國立交通大學

資訊科學系

碩 士 論 文

輸入佇列光交換機中具服務品質保證之光暫存器設計

On the Design of Optical Buffer for Optical Input-Queued
Switches with Quality of Service Guarantees

研 究 生：林俊源

指導教授：陳　健　教授

中 華 民 國 九 十 四 年 六 月

輸入佇列光交換機中具服務品質保證之光暫存器設計

On the Design of Optical Buffer for Optical Input-Queued
Switches with Quality of Service Guarantees

研 究 生：林俊源　　　　　Student：Chun-Yuan Lin

指導教授：陳　健　　　　　Advisor：Chien Chen

國 立 交 通 大 學

資 訊 科 學 系

碩 士 論 文

A Thesis

Submitted to Department of Computer and Information Science
College of Electrical Engineering and Computer Science
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of
Master
in

Computer and Information Science

June 2005

Hsinchu, Taiwan, Republic of China

中華民國九十四年六月

輸入佇列光交換機中具服務品質保證之光暫存器設計

研究生：林俊源　　　　　　　　　　　指導教授：陳　健

國 立 交 通 大 學 資 訊 科 學 系

中文摘要

在這篇論文中，我們提出了用於光交換機的光延遲線架構，此架構能夠解決資源衝突並且利用多階層優先權佇列來確保封包在傳輸過程中的服務品質。雖然光延遲線架構已被用來解決在光交換機中的資源衝突問題，但由於光具有持續進行的特性使得封包在隨機存取管理時受到相當大的限制並同時增加了封包遺失的機率。更甚者，為了能夠依照不同優先權的封包提供服務品質保證，往往需要數組的光延於線來實現與確保服務品質。因此，我們提出了單點傳播循環的光延遲線架構，它能夠解決輸入佇列光交換機中的資源衝突問題。而為了改善在輸入佇列交換機中因連線頭端阻塞而降低的封包處理量，我們亦改善了單點傳播循環光延遲線架構使其適用於更具有效率的窗框前瞻式排程演算法。除此之外，為了能夠利用一組光延遲線來暫存不同優先權的封包，我們也設計了一套控制法則將光延遲線劃分為許多個邏輯佇列來存放封包。最後，模擬結果顯示單點傳播循環的光延遲線架構能夠有效地在維持服務品質時降低高優先權封包的遺失率，並且此架構能夠容易地在光封包交換機上實作與控制。
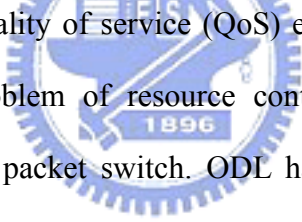
關鍵字：光延遲線、光暫存器、光交換機、服務品質、資源衝突。

On the Design of Optical Buffer for Optical Input-Queued Switches with Quality of Service Guarantees

Student: Chun-Yuan Lin   **Advisor: Dr. Chien Chen**

Institute of Computer and Information Science
National Chiao Tung University

Abstract

This paper presents a quality of service (QoS) enable optical delay line (ODL) architecture to solve the problem of resource contention and support multilevel priority queues in an optical packet switch. ODL has been used in optical packet switches to resolve resource contention; however, the packets travel continuously in ODL limits the management of random access of the packets and increases the packet loss probability. Moreover, multiple ODL sets usually are needed to realize multiple priority queues in order to support QoS. In this paper, a new Unicast Recirculatiion ODL (URODL) architecture is proposed to resolve the output contention problem in an input-queued optical packet switch. To improve relatively poor throughput due to the head of Line (HOL) blocking in the input-queued switch, we modify URODL to support a more efficient window-based lookahead scheduling algorithm. Furthermore, a control strategy is designed to turn a single set of URODL into multiple logical queues to hold different priority packets. The simulation results show our URODL model reduces high priority packet loss effectively with the capability to support QoS.

This URODL model can be easily implemented and managed in a fast optical packet switch.

Acknowledgements

I am deeply grateful to my advisor, Professor Chien Chen, for his continuous encouragement, support, and enthusiastic discussion through this research. His valuable suggestions help me to success this work.

In addition, I like to express my sincere appreciation to all the secretary staffs in the Computer and Information Science, National Chiao Tung University, for their encouragement and help. Especially, I would like to thank the members of Optical Networks and Network SoC laboratory for their adherence and help.

This thesis is dedicated to my parents for their patience, love, and encouragement. At last but not the least, thank everyone who supports and concerns me.

Table of Content

List of Figures

List of Tables

Chapter 1: Introduction

With advancement of optical device technologies, photonic packet switching is emerged as a promising technology to provide very large switching capacity [1] [2]. Packet switches based on photonics have a few advantages over their electronic counterparts. An advantage is the optical transparency. Transparency means that, except for the packet header, the packet payload can be encoded in arbitrary format with arbitrary bit rate. In addition, the optical packet switches are considered an offered solution for the future broadband packet networks in term of cost, power consumption, and footprint size.

One issue in the implementation of optical packet switching systems is the lack of optical random access memory (RAM). In every switch, resource contention is an unavoidable problem, which degrades overall switch performance. The contention problem refers to the fact that more than one packet is destined to the same output port of the switching unit in the same scheduling time slot. When contention occurs, the control unit selects one of the contending packets and the switching unit switches it to its output port. The others packets, that lose contention, are buffered for scheduling in the next time slot. Unfortunately, optical RAM suitable for buffering photonic packets has not yet been found. Instead of converting optical signal to electronic format, the alternative way is to use optical delay lines to realize an optical buffer.
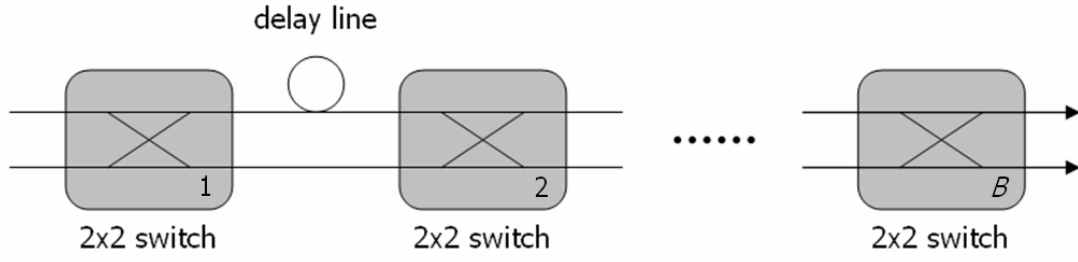
(a) optical delay line buffer I
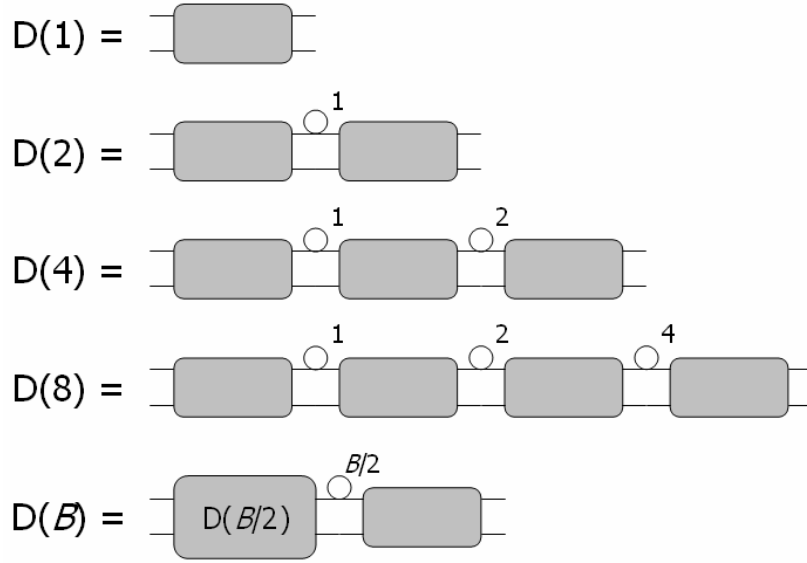
(b) optical delay line buffer II

Figure 1.   Proposed optical delay line buffer architectures.

Figure 1 shows the related works of optical delay line based optical buffer architectures [7] [8]. Both two architectures take 2x2 switches and delay line loops to store optical signal for buffering when resource contention occurs. In Fig. 1.a, B indicates the size of optical buffer, which allows optical packet to stay in optical buffer for maximal B time slot. The proposed architecture in [8] needs $B+1$ 2x2 switches, which means that the control unit must manage all 2x2 switches when an optical packet passing through this optical buffer. Although [7] proposes a preferable architecture which only needs $log_2(B)+1$ 2x2 switches, it still needs to control all 2x2 switches even though the optical packet do not need to store in the optical buffer. The

2

control operation of each 2x2 switches increase the processing time of optical packet while switching and forwarding.
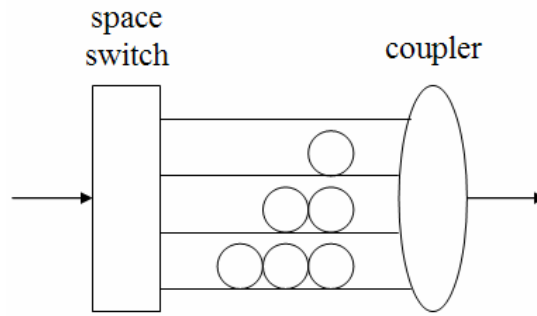
(a) traveling-type buffer
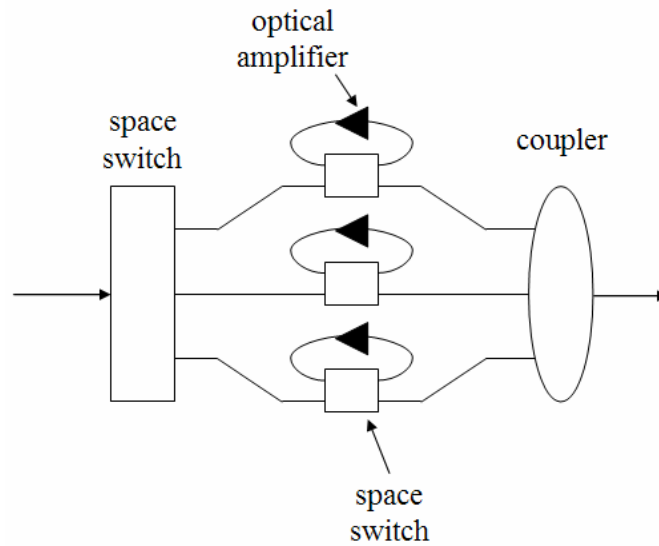
(b) recirculating-type buffer

Figure 2.   Schematic of photonic packet buffers based on optical delay lines.

In general, optical delay line based buffer can be divided into two forms: traveling-type and recirculating-type as shown in Figure 2. A traveling-type buffer generally consists of multiple optical fiber delay lines whose lengths are equivalent to multiple of delay time unit T, where T is a basic period of light traveling a single delay loop. In such traveling-type buffer, the storage time of a packet is simply

determined by the duration of a packet propagating through the length of optical fiber. Note that the number of delay lines increases as the buffer increases. Therefore, a large number of ODLs are needed to maintain a low packet loss rate. Furthermore, an optical packet is dropped due to the packet reaching the end of fix-length optical delayed line while its destine output port is not free.
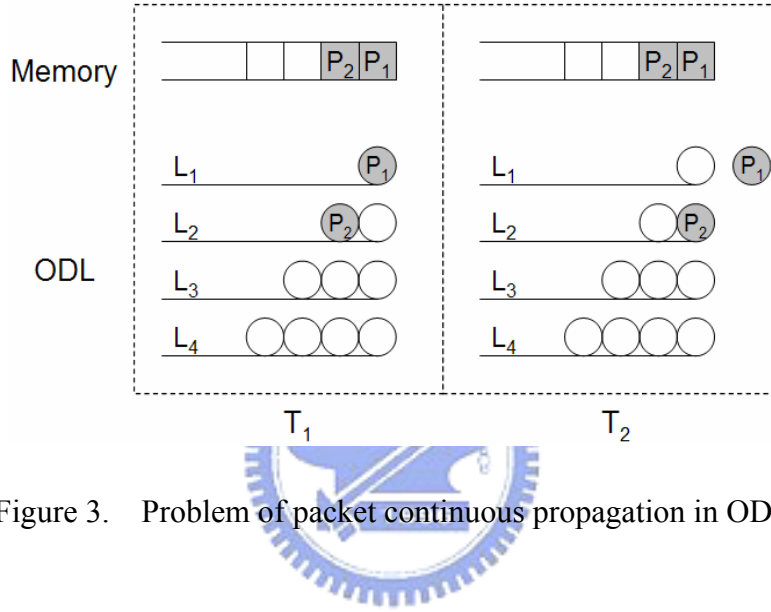
Figure 3.   Problem of packet continuous propagation in ODL.

Figure 3 describes how a packet is dropped while it travels to the end of an ODL. At time slot T_1 assume there are two packets in buffer, P_1 and P_2. If contention happens at T_2, P_1 should remain in the buffer to wait for scheduling in next time slot T_3. Unfortunately P_1 leaves ODL and is dropped because packet P_1 reaches the end of ODL even when the ODL buffer is not full. Figure 2.b schematically shows a recirculating-type buffer, which consists of multiple optical delay lines, each forming a loop with one circulation time equal to delay time unit T. The storage time of a packet in such a recirculating-buffer is determined by the number of circulation multiplied by the delay time unit T. Both of above two buffers are able to store multiple packets with the constraint that only one packet enters and leaves the buffer at a time. The recirculating-type buffer is more flexible than the traveling-type buffer

in that the packet storage time is adjustable by changing the number of circulating. The optical packets would not be dropped even through output contention is happening, since packets can be circulated back to the ODLs. The optical packets are only dropped while optical buffer is full. However, a significant problem with the recirculating-type buffer is that the signal has to be amplified during each circulation to compensate for the power loss. This results in accumulated spontaneous emission (ASE) noise [3] from the optical amplifier and severe signal-level fluctuation, which eventually limits the maximum number of recirculation.

Recently, the issue of supporting QoS guarantees in optical networks has become increasingly important [4] [5]. The switch mostly uses multiple priority queues to store packets of different priorities to manage QoS. It means that multiple sets of ODLs are needed to support multiple priority queues. However, ODL is much more expensive than electronic RAM. Besides, it remains a challenging design issue to provide multiple priority queues using a single set of ODL.

To resolve these problems, we propose a QoS enable ODL architecture, called unicast recirculation ODL (URODL) for the input-queued optical packet switch. The URODL architecture takes advantages of both traveling-type and recirculating-type buffer. It efficiently suppresses the packet loss probability from resource contention by redirecting packets back to optical delay lines with different delay lengths. Due to the recirculation capability, our URODL not only provides better random access and larger buffer size than traveling-type ODL for the same number of optical delay line. But also it significantly reduces the number of circulation comparing with recirculating-type ODL for the same delay requirement. This is because of, in URODL, multiple delay lengths instead of a single delay unit can be selected for more

flexible delay combination to satisfy different delay requirements. To improve

relatively poor throughput due to the head of Line (HOL) blocking in the

input-queued switch, a window-based lookahead scheduling algorithm is

implemented in URODL by selecting packets other than just the one at the head of the

line. Furthermore, we extend the URODL to support QoS. A simple control strategy

turns a single set of URODL into multilevel logical queues to hold different priority

packets. The lengths of different priority queues are not fixed. The variable length

priority queues in URODL are much more cost effectively than using multiple set of

ODLs to support QoS.

This paper is organized as follows. In Chapter 2, we describe the architecture of

unicast recirculation ODL. The variant of URODL to support a window-based

lookahead scheduling algorithm is proposed in Chapter 3. In Chapter 4, a detailed

control strategy to turn URODL into multiple priority queues is given. It shows that

the URODL is easy to support QoS. In Chapter 5, some simulation results show the

URODL indeed has better performance than traveling-type and recirculating-type

ODL. In supporting QoS, the simulation results show the higher priority packets have

excellent loss probability. By measuring the number of recirculation, we then can

limit the maximum number of recirculation of each packet to prevent ASE noise.

Finally we conclude the paper in Chapter 6.

Chapter 2: Unicast Recirculation ODL Architecture
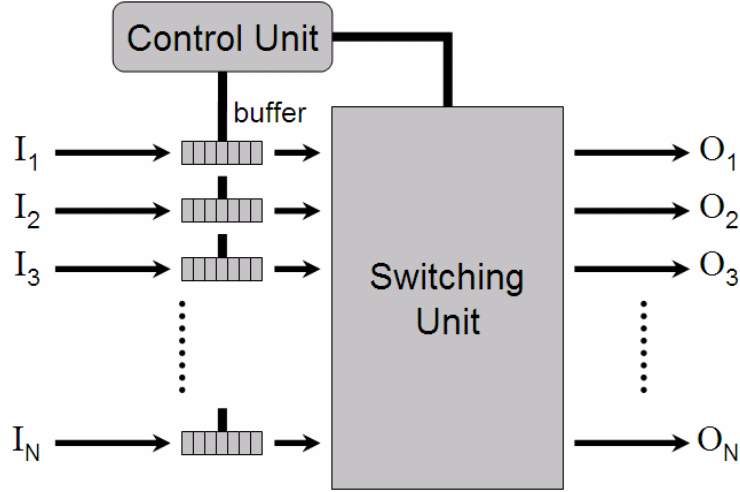
Figure 4.   Input-queued optical packet switch.

In a conventional electronic switch, RAM is used as a buffer to store packets when contention occurs. However, in all optical packet switches, ODL rather than RAM is used to store packets in optical form without optical-electronic and electronic-optical conversion. Figure 4 is input-queued optical packet switch architecture. Packets can be stored in buffer (ODLs) for several time slots if the contention happens continuously.

[6]-[8] propose several cascaded optical delay line architectures using 2x2 switch in multi-stage construction. It allows redistributing packets contending for the same resource over time and space, while keeping them in optical form. Thus, rescheduling of resource access by the contending packets can be obtained. However, some extended problems such as packets travel continuously without recirculation and QoS support still occur when using multi-stage ODL.

Another optical buffer model called wavelength routing-based buffer, which consists of a set of delay lines sitting between two arrayed-waveguide gratings (AWGs), is proposed in [9]. The first AWG switches packets to different delay lines according to packets' resident wavelengths. Note that all the delay time slots of each packet are disjoint because no more than one packet is switched to the same delay line. Therefore, wavelength routing-based buffer can be shared by all the switch input ports to act as a shared memory buffer. A wavelength routing-based buffer [9] with an added circulative loop which provides additional delay time is proposed in [10]. The AWG and erbium-doped fiber amplifier (EDFA) are also used in the loop. However, due to noise caused by EDFA, the number of packet recirculation time is still limited.
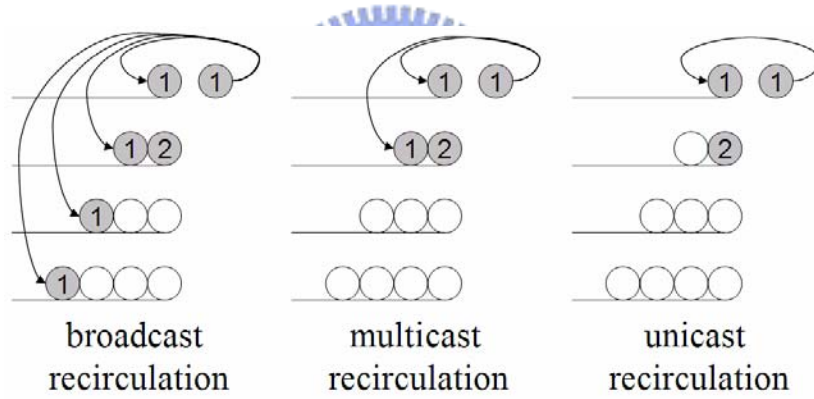
Figure 5.   Three recirculation methods in ODL. The buffer size is 4 in this example.

To eliminate the optical packets continuous propagating out of the optical delay line, recirculation ODL (RODL) redirects the propagated packet back to the buffer to store optical data. Figure 5 shows three possible recirculation methods - broadcast, multicast and unicast. In this case, broadcast RODL makes four, i.e. the buffer size, copies of propagating packet and redirect them back to all the delay lines. The redirected packet can wait for at most four time slots without being recirculated again. The control unit can select one of the four packets when it wins contention. This

ability to select from four packets is useful when contention occurs continuously. Unfortunately, the RODL with broadcast may block incoming packet when contention occurs and the input packet arrivals simultaneously. Hence, the broadcast method is inefficient and the packet loss probability is still high. The second method multicast RODL in Fig. 5 makes just two (two is the number of packets in the ODLs) copies to prevent blocking incoming packet in broadcast RODL. However, control is not easy because the number of copies is not fixed and the buffer still contains too many copies of the same data. The third recirculation method, unicast RODL (URODL), which is preferred herein, only redirects the packet back to particular delay line without any additional copy's being made. Not only does URODL excel in packet blocking performance, but also URODL can more easily be controlled than the other two recirculation methods.
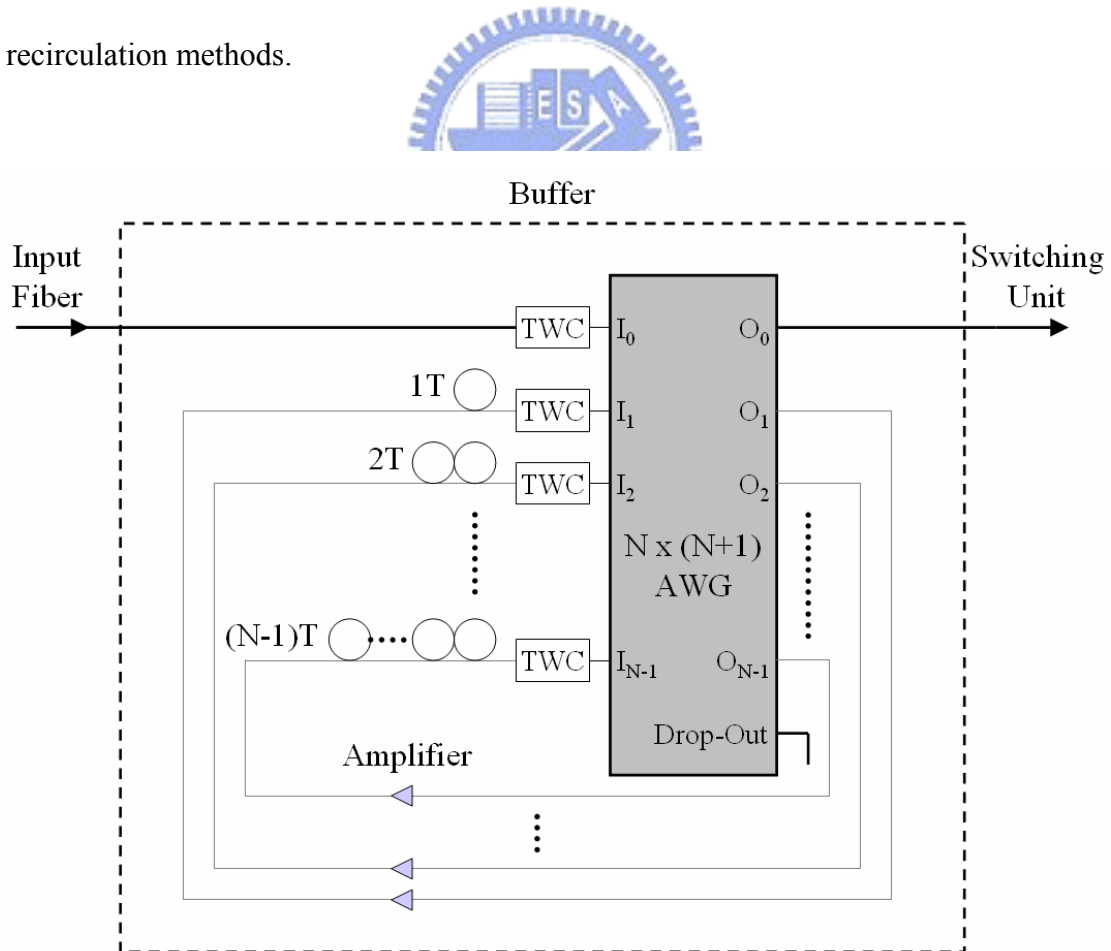

Figure 6.   URODL architecture.

Figure 6 depicts the architecture of URODL. An N**x**(N+1) (N is number of delay line) AWG and tunable wavelength converters (TWC) is used to switch optical packets to a certain delay line for buffering, pass-by, or drop-out. The optical amplifiers may be needed to compensate the insertion loss when light travels through AWG and ODL. To compare with a multi-stage delay line buffer [7]-[8], their control unit must control the entire 2**x**2 switches which cause noise and crosstalk. In general, fewer optical components not only reduce the cost of optical switches, but also imply better noise and crosstalk performance. However, URODL only needs to control TWCs to switch packets as required. The URODL is more easily implemented and more cost effective than the multi-stage delay line buffer as shown in Table 1.
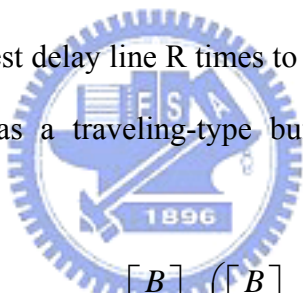
Number of Proposed Optical Buffer	Control Operations	2x2 Switch	AWG	Total Delay Loops
URODL	$O(R)$	none	1	$\dfrac{\left\lceil\frac{B}{R}\right\rceil \times \left(\left\lceil\frac{B}{R}\right\rceil+1\right)}{2}$
2x2 Switched Buffer in [7]	$O(\lceil\log_2 B\rceil+1)$	$\lceil\log_2 B\rceil+1$	none	B
CORD in [8]	$O(B+1)$	$B+1$	none	B

Table 1. Control complexity and amounts of components in proposed optical buffer architectures. Assume that all these optical buffers can delay packets *B* time slots in maximal. *R* is the maximal recirculation time in URODL.

Table 1 indicates the control complexity and the amount of components in URODL and [7]-[8]. In Table 1 we only consider an optical buffer that has maximal delay B. time slots. The processing time of each scheduling is proportional to the

control components. It includes the processing time to control all the optical components such as 2x2 switches, TWCs and AWG. In [7]-[8], the delay line architectures are organized as cascaded construction and the processing time is long due to the result that the control unit has to control each 2x2 switch. The first column "Control Operations" in Table 1 indicates that how many control operations a packet needs when it passes through the buffer. URODL only needs $O(R)$ because a packet can be redirected back to delay line R times at most, where R is the maximal recirculation time. In [7]-[8] each packet needs $O(\lceil \log_2 B \rceil +1)$ and $O(B+1)$ control operations respectively due to their cascaded architecture. The "total delay loops" in Table 1 indicates the number of total delay loops needed to support maximum delay B. In URODL the longest delay line only needs $\lceil B/R \rceil$ delay loops, since packets can be redirected back to the longest delay line R times to achieve maximum delay time B. When R=1 URODL works as a traveling-type buffer. The total delay loops of URODL is

$$
1 + 2 + \cdots + \left\lceil \frac{B}{R} \right\rceil = \frac{\left\lceil \dfrac{B}{R} \right\rceil \times \left(\left\lceil \dfrac{B}{R} \right\rceil + 1 \right)}{2} . \tag{1}
$$

Therefore, if number of delay loops needed in URODL is concerned, we can increase number of recirculation R.

11

Chapter 3: Window-Based Lookahead URODL

Due to Header of Line (HOL) blocking in the input-queued optical packet switch architecture, the URODL can only achieve 58% of throughput theoretically [11]. When a HOL packet cannot be cleared because of its loss in output contention; it may block the packet behind, which may be destined for an idle output. This phenomenon is called HOL blocking. In order to increase throughput, the window-based lookahead scheduling is used to increase scheduling efficiency. That is, at the beginning of each time slot, the first w packets in each input queue sequentially contend for access to the switch output where w is the window size. Due to output conflict, some inputs may not be selected to transmit the HOL packets, and they send their second packet in line to contend for access to the remaining output that are not yet assigned to receive packets in this time slot. This contention process repeats up to w times in each time slot. The throughput is significantly improved on increasing the window size from 1 to .2, 3, and 4. However, the improvement saturates with large number of window size [12].

To improve switch performance, we propose a window-based lookahead URODL with window size w = 3 as shown in Fig. 7. Comparing with URODL, the window-based lookahead URODL with window size w = 3 adds only two more delay lines which can delay one time slot. The first three packets in a queue are put in the three one time slot delay lines sequentially. The packet in the first delay line contends for the output ports first. If the output contention happens to the packet in the first delay line, then the packet in the second delay line contends for the available output ports, and so on.

In the window-based lookahead URODL we prefer small window size, since the larger window size used, the more optical components needed and the longer scheduling time required. Nevertheless, the simulation result shows small window size can improve the switch performance effectively.
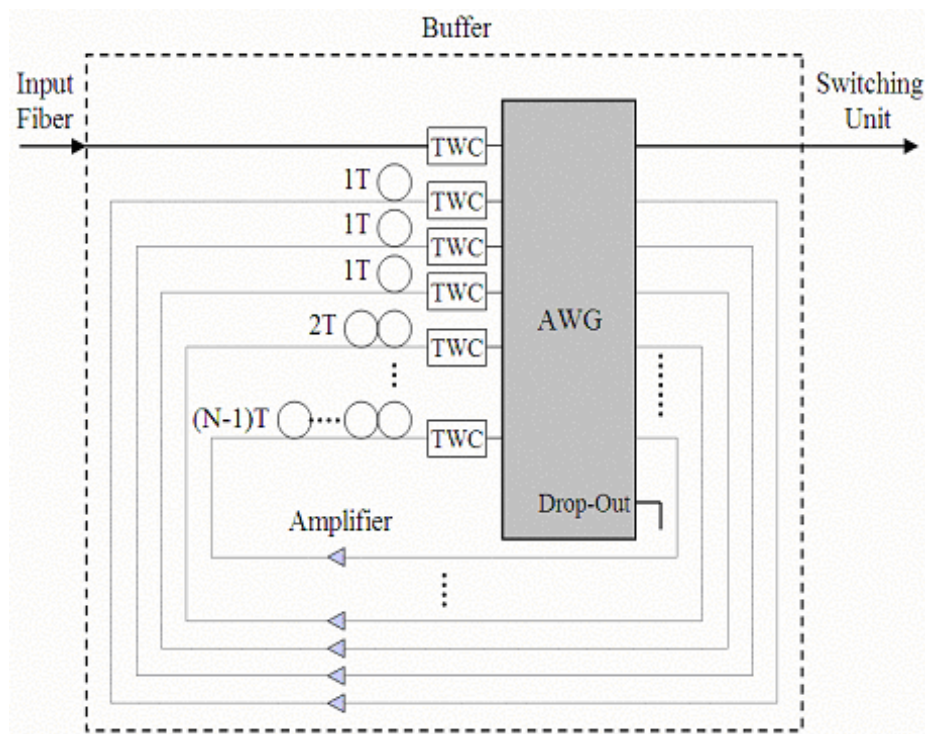

Figure 7.   Window-based lookahead URODL with window size w=3.

Chapter 4: QoS Support in URODL

Generally, a switch uses several queues of different priority levels to support QoS. URODL can support multiple priority queues in a single set of ODL. The lengths of different priority queues are not fixed. The variable length priority queues in URODL are much more cost effectively than using multiple set of ODLs to support QoS. Hence, URODL is an efficient component for temporal storage of multiply priority optical packets.
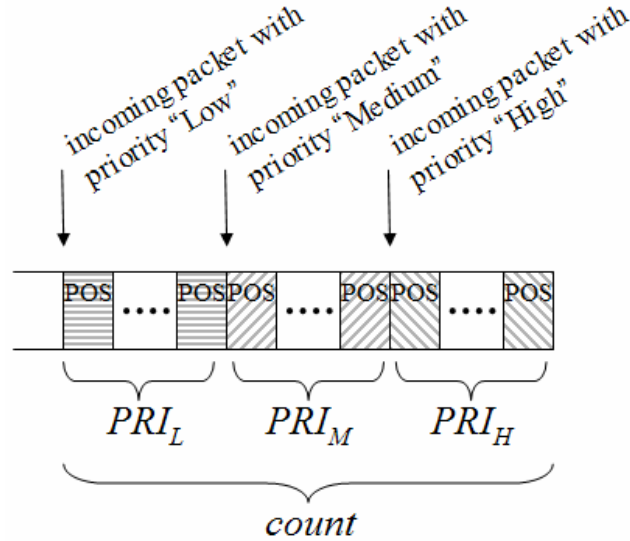

Figure 8. Diagram of control strategy of URODL. count: number of packet in URODL; PRI_H: number of high priority packets; PRI_M: number of medium priority packets; PRI_L: number of low priority packets; POS: each packet's position in URODL.

URODL is hereafter assumed to support three priority levels - high, medium and low. Figure 8 shows an example of URODL with three multilevel variable priority queues. Each packet has its own POS value, which indicates the position of the packet in the queue. If a packet's POS is 4, it is allocated to the fourth position from the head

of the queue. When this packet enters AWG, it will be switched to O_4 in Fig. 6 and is delayed for four time slots. PRI_H, PRI_M and PRI_L are the numbers of packet with high, medium and low priority respectively, and count specifies how many packets are in the URODL; it equals to the sum of PRI_H, PRI_M and PRI_L. The following shows the details of our algorithm.

---

1. $count = 0$;
2. **if** incoming packet enters and $count < queue\_size$ **then**
    switch "incoming packet priority"
      case "High":$POS_{incoming} = PRI_H$;
               $PRI_H = PRI_H + 1$;
               $POS = POS + 1$, for packet whose priority is "Mid" and "Low";
      case "Mid": $POS_{incoming} = PRI_H + PRI_M$;
               $PRI_M = PRI_M + 1$;
               $POS = POS + 1$, for packet whose priority is "Low";
      case "Low": $POS_{incoming} = PRI_H + PRI_M + PRI_L$;
               $PRI_L = PRI_L + 1$;
               $count = count + 1$;
3. **if** no contention and $count > 0$ **then**
    if $PRI_H > 0$ **then** $PRI_H = PRI_H - 1$;
      else if $PRI_M > 0$ **then** $PRI_M = PRI_M - 1$;
      else $PRI_L = PRI_L - 1$;
    $count = count - 1$;
    $POS = POS - 1$, for all packets;
4. AWG destines an incoming packet (if exists) and packets at head of delay line to their corresponding output ports
5. jump 2.

When an incoming packet enters this URODL, it is classified by priority and inserted into a certain position of the queue. If the packet priority is "high", the packet will be inserted between the last high and the first medium priority packets of the queue, its POS is PRI_H+1. If the priority is "medium", it is inserted between the last

medium and the first low priority packets, and its POS is PRI_H+PRI_M+1. If the priority

is "low", it is inserted at the end of queue, and its POS is $PRI_H+PRI_M+PRI_L+1$. This

algorithm ensures that no lower priority packet can be switched out before a higher
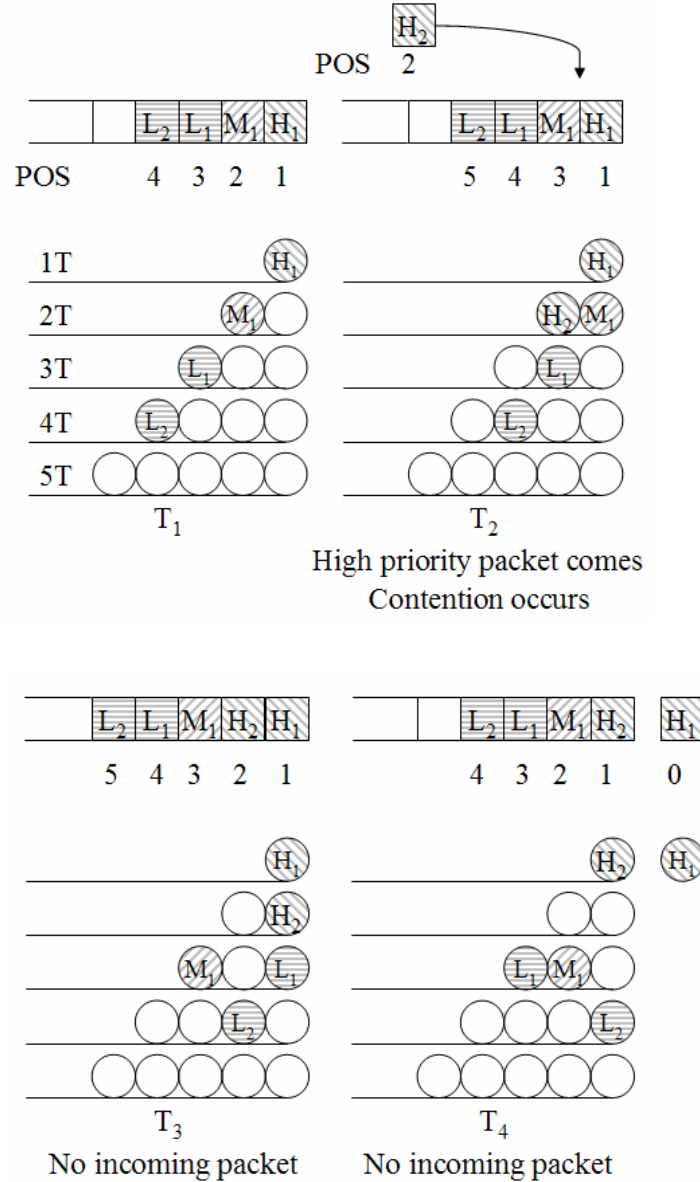
priority packet.

Figure 9.   Example of packet allocation in URODL.

Figure 9 is an example shown in time slot by time slot. The control strategy

works in each time slot as follows.

T_1: Initially assume four packets in URODL: one is high priority, one is medium priority and the other two are low priority. In this time slot, PRI_H and PRI_M are both one and PRI_L is two.

T_2: In this time slot, a high priority packet enters ODL; its POS is assigned as two because one high priority packet is present in ODL, and then PRI_H increases by one unit. POS of each medium and low priority packet increases by one, since the incoming packet is inserted in front of these packets. Resource contention also occurs in this time slot, so H_1 is redirected back to delay line 1T to prevent losses.

T_3: The resource contention occurs continuously. Packets located on the head of the delay line are redirected back into the delay line according to their own POSs. Hence, H_1 is switched to 1T to cause a delay of one more time slot and M_1 is switched to 3T. In fact, redirecting a packet back to URODL does NOT reorder the sequence of all the packets. Only a packet who's POS is equal to zero can be switched to O_0 as in Fig. 6 and can leave URODL. No packet can leave URODL if some other packets exist whose POS is smaller.

T_4: In this time slot, no incoming packet and no contention occur. POS of all packets decreases by one. H_1 leaves ODL because its POS is equal to zero, it is switched to O_0 as in Fig. 6 and enters switching unit. H_2 and L_1 are redirected back to ODL to wait for scheduling in the next time slot.

Chapter 5: Simulation Results

This chapter presents the simulation results concerning the packet loss probability and average recirculation time. We consider input-queued switch in evaluating the performance of URODL. Only the packets at the head of the queues (i.e. the packet who's POS is zero) can content for their destined outputs. In window-base lookahead URODL, the first w packets in each input queue sequentially content for access to the switch outputs. In our simulation, packets arrive at each input port following Bernoulli process with different offered load. The destination of each incoming packet is uniform distributed. That is, for a given offered load ρ in an NxN switch, the packet arrives at each input port with probability ρ and each packet destines to a particular output port with probability $1/N$. The multilevel priority URODL serves packets on a "straight priority" basis, i.e., if there exists a higher priority packet in multilevel priority URODL, and then no lower priority packets can be served. When multilevel priority URODL is full and an incoming packet whose priority is higher than one of the packets in URODL, the packet at end of URODL is dropped and the incoming packet is inserted to a position determined by the control algorithm detailed in Chapter 4. URODL is hereafter assumed to support three priority levels – high, medium and low.
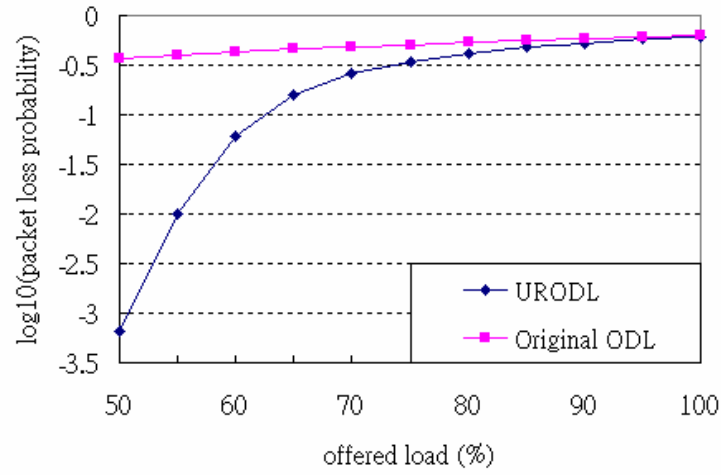
(a) buffer size = 16
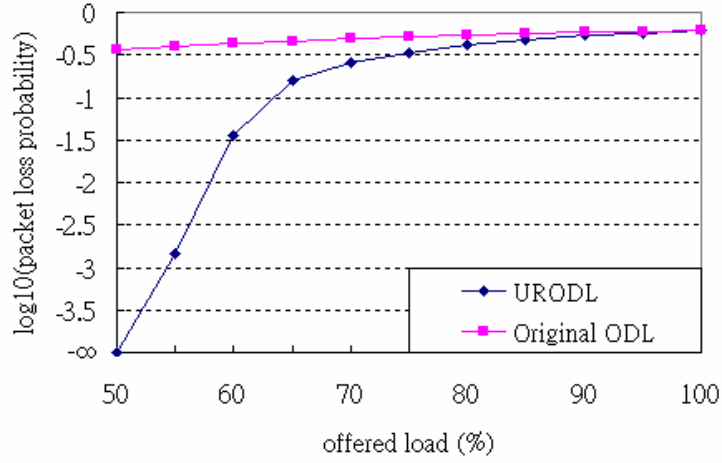
(b) buffer size = 64

Figure 10.   Simulation results concerning packet loss probability of 32-port
input-queued switch with different buffer size.

Figure 10 presents the simulation results of traveling-type ODL with and without recirculation in a 32x32 switch. Fig. 10.a and 10.b plot the distribution of packet loss probability with buffer sizes of 16 and 64 respectively. In Fig. 10, the gap between the two curves is caused by the packet traveling out of ODL, while contention happens. The traveling-type ODL cannot hold the packets propagate out of ODL, but URODL can do so by recirculation. The simulation result shows URODL can reduce the packet
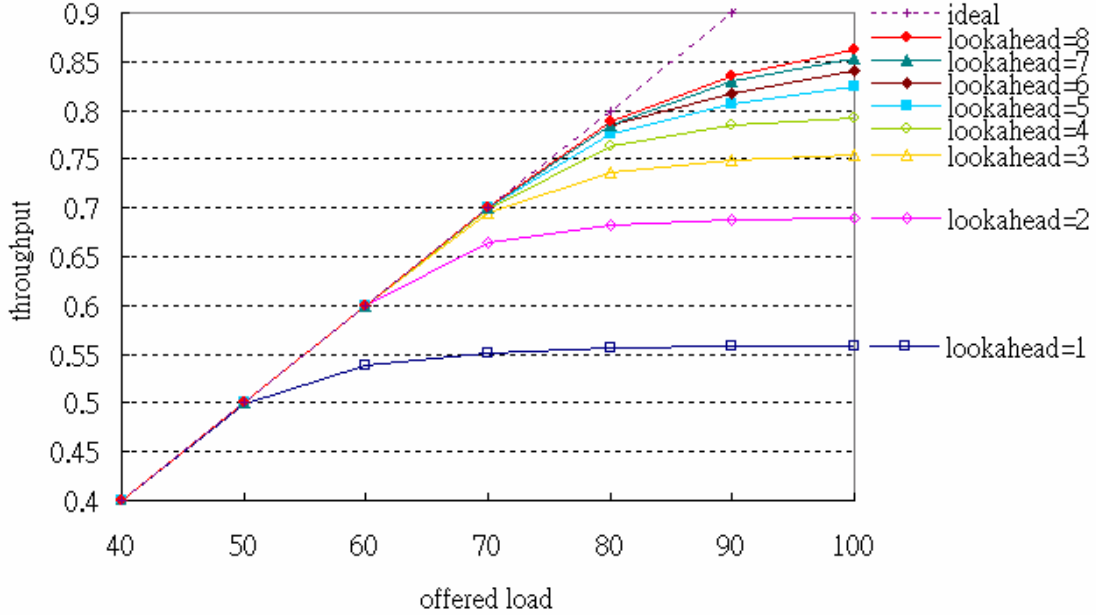
19

loss probability significantly.

Figure 11.　The throughput of different window-based lookahead architecture in 32-port switch with buffer size 16.

In Fig. 11, we show the simulation result for the proposed window-based lookahead URODL architecture with window size w increases from 1 to 8. Notice that when a window size w = 1 corresponds to the original URODL. The maximum throughput can be achieved are 0.56, 0.69, 0.75, 0.79, 0.82, 0.84, 0.85, and 0.86 respectively for various window size w = 1, 2, 3, 4, 5, 6, 7 and 8. The throughput improves significantly (about 23%) by increasing window size w to 2. The window size of 4 improves the maximum throughput by about 5% compared with the window size of 3. As shown in the simulation results, the improvement decreases to about 1%, as the window size increases from 6 to 7, and 7 to 8. Therefore, in order to reduce the HOL blocking people can choose the window-based lookahead URODL with window size of 2, 3 or 4 instead of large window size which needs more additional optical components and slower scheduling time.
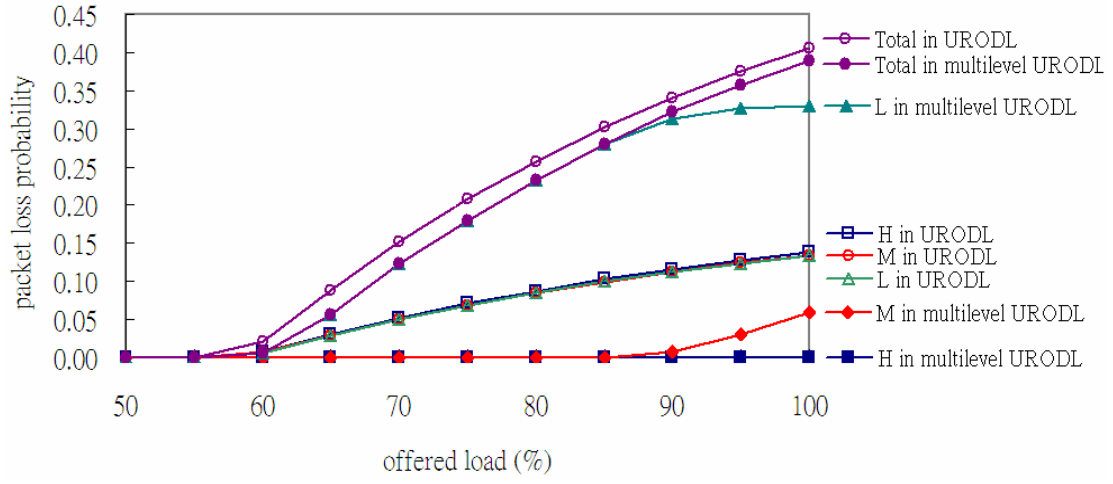
(a) (H, M, L) = (33.3%, 33.3%, 33.3%)
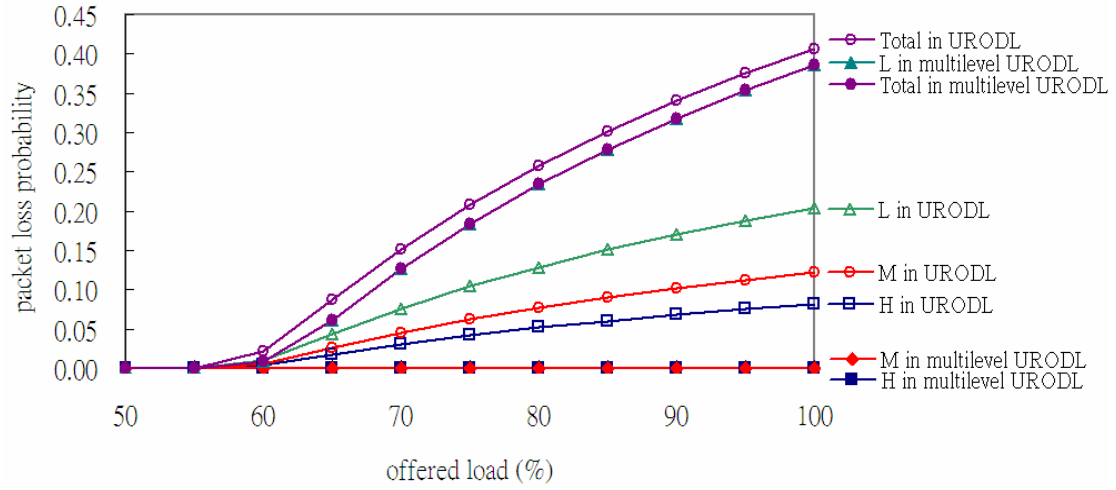
(b) (H, M, L) = (20%, 30%, 50%)
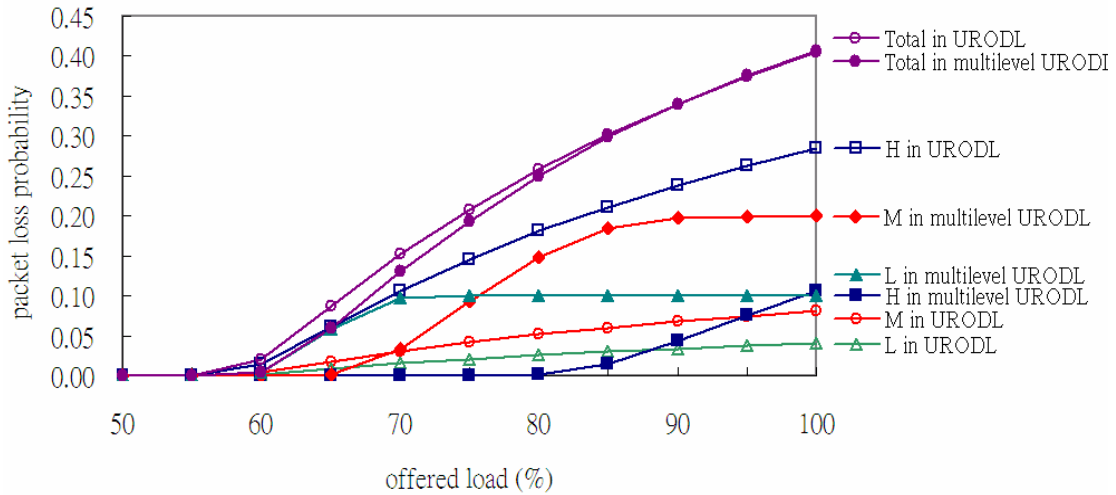
(c) (H, M, L) = (70%, 20%, 10%)

Figure 12.   Simulation results of packet loss probability of 32-port input-queued switch with different distribution of packet priority.

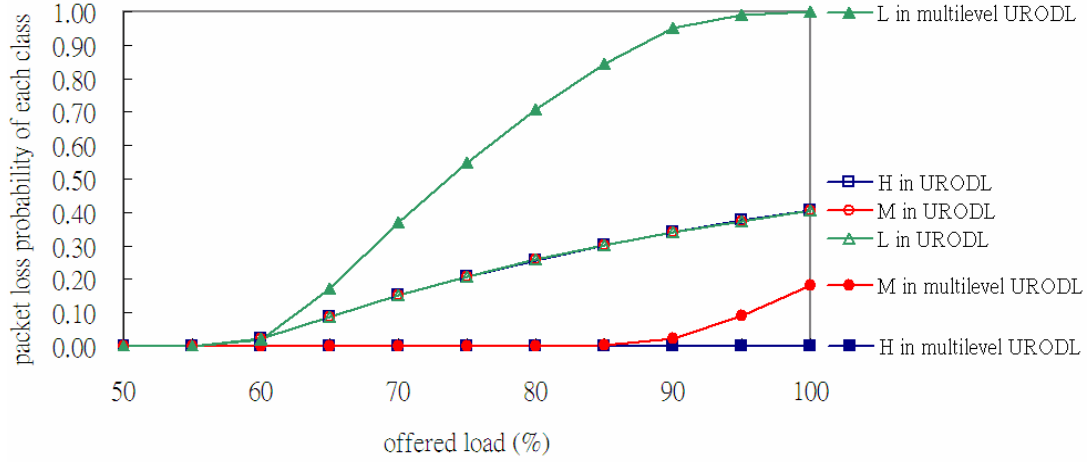(a) (H, M, L) = (33.3%, 33.3%, 33.3%)
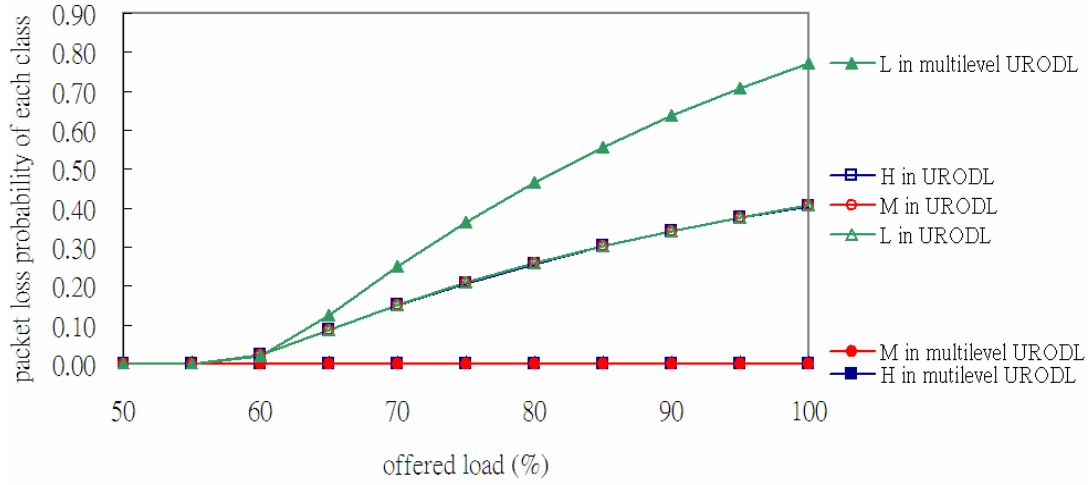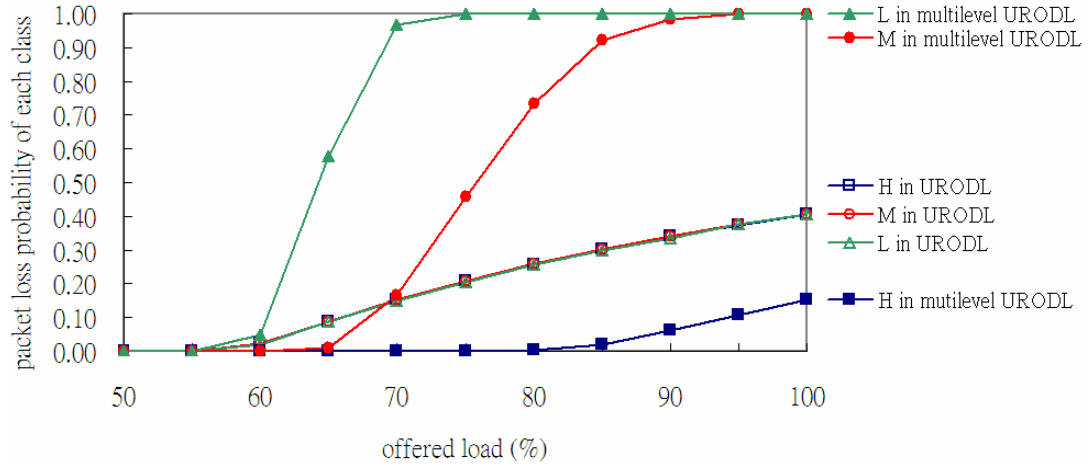

(b) (H, M, L) = (20%, 30%, 50%)


(c) (H, M, L) = (70%, 20%, 10%)

Figure 13.   Simulation results of packet loss probability of 32-port input-queued switch with different distribution of packet priority.

Figure 12 and Figure 13 depicts the simulation results of URODL with QoS support, and compares it to the URODL without QoS support, using a 32x32 switch and a buffer size of 16. In Fig. 12 the packet loss probability is computed by number of packet loss in each class divided by total number of incoming packet. In Fig. 13 the relative packet loss probability is computed by dividing the number of lost packets in each class by the total number of incoming packets in each class. Three priority levels, high, medium and low, are assumed. The URODL without QoS support serve packets without distinguishing the packet priority. The control strategy we proposed in this paper for the QoS enable URODL is used to simulate the packet loss probability among three priority levels. Three different combinations of incoming packet priority are used to observe the packet loss probability. In uniform priority distribution as shown in Fig 12.a, the low and medium priority packets of multilevel priority URODL inherit the packet loss probability of the high priority packets, so the loss probability of high priority packets almost reaches zero. Even in non-uniform priority distribution shown in Fig. 12.b and Fig. 12.c, the high priority packet loss probability is still close to zero. In Fig 12.c, the high priority packets take 70% of overall incoming packets, the high priority packet loss probability remains zero under 80% offered load. However, in URODL without QoS support, the loss probability associated with each priority level is proportional to the distribution of the incoming packet priority; the higher priority packet may be dropped even when the lower priority packet is in the queue. Fig 12.a shows the packet loss rates are same among high, medium and low priority packets while URODL without QoS support is used. Furthermore, in Fig 12.c and Fig 13.c, we can observe when the loss rate of high priority packets reaches zero under 80% traffic load, the loss rate of medium priority begins decreasing. It happens again when the loss rate of medium priority packet gets down to zero, the loss rate of low priority packets starts decreasing. The results show

that the higher priority packets are given the preemptive authority to prevent loss in multilevel priority URODL. In Fig. 12.c the packet loss probability of medium priority packets in multilevel priority URODL is higher than the low priority packets, this is because the medium priority packets take 20% of overall incoming packets and the low priority packets only take 10% of overall incoming packets. Figure 13 shows that the relative packet loss of each priority class. In Figure 13 we see more clearly about packet loss probability among three different priority classes. They show that the relative packet loss probability of higher priority packets is always less then the lower priority packets in multilevel priority URODL. However, in URODL without QoS support the relative packet loss probability among three priority classes are the same. This shows that our multilevel priority URODL can guarantee QoS by distinguishing packets among the three different priorities.
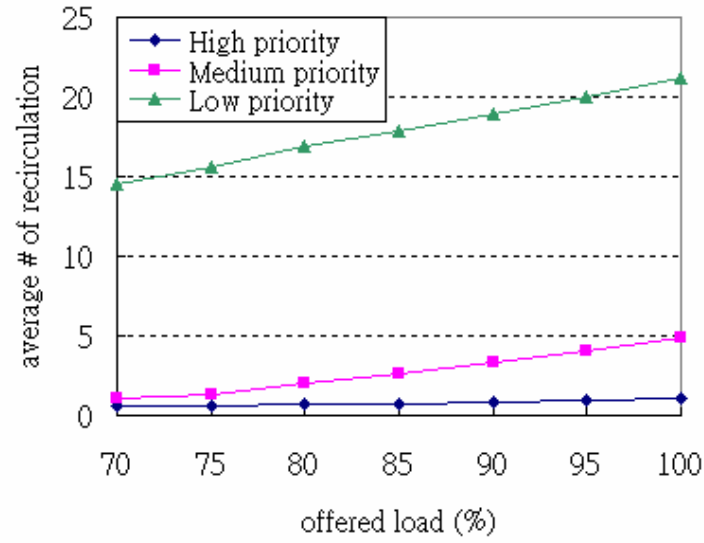
(a) average number of recirculation
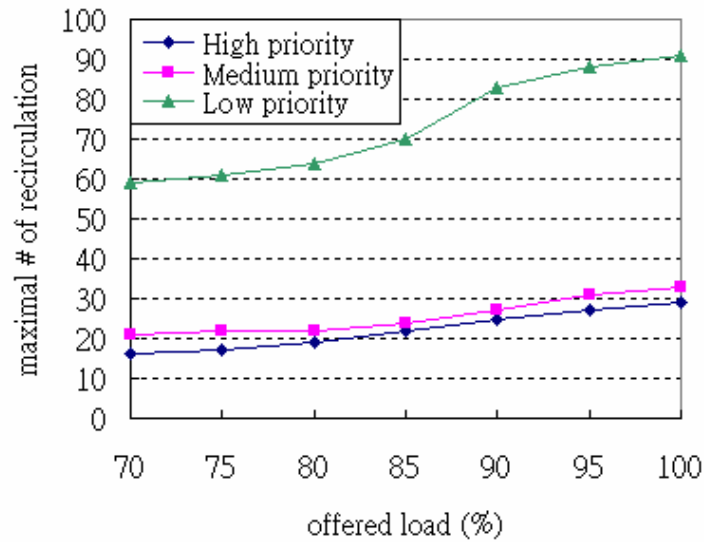
(b) maximal number of recirculation

Figure 14.   Average and maximal times of recirculation of 32-port switch with buffer size 16.

In simulating packet loss probability of URODL and multilevel priority URODL, statistics are also gathered on the average and maximal times of recirculation in multilevel priority URODL, as shown in Fig 14. Because of insertion loss appearing invariably in AWG and ODL, the times of recirculation while a packet is waiting for scheduling may cause the intensity of light to decline. An optical amplification can be

used to URODL to boost the optical signal. However, ASE noise limits the number of times that a packet can travel through the optical amplifier. In URODL, the average times of recirculation in Fig 14.a indicate that the high and medium priority packets require only few recirculations on the average. The high priority packets experience average less than two recirculations. The maximal number of recirculations in Fig 14.b shows that in the worst case a low priority packet could be recirculated in a large number of times. Even though the lower priority packets may be dropped due to large number of recirculation, the optical signal integrity of higher priority packets can be guaranteed.

Chapter 6: Conclusion

In this paper, we proposed a novel URODL architecture to reduce the high packet loss probability in ODL. Although several traveling-type fiber delay line buffer architectures have been previously proposed, none of them can keep the optical packets propagated at the end of delay line while they lose contention due to output conflict. Recirculation in URODL can redirect propagated packets back to the buffer to wait for scheduling in next time slot. All the packets can be stored in URODL without dropping packets except when the buffer is full. In order to improve input-queued switch performance, we develop a variant of URODL to alleviate head of line blocking by window-based lookahead selection. With URODL architecture proposed in this paper, multilevel priority queues with an efficient control strategy can be supported. Multilevel priority URODL supports multiple priority queues in a single optical delay line set. A flexible priority management scheme efficiently takes advantage of shared buffer space. It substantially reduces the packet loss probability by several orders of magnitude for high and medium priority packets when supporting QoS.

The simulation results show that URODL can reduce the packet loss probability, compared to conventional traveling-type ODL which cannot hold packets in the buffer continuously. The window-based lookahead URODL furthermore improves 20% of switch performance. While supporting QoS guarantees, multilevel priority URODL also provides negligible loss probability of high priority packets. Finally, the number of recirculation is investigated due to the concern of ASE noise from the optical amplifier in recirculating-type buffer. The simulation results show that average

number of recirculation is less than two for the high priority packets. Therefore, URODL is much more practical than the conventional recirculating-type ODL in the implementation of optical switches.

References:

[1] F. Callegati, G. Corazza, C. Raffaelli, "Design of a WDM optical packet switch for IP traffic," Proceedings of IEEE GLOBECOM 2000, San Francisco, pp. 1283-1287, November 2000.

[2] P. Green et al., "All-optical packet-switched metropolitan-area network proposal," J. Lightwave Technol., Volume 11, pp. 754-763, May 1993.

[3] A.Dickson and R.S. Tucker, "Noise accumulation in fiber loop memories," in Proc.17th Australian Conf. Optic. Fiber Technol., postdeadline paper, pp.4, 1992.

[4] Myungsik Yoo, Chunming Qiao, Sudhir Dixit, "QoS performance of optical burst switching in IP-over-WDM networks," IEEE Journal on Selected Areas in Communications, Volume: 18 Issue: 10, pp. 2062-2071, Oct 2000.

[5] Callegati F., Corazza G., Raffaelli C., "Exploitation of DWDM for optical packet switching with quality of service guarantees," IEEE Journal on Selected Areas in Communications, Volume: 20 Issue: 1, pp. 190-201, Jan 2002.

[6] Harai H., Wada N., Kubota F., Chujo W., "Contention resolution using multi-stage fiber delay line buffer in a photonic packet switch," IEEE International Conference on Communications, vol 5, pp. 2843-2847, May, 2002.

[7] Hunter, D.K., et al., "2×2 buffered switch fabrics for traffic routing, merging, and shaping in photonic cell networks," Journal of Lightwave Technology, Volume: 15 Issue: 1, pp. 86-101, Jan 1997.

[8] Cerisola, M., Chang-Jin Suh, et al., "CORD: contention resolution by delay lines," Selected IEEE Journal on Areas in Communications, Volume: 14 Issue: 5, pp. 1014-1029, Jun 1996.

[9] Wen De Zhong, Rodney S. Tucker, "A new wavelength-routed photonic packet buffer combining parallel delay lines with delay-line loops," Journal of Lightwave Technology, Volume: 19 Issue: 8, pp. 1085-1092, Aug 2001.

[10] Wen De Zhong, Rodney S. Tucker, "Wavelength routing-based photonic packet buffers and their applications in photonic packet switching systems," Journal of Lightwave Technology, Volume 16, No. 10, pp. 1737-1745, October 1998.

[11] M.G. Hluchyj and M. J. Karol, "Queuing in high-performance packet switching," IEEE JSAC, 6(9), pp.1587-1597, Dec. 1988.

[12] H. Jonathan Chao, Cheuk H. Lam, Eiji Oki, "Broadband packet switching technologies," John Wiley, New York, 2001.

[13] Cheng-Shang Chang, Duan-Shin Lee, and Chao-Kai Tu, "Using Switched Delay Lines for Exact Emulation of FIFO Multiplexers with Variable Length Bursts," IEEE INFOCOM, Volume 3, pp. 1998-2007, 30 March-3 April 2003.

[14] I. Chlamtac, A. Fumagalli, and C.J. Suh, "A Delay line Receiver Architecture for All-Optical Networks," IEEE INFOCOM, pp 419-429, 1996.

[15] Y. Liu, M. T. Hill, N. Calabretta, H. de Waardt, G. D. Khoe, and H. J. S. Dorren, "All-optical buffering in all-optical packet switched cross-connects," IEEE Photon. Technol. Lett. 14, 849-851 (2002).

[16] J.-T. Tsai, "Architectures of 2-to-1 COD Multiplexers for Asynchronous Packet Arrivals," Proc IEEE BSS'97, pp.106-111, December 1997.