switch.

The bursty traffic analysis of different switch architectures is also presented. It can be seen that

1. As the average burst length is increased the performance of the switch degrades rapidly.
2. The slope of log[loss probability] vs. offered traffic load becomes less steep as the average burst length increases so that lower offered traffic load does not decreases loss probability as much as non bursty traffic.

On comparison of different switch architectures performance under bursty traffic the above results of random traffic do not hold. It is seen that the performance of all the switch architectures is almost the same.

Finally, it is seen that the performance of maximum matching algorithm is better in comparison to sequential algorithm for staggering switch in terms of packet loss probability and latency.

6.2 Future Work

In addition to the logical performance of the switch architectures in terms of packet loss probability and latency, a physical layer model can be developed for all the switch architectures to obtain their bit error rate (BER) and a comparison can be made.

Simulations can be done for other traffic models as well like Pareto traffic model which have been shown to model Internet traffic better.

Also, the maximum matching algorithm gives the optimum performance in terms of packet loss probability and latency in a time slot. Evaluation of the criteria for total throughput optimization, in all time slots together can be done.

REFERENCES

1. Rodney s. Tucker and Wen De Zhong, "Photonic Packet Switching: An overview," *IEICE Trans. Communication*, vol E-82-b, no.2, pp 254-264, Feb 1999.

2. L. Zucchelli, M. Burzio, and P. Gambini, "New solutions for optical packet delineation and synchronization in optical packet switched network," *Proc. ECOC'*96, pp.3.301-3.304, 1996.

3. Shun Yao *et al.*, "A Unified Study of Contention-Resolution Schemes in Optical Packet-Switched Networks," *IEEE/OSA J. Lightwave Technol*, vol. 21, no. 3, March 2003.

4. P. Gambini *et al.*, "Photonic packet switching based on fibre memories," *Proc. Photonics in Switching*, pp.2-3, 1996.

5. J. Spring, R.M. Fortenbarry, and R.S. Tucker, "Photonic Header replacement for Packet Switching," *Electron. Lett.*, vol. 29, pp. 1523-1524, Aug. 1993.

6. Rajiv Ramaswamy and Kumar N. Sivarajan. "Components," in *Optical Networks: A practical perspective*, 2nd ed., Harcourt Asia Pte. Ltd., 2002, pp. 83-172.

7. Paul E. Green Jr. "Fibers, Couplers and Taps," in *Fiber Optic Networks*, Prentice Hall, 1993.

8. Ian White, *et al.*, "Wavelength Switching Components for Future Photonic Networks," *IEEE Commun. Mag.*, Sep 2002.

9. H. Takasahi, et al., "Arrayed-waveguide grating for wavelength division multi/demultiplexer with nanometer resolution," *Electron. Lett.*, vol. 26,pp. 87-88,1990.

10. C. Dragone, "An NxN optical multiplexer using a planar arrangement of two star couplers,' *IEEE photon. Technol. Lett.*, vol. 3, no. 9, pp. 896-899, Sep 1991.

11. R.F. Kalman, L.G. Kazovsky, and J.W. Goodman, "Space division switches based on semiconductor optical amplifiers," *IEEE Photon. Technol. Lett.*, vol. 4, pp.1048-1051, Sept. 1992.

12. Georgios I. Papadimitriou et al., "Optical Switching: Switch fabrics, Techniques, and Architectures," *IEEE/OSA J. Lightwave Technol*, vol. 21, pp. 384-405, Feb. 2003.

13. D.J. Blumenthal, *et al.*, "Photonic packet switches: Architectures and experimental implementations," *Proc IEEE*, vol. 82, pp. 1650-1667, Nov 1994.

14. Shun Yao, Biswanath Mukherjee, and Sudhir Dixit, “Advances in Photonic Packet Switching: An Overview,” *IEEE Commun. Mag.*, Feb 2000.

15. J.M. Gabriagues *et al.*, “Design and implementation of a gigabit ATM photonic switching matrix,” *J. High speed Networks*, vol. 4, no. 4, Oct 1995.

16. J.M. Gabriagues and J.B.Jacob, “Photonic ATM switching matrix based on wavelength routing,” *Proc. Photonic Switching*, p. 2H2, 1992.

17. D. Chiaroni, *et al.*, “A novel photonic architecture for high capacity ATM switching applications,” *Proc. Photonics in Switching*, p.PThC3, 1995.

18. W.D.Jhong and R.S. Tucker, “A wavelength-routing based photonic packet buffer and its application in photonic packet switching systems,” *IEEE/OSA J. Lightwave Technol*, vol. 16, pp. 1737-1745, Oct. 1998.

19. Z Hass, “The staggering switch: An electronically controlled optical packet switch,” *IEEE/OSA J. Lightwave Technol.*, vol. 11, pp 925-936, May/June 1993

20. “Hungarian Algorithm”, http://www.cob.sjsu.edu/anaya_j/HungMeth.htm

21. “Hungarian Algorithm”, http://www.mcs.vuw.ac.nz/courses/MATH214/2006T1/matching_3.pdf