A Smart Photonic ATM Switch Architecture with Compression Strategy

Shiann-Tsong Sheu, Yang-Han Lee, and Chih-Chiang Wu

***Abstract*—Generally, the limitations of optical delay line and link capacity limit the switching efficiency in the photonic asynchronous transfer mode (ATM) switch. Under the constraints, a smart photonic ATM switch designed for high-speed optical backbone network should have some fast switching strategies so that the congestion can be avoided or reduced. In this paper, we will propose a novel smart photonic ATM switch architecture with a novel compression strategy. In the smart architecture, while more than two frames are destined for the same destination, the losers will be queued and compressed to reduce the degree of congestion. Therefore, not only the total switching time (TST) can be reduced but also the scarce buffer is able to store more incoming cells. To meet the high-speed switching performance, a simple and efficient compression decision algorithm (CDA) is proposed. The timing of employing compression strategy and the saturated performance of proposed strategy are analyzed. Simulation results show that compared to the conventional photonic ATM switch without compression strategy, the proposed strategy offers a much better performance in terms of queueing delay.**

***Index Terms*—Asynchronous transfer mode (ATM) switch, compression, smart photonic.**

I. Introduction

ASYNCHRONOUS transfer mode (ATM) is the most promising switching technology for high-speed packet switching technology [1], [2] and the better technical and commercial solutions satisfying the increasing demand for heterogeneous communications and tremendous bandwidth requirements. So far, many electrical switches are still limited by the electronic switching speed. Therefore it is very important to investigate photonic or optical switching technologies for the future high-speed optical backbone networks. Moreover, ATM networks provide each connection with various quality-of-service (QoS) levels. To do this, the ATM protocol imposes specific switch requirements in terms of cell length, header translation, cell ordering that make the design of an optical switch (OS) becomes more difficult [6]. In [6], Lockwood *et al.* implemented a fully functional 128 Gb/s photonic ATM switch. Ultrafast photonic ATM switch with optical output buffers had been proposed in [8]. Using the multi-cell transfer scheme in the optical backbone network for the photonic ATM switch had been analyzed in [4] to improve the backbone transmission efficiency. In [5], Choi *et al.* investigated the photonic ATM switch architecture for WDM optical networks. In [9], Zhong *et al*. used the modular and expandable switch architecture to obtain the extra high-speed data rate of Terabits per second. Some novel optical bit-interleave multiplexing ATM switch architectures have been proposed in [10]. This research offers the possibility of implementing the high-performance photonic ATM switches.

Because the diverse requirements of services, the traffic load in networks becomes more heavy and unpredictable. Once the transmission capacity or the switching speed cannot afford it, the cell loss rate and the QoS of service will be degraded accordingly. In ATM-based networks, call admission control (CAC) makes the important decision of whether or not to accept a new call. In general, the decision is made according to whether the available bandwidth can support it without degrading the QoS of existing calls. A well-known approach, which is denoted as usage parameter control (UPC), tries to utilize some simple traffic parameters (e.g., peak cell rate and average cell rate, etc.) to describe the behavior of services. Based on these provided traffic parameters, the controller will compute the required bandwidth for it. If the remainder bandwidth is enough, the call is accepted; otherwise it is rejected. However, it is still difficulty to precisely predict and describe the behavior of services in advance. That is, when burst traffic arrives in network and sustains for a considerable time, lots of cells may be dropped and the QoS (including cell delay, delay jitter, etc.) may be violated. To avoid transmitting bulk data on network, the host of user often compresses data before delivering. During the progress of switching/transmission, the data will not be further considered in data link layer or physical layer in order to speedup the routing/switching operation. Note that when the network loaded is light (heavy), the buffer in switch is almost empty (full), and the following incoming cells will be forwarded (dropped). In other words, if the switching capability is able to deal with the incoming traffic loaded, all packets/cells can be switched immediately without latency. Contrarily, as traffic load excesses the service rate or the output link capacity, most of cells will be stored in queue and wait for a long time before delivering. Because of the limitation of available buffer space as well as the nature of destination contention traffic, the congestion often occurs in the ATM switch. As the front cell fails in contention, the following cells even with different destination port will be delayed also. Such problem is known as the head-of-line (HOL) problem in networks. Intuitively, the HOL problem can be solved by a scheduling algorithm or employing the output queueing or central queueing disciplines. The former first scans a number of buffered cells in each input queue and adjusts their outgoing sequence to minimize contention under the sequence ordering

Manuscript received February 22, 2000; revised April 22, 2000.

The authors are with the Department of Electrical Engineering, TamKang University, Tamsui, Taipei Hsien, Taiwan 251, R.O.C. (e-mail: stsheu@ee.tku.edu.tw; yhlee@ee.tku.edu.tw).

Publisher Item Identifier S 0733-8724(01)00412-1.

of a connection is still maintained. However, because this problem has been proved as a NP-complete problem and the heuristics algorithms may take a considerable time to find out the near optimal scheduling pattern [3], most of solutions may slow down the switching efficiency and increases the difficulty of implementation on high-speed ATM switch. In the output queueing approach, it adopts a fast switching transfer medium (switching fabric) to switch all incoming cells into their desired output buffers. In principle, all inlets can have simultaneously a cell destined to a single outlet. To ensure that no cell is lost in transfer medium before it arrives at the output buffer, the cell transfer must be performed at N times the speed of the inlets (where N is the number of inlets). Moreover, such architecture must be able to write N cells in the queues during one cell time. In the central queueing approach, the queueing buffers are not dedicated to a single inlet or outlet; but shared among all inlets and outlets. In this case, all incoming cells will directly be stored in the central queue. Every outlet will select the cells that are destined for him from the central memory in a FIFO discipline. Note that the read/write discipline of the central queue is not a simple FIFO discipline and this increases the complex of memory management. Considering an $N \times N$ ATM switch, if the link speed of each input port is B Mb/s, the switching fabric in both output and central queueing approaches is required to provide the switching speed at least $N \times B$ Mb/s. Assume the memory can be accessed by W bits in parallel but cannot support simultaneously read and write operations. In the case of output queueing, the memory access time should be less than or equal to $W/((N+1) \times B)$. Similarly, in central queueing approach, the memory access time should be less than or equal to $W/((N+N) \times B)$. Let $N = 8$, $B = 150$ Mb/s, and $W = 32$ b. The memory access times for output queueing and central queueing disciplines are 23.7 ns and 13.3 ns, respectively. This causes a very expensive ATM switch architecture. Contrarily, in the input queueing architecture, the required memory access time is only $W/((1+1) \times B) = 106.7$ ns. We also note that, no matter what kind of discipline is adopted, the incoming optical signals must be converted into electrical signals before storing into memory and converted into optical signals before switching out. This will slow down the high transmission capacity of optical link.

In this paper, we will propose a smart photonic architecture with photonic compression technology. This smart architecture includes the frame processor (FP), the cell processor (CP), the optical delay queue (ODQ), and the optical data compressor (ODC) as well as a simple photonic switch fabric. While more than two frames (consisting of cells for the same output port) in different paths content with the same output port, the losers will be queued in the optical delay line and compressed by ODC to reduce the degree of contention in the near future. The basic idea of compressing data is to group a number of consecutive signals as one signal pulse with particular optical power level. As a compressed frame has been successfully switched to the corresponding output port, it must be decompressed to restore original cell string (bit stream) before forwarding out this switch. Based on this concept, the total switching time (TST) and the required buffer size can be reduced significantly. This introduces the smart photonic ATM switch architecture. We note that there are several researches have been proposed to compress data in physical layer to enhance the transmission performance. In [11], a physical layer compression technique has been proposed. It is based on a feed-forward fiber-optical delay line structure. Based on the same structure, a compression/decompression subsystem is proposed in [12] to perform ultrahigh-data-rate compression and decompression of optical time-division-multiplexed (TDM) packets for photonic packet switching networks. The basic concept is that the time interval between two adjacent data bits is being shortened to achieve a higher transmission rate. In practice, this subsystem is used for data-rate conversion and usually plays the role of edge ATM switch if any. If one employ the compression and decompression techniques in the front and the rear of a photonic ATM switch, respectively, he may suffer from designing high-speed switching fabric and achieving a shorter decision time. This increases the design complexity. Contrarily, our proposed compression strategy will not increase the transmission speed. This will provide a relative low-cost photonic ATM switch architecture. It is worth noting that the compression/decompression subsystem can also be used with our proposed strategy.

In order to become an attractive and practical architecture, the smart photonic switch architecture should also concern the following two basic issues.

1) How to determine the timing of employing the compression strategy.
2) How to prevent a situation where the output link is available (idle), meanwhile all cells/frames are still being compressed. (This is due to the fact that, once a frame is being compressed, the earliest switching time is the timing of completely finishing compression. For instance, if a frame of length k cells arrives at the tth cell time, it will be compressed completely at the $(t+k)$th cell time.)

To solve these two critical problems, we propose a simple and efficient compression decision algorithm (CDA). According to the routing information of cells, the CDA will decide whether to perform the compression or not so that the link capacity will not be wasted and the link utilization is maximal.

The rest of paper is organized as follows. In Section II, the switch architecture is described. In Section III, the CDA is introduced. In Section IV, the saturated throughput of the proposed architecture is analyzed. In Section V, the simulation models and simulation results are reported. Finally, some conclusions are given in Section VI.

II. System Description

The architecture of smart photonic ATM switch with compression strategy is shown in Fig. 1. For simplicity, we only consider a switch with two input ports and two output ports (denoted as 2×2 switch). The proposed architecture can be easily extended to $N \times N$ switch. When a cell comes from an input link, it will go into the FP for detecting its destination. A number of consecutive cells will be grouped into a frame if they have a same destination port. The FP will put the start pulse (F_s) and the end pulse (F_e) in the beginning position and

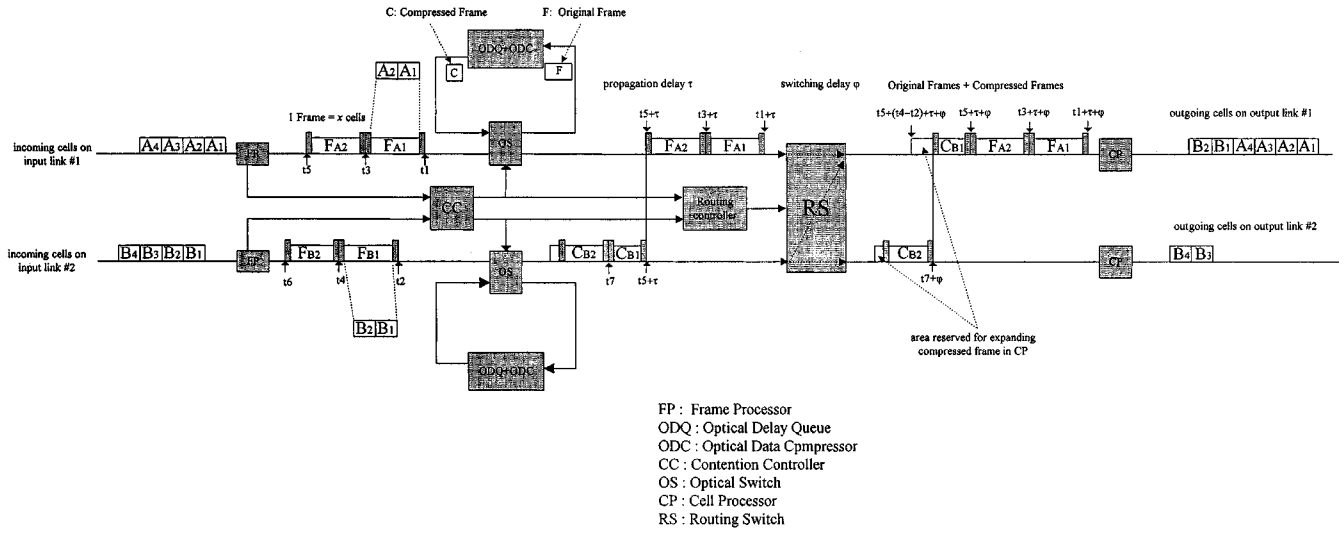


Fig. 1. Architecture of smart photonic ATM switch with compression strategy.

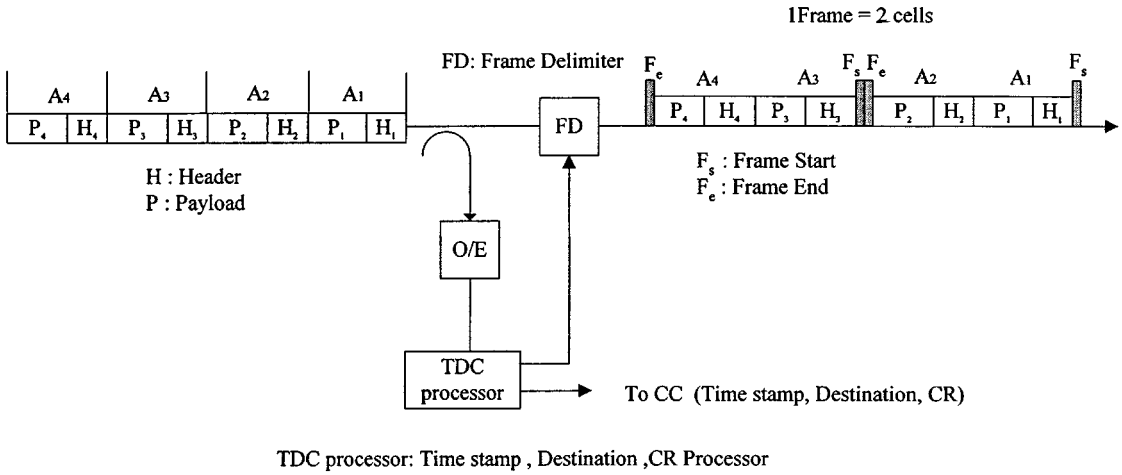


Fig. 2. Structure of Frame Processor (FP).

the ending position of a frame, respectively. In order to distinguish the start/end signals from data, the optical power level for these two pulses are higher than that of frame data. Fig. 1 shows an example of the proposed architecture. In Fig. 1, we assume the destination port of cells A1, A2, A3, A4, B1, and B2 is the upper output link (output link #1) and the destination port of cells B3 and B4 is the lower output link (output link #2). They arrive with a burst length of two cells. Therefore, every two consecutive cells are grouped into a frame by FP. Meanwhile, FP estimates the compression ratio (CR) of this frame and passes to the contention controller (CC). The CR is defined as the ratio of the length of compressed frame and the length of original frame (the compression method and CR estimation will be described later). If a frame cannot be switched immediately, it will be compressed and buffered. (For example, F_{B1} and F_{B2} in Fig. 1). The exact queueing delay of each queued frame is determined by the CC. The CC collects all information from the FPs and determines which frame can be switched out in the OS. That is, CC will check the timing information of frame to determine whether it collides with others. If collision is detected, the losers will be first compressed by the ODC and then put into the ODQ. On the contrary, this winner frame will be forwarded to the routing switch (RS) directly. Once a frame passes through OS, the RS will switch it to the correct destination port. Since the contention is solved by the CC in advance, no collision will occur in RS. The RS can be constructed by a simple electro-optics (EO) cross-bar switch which is controlled by the routing controller. When the RS switches a frame out, the CP will restore all original cells from the frame and remove the start/end pluses (including compressed frame or noncompressed frame) before they leave this switch. As shown in Fig. 1, we can find that the impact of HOL blocking can be reduced by the simple compression strategy.

A. Structure of Frame Processor (FP)

The detail structure of FP is shown in Fig. 2. The frame processor consists of fiber coupler, O/E converter, frame delimiter (FD) and a TDC (Timestamp, Destination, CR) processor. The fiber coupler and O/E converter extract the header and payload information from optical to electrical signals. Both incoming Virtual Path Indication (VPI) and Virtual Channel Indication (VCI) in cell header will be transferred into the new VPI and

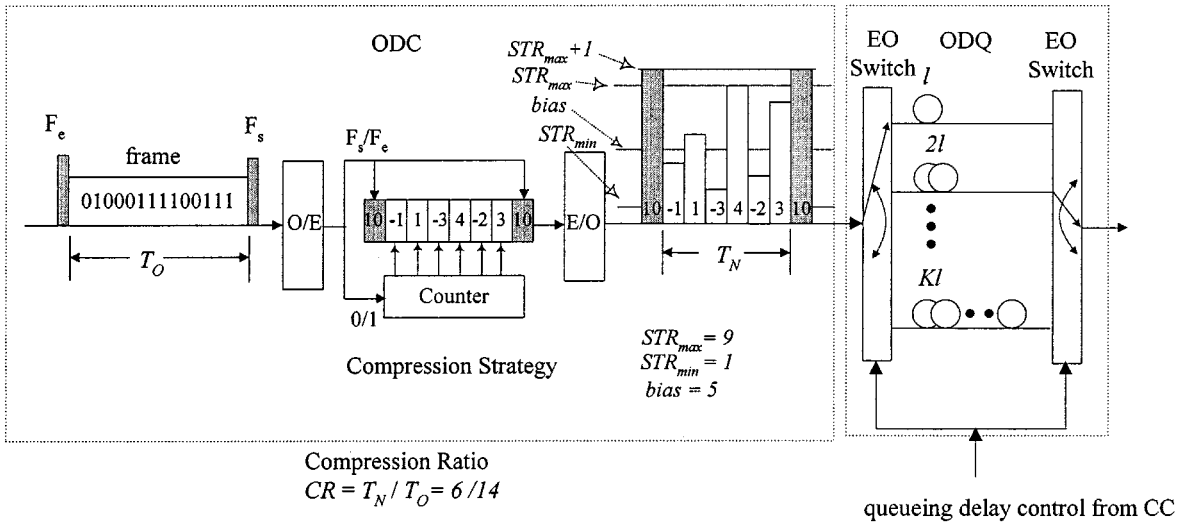


Fig. 3. Structure of ODQ + ODC in smart photonic ATM switch.

VCI for the next ATM switch according to its lookup/routing table. The TDC processor discriminates the timing, destination and CR of incoming cells. Cells with the same destination are grouped into a frame by the frame delimiter FD. While a frame is forwarded out, the TDC passes the corresponding information (time stamp, destination, CR) of this frame to CC.

B. Structures of Optical Data Compressor (ODC) and Optical Delay Queue (ODQ)

When contention occurs on two frames, the CC will choose the one coming early and put the other frame into the ODQ + ODC. (If two frames arrive simultaneously, the CC will select one of them randomly.) Meanwhile, the CC will also estimate the queueing delay for the queued frame. The structures of ODQ and ODC are shown in Fig. 3. There are two kinds of electrical signal levels at the output of the O/E converter. One is the high-level signal for the start and end pulses in the frame. The other is the frame data signal level. After receiving the high-level signal of the start pulse in the frame, the ODC will start to compress incoming data. The incoming data will be scanned and partitioned into several groups so that each group consists a maximal number of consecutive 1's or 0's bits. The weight of each group is assigned according to the number of bits (either 0's bits or 1's bits) within it. When all weights of groups are derived, the ODC will assign different signal power level for each weighted value. As soon as receiving the next high-level signal pulse (i.e., the end signal), the ODC will output the compressed frame with various optical pulse's levels together with the start/end pulses, as shown in Fig. 3.

The compression mechanism is shown in Fig. 4. There are four states in the compression mechanism: 1) IDLE state; 2) START state; 3) PLUS state; and 4) MINUS state. Initially, it states in the IDLE state. As it receives the frame start signal F_s, it transfers into START state. In this state, if it encounters bit 1(bit 0), it transfers into PLUS (MINUS) state. Meanwhile, the counter C is set to $bias + 1(bias - 1)$. The consecutive number of 1's and 0's will increase and decrease the counter, respectively. We note that in state PLUS (MINUS), it will transfer into the MINUS (PLUS) state once it receives bit 0. At the same time, it outputs the current value of counter and resets the counter as $bias - 1(bias + 1)$. In this method, a group contains 1's bits (0's bits) will generate a value larger (less) than the *bias*. Each output counting value is mapped into a predefined output power level of the optical pulse, as shown in Fig. 3. Based on the compression mechanism, the bit pattern of incoming frame (01 000 111 100 111) in Fig. 3 is partitioned into six groups: g_1, g_2, g_3, g_4, g_5, and g_6. We have $g_1 = \{0\}$, $g_2 = \{1\}$, $g_3 = \{000\}$, $g_4 = \{1111\}$, $g_5 = \{00\}$, and $g_6 = \{111\}$. Therefore, we obtain the CR of this frame is 6/14. Based on this compression strategy, the CR of frame can be easily preestimated by FP during the progress of framing (i.e., linear scanning).

So far, it is impractical to support numerous optical power levels in optical fabric. That is, the group size in the compression mechanism is limited within a reasonable range. Thus, two thresholds are particularly assigned. One is the $STR_{\max}$ and the other is $STR_{\min}$. The former and the latter are the maximum numbers of consecutive 1's and 0's bits in group, respectively. Once the value of counter is equal to one of thresholds, it will output the value and reset the counter (start a new group). And then, it enters the START state as usual. Notice that the start/end pulses should not ambiguous with the compressed data. The output power level for the start and end optical pulses is given by $STR_{\max} + 1$. Moreover, the bias threshold can be assigned as $(STR_{\max} + 1)/2$ for the sake of simplicity. Consider Fig. 3 for example again, if $STR_{\max} = 5$, the incoming frame (01 000 111 100 111) will be partitioned into nine groups: $g_1' = \{0\}$, $g_2' = \{1\}$, $g_3' = \{00\}$, $g_4' = \{0\}$, $g_5' = \{11\}$, $g_6' = \{11\}$, $g_7' = \{00\}$, $g_8' = \{11\}$, and $g_9' = \{1\}$. As a result, the CR of this frame becomes 9/14. It implies that a larger $STR_{\max}$ is provided, a better CR will be obtained.

The ODQ can be constructed by employing scalable optical buffer architectures (e.g., simple loop buffers [13], phase sensitive amplifier fiber buffer [14], and recirculating fiber optical loop buffers [15], etc.). The advantages of these buffers are

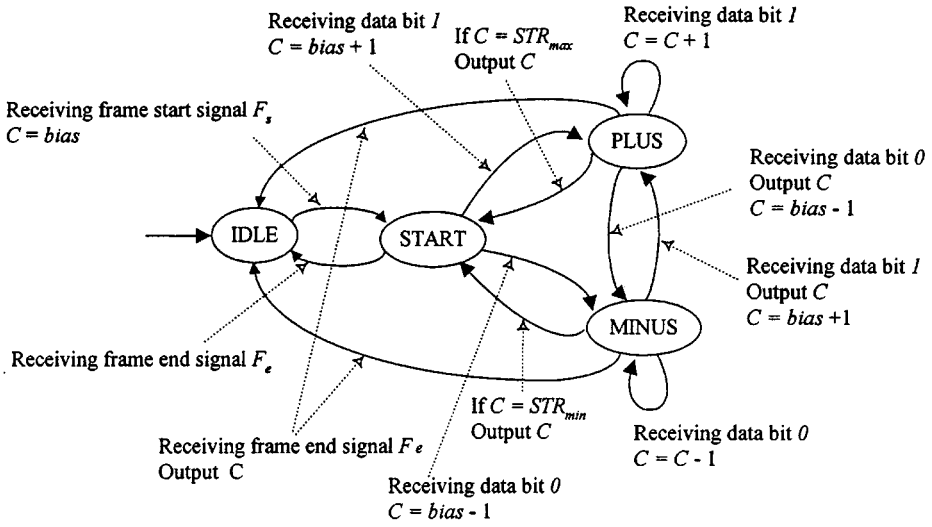


Fig. 4. State transition diagram of compression mechanism.

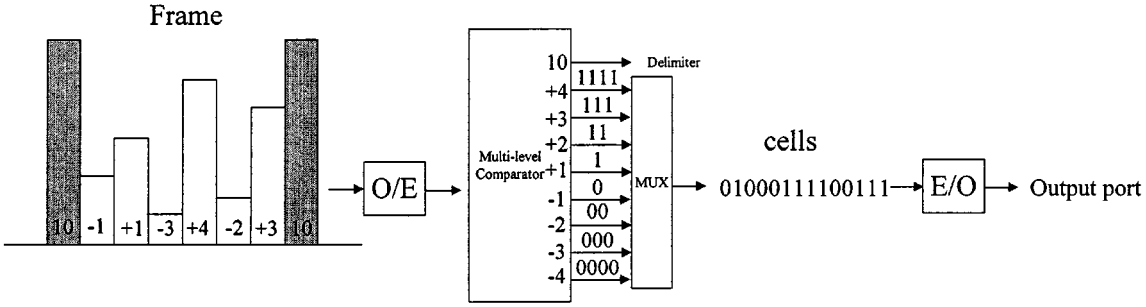


Fig. 5. Structure of Cell Processor (CP).

the scalability and low-loss characteristic. For the sake of simplicity, the proposed ODQ is composed by simple parallel optical delay line array. The selection of an appropriate delay line for a queued frame is done by the CC. The definition of unit delay line length (l) is a slot time, and an ATM cell may occupy many time slots. We note that this architecture may incur a significant amount of loss due to limited buffer space. Hence, the proposed architecture is suitable to general photonic ATM switch which does not require tremendous buffer size.

C. Structures of Cell Processor (CP)

The structure of CP is shown in Fig. 5. When a frame reaches the CP, the O/E converter will detect the high-level threshold of the start and end pulses in this frame. Once the start pulse is found, the CP starts to transfer the different electrical pulse's levels into the number of the consecutive 1's or 0's signals for the E/O converter to recover the original data stream. This can be done by employing a multi-level comparator. After detecting the high-level electrical threshold signal of the end pulse in the frame, the CP stops the transformation process. As a result, the outgoing cells on output link will be the same as the incoming cells at the input port of the smart photonic ATM switch, as shown in Fig. 1. To prevent from the overlapping, an enough expansion time should follow each compressed frame. The time interval allocated for expanding is the difference between the lengths of a frame before and after compression.

We note that if the CP equips with the ODQ, it is not necessary for CC to reserve the expansion time for each compressed frame. Each compressed frame will be queued in CP until all precedent frames are decomposed in complete. Such architecture is referred as output queueing discipline. Intuitively, this increases both implementation complexity and cost. In this paper, we only consider the case of input queueing-based ATM switch architecture.

III. Resolving Destination Contention

When a cell arrives at a switch, it will be routed according to the routing information with it. In ATM-based networks, cells are independently switched to the output port by observing the fields of VPI and VCI in its header. It is clear that two cells came from different input ports are contenting with each other if they destine to a same output link. The simple contention resolution is to buffer one of them and to forward it later. When the queue becomes full, the following incoming cells are dropped accordingly. In this case, the queued frames should be compressed as small as possible to accept more incoming cells. Nevertheless, in the case of light loaded, the queued frames should not be compressed to save the decompression time spends in the CP. As a result, it is desirable to design a smart CDA for proposed architecture.

A. Compression Decision Algorithm (CDA)

Let $F^k = \{f_0^k, f_1^k, \cdots, f_n^k\}$ denote the set of incoming frames on link k. Without losing generality, let frame f_i^k arrive

earlier than frame f_j^k if $i < j$. Before describing the CDA, we first define some useful terminologies:

- LAT_j latest available time on output link j;
- A_i^k arrival time of the ith frame on link k where $A_i^k < A_j^k$ for all $i < j$;
- D_i^k destination port of frame f_i^k;
- O_i^k original frame length of frame f_i^k;
- N_i^k frame length of frame f_i^k after compressing; (i.e., $N_i^k = O_i^k \times \mathrm{CR}$);
- S_i^k assigned switching time for frame f_i^k;
- Q_i^k assigned queueing delay for frame f_i^k.

Each time a cell arrives, the FP checks the cells' destinations and groups a number of consecutive cells with same destination into a frame. Each frame will be investigated by the proposed CDA to determine whether to perform compression or not. Since the switching fabric in the architecture does not allow collision, the exact switching time should be scheduled in this phase. In other words, the queueing delay of frame is decided right now. If no contention is detected, the frame will be switched immediately. On the contrary, once contention occurs, the queueing delay of frame has to be estimated precisely and the compression process will be triggered when they reached ODQ + ODC. For the sake of simplicity, we ignore the signal propagation delay from FP to RS in the following algorithm.

We now formally address the algorithm of considering an incoming frame f_i^k:

```
Compression Decision Algorithm (CDA)
1) Let j = D_i^k;
2) If (LAT_j ≤ A_i^k) { // no contention with
  other frames on different input links
3)     If ((S_{i-1}^k + N_{i-1}^k) ≤ A_i^k) { // precedent
  frame f_{i-1}^k was switched already
4)         S_i^k = A_i^k; // without compression
5)         Q_i^k = 0;
6)         LAT_j = S_i^k + O_i^k;
7)    } Else { // delay by frame f_{i-1}^k to
  maintain the sequence order
8)           S_i^k = S_{i-1}^k + N_{i-1}^k; // perform com-
  pression
9)           Q_i^k = S_i^k - A_i^k;
10)          LAT_j = S_i^k + N_i^k;
11)       }
12)   } Else { // delay for switching
  earlier arrived frames
13)     If ((S_{i-1}^k + N_{i-1}^k) ≤ LAT_j) { // delay is
  not caused by its precedent frame f_{i-1}^k
14)         S_i^k = LAT_j; // perform compres-
  sion
15)         Q_i^k = S_i^k - A_i^k;
16)         LAT_j = S_i^k + N_i^k;
17)   } Else { // delay by its precedent
  frame f_{i-1}^k
18)           S_i^k = S_{i-1}^k + N_{i-1}^k; // perform
  compression
19)           Q_i^k = S_i^k - A_i^k;
20)           LAT_j = S_i^k + N_i^k;
21)           }
22)  }
```

B. An Example of Performance Improvement

Intuitively, frames are compressed before switching will shorten the switching time. A shorter switching time will decrease the degree of contention/congestion. In order to evaluate the performance of the proposed architecture with CDA algorithm, we investigate two measurements: the total queueing delay (TQD) and total switching time (TST). The queueing delay of a frame is the time interval between its forwarding timing and its arrival time. Given a stream of incoming cells, the TST is the total time period for switching out all cells completely. A small TQD implies a high system throughput. A small TST implies that the switch allow the queued frames have a chance to be switched out earlier. Therefore, a good switching architecture should obtain a smaller TQD and a smaller TST. Consider Fig. 6 for example. Fig. 6(a) and (b) show the traditional switching method without using any scheduling algorithm and the smart switching method with compression mechanism, respectively. In Fig. 6(a), we can find that the TQD is 10 cell times. In this case, the obtained TQD in Fig. 6(b) is only 8 cell times. Moreover, the TSTs in the traditional switching method and the smart switching method are 10 cell times and 9 cell times, respectively.

We note that even though the compression strategy will shorten the switching time, the switch performance may not be increased significantly in some particular cases. Fig. 7(a) shows the case that there are two compressed frames on each inlet: frames $\mathrm{F_{A1}}$($\mathrm{F_{A2}}$) and $\mathrm{F_{B1}}$(F_{B2}) are destining to outlet $\alpha(\beta)$. In Fig. 7(b), even frames $\mathrm{F_{B1}}$ and $\mathrm{F_{B2}}$ have different destinations, $\mathrm{F_{B2}}$ must be delayed until time $O_{\mathrm{A1}} + N_{\mathrm{B1}}$. This is because that the architecture inhibits the overlapping occurs on the path from OS to RS. This will waste a time period of $(O_{A1} + N_{B1}) - (O_{A2} + N_{A1})$. Such wasting will not happen if $(O_{A2} + N_{A1}) \geq (O_{A1} + N_{B1})$, as shown in Fig. 7(c).

IV. Saturated Throughput Analysis

To model the behavior of the proposed ATM switch architecture with input queue and optical compressor, we assume the cell arrival process at each inlet is based on independent and identical Bernoulli processes [7]. Each cell is directed to one of the N outputs with probability $1/N$. The arrival cells are stored in an input queue which acts as a FIFO. If only a single cell of the possible N first cells of the N input queues is destined for one outlet it will be selected. If j cells ($j > 1$) are destined for a particular outlet, the selection of winner is performed at random, each cell to be selected with a probability of $1/j$. The blocked cells must wait for the following selection process in the next cell time.

To analyze the behavior of the input queues, we assume that all input queues are saturated. This is the case when the input queue always has a packet waiting. Let B_m^i be the number of cells at the heads of the input queues that are blocked at the end of the mth cell time and are destined for outlet i that are not selected by the CC during the mth cell time. During the mth cell time, a number of cells will be served and transported to their respective outlet, making room for a number of new cells head of the line cells. We suppose that A_m^i denotes the number of new cells appear at the head of the queue for output i during the mth

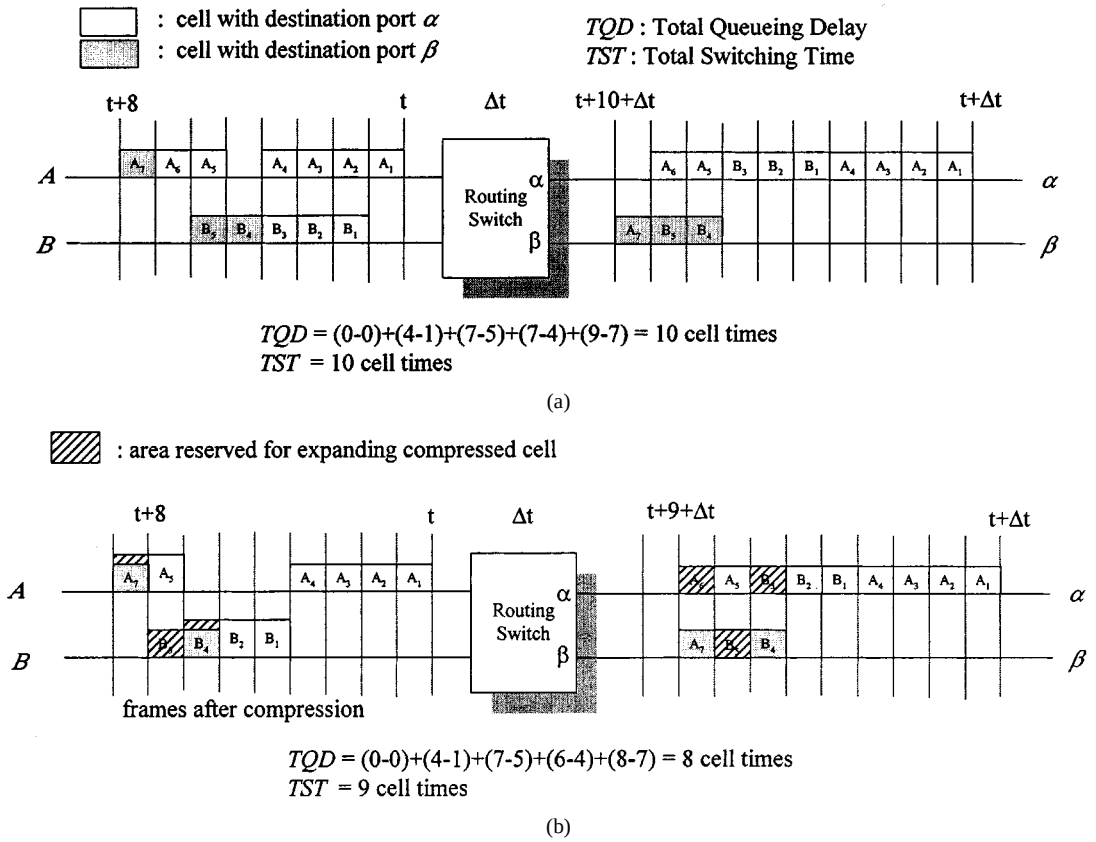


Fig. 6. Example of smart switching method. (a) Traditional switching method. (b) Smart switching method.

cell time. Since only one cell can be served by outlet i during a cell time, it follows that at every new cell time the number of blocked cells at the HOL equals the number of blocked cells during the previous cell time B^i_{m-1} minus 1 (already served) plus the new arriving cells A^i_m, i.e.,

$$B^i_m = \max\left(B^i_{m-1} - 1 + A^i_m, 0\right).$$

In every input queue, new cells are progressing to the head of the line. Let L_m denote the total number of input queues with new cells at their heads during the mth cell time. Therefore, we have

$$L_m = \sum_{i=1}^{N} A^i_m.$$

Consider the case of input saturated, the number of new cells at input queues in the mth cell time is equal to the number of serviced cells in the $(m-1)$th cell time. Thus, we have

$$L_m = N - \sum_{i=1}^{N} B^i_{m-1}.$$

Let $\overline{L}$ and ρ denote the average of L_m and ρ is the utilization of the output lines or the switch throughput in steady state situation, respectively. We can easily obtain the following equation:

$$\overline{L} = N \times \rho.$$

To obtain analytic formula which are not too complicated and result in a good approximation, we assume the reserved expansion area of each compressed frame could successfully overlap with other frames (or compressed frames) without wasting switching capacity. Let r is the average data CR in system. If $N \to \infty$, the equilibrium number A^i of cells destined for output i and moving to the head of the input queues in each cell time becomes Poisson Distribution with average arrival rate $r \times \rho$. This means that if the switching throughput is considered as 1, the normalized average arrival rate will become $r \times p$. This also implies that the behavior for $N \to \infty$ of the mean steady state value of B^i is like that of an M/D/1 system. That is,

$$\overline{B^i} = \frac{(r\rho)^2}{2 \times (1 - r\rho)}.$$

If we average equation $L_m = N - \sum_{i=1}^{N} B^i_{m-1}$, we have

$$\overline{L} = N - \sum_{i=1}^{N} \overline{B^i}$$

or

$$\sum_{i=1}^{N} \overline{B^i} = N - \overline{L} = N - N \times \rho = N \times (1 - \rho).$$

As $N \to \infty$, we have $\overline{B^i} = 1 - \rho$. By solving the following equation

$$1 - \rho = \frac{(r\rho)^2}{2 \times (1 - r\rho)}$$

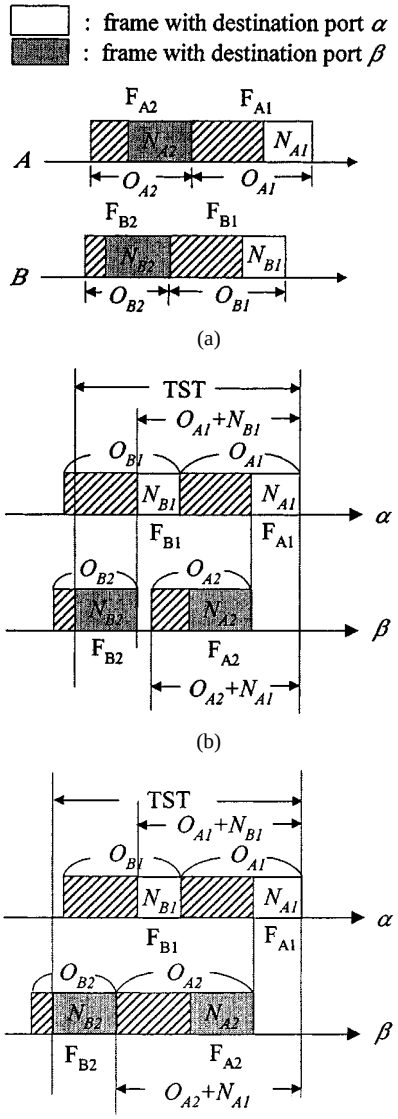


Fig. 7. Performance analysis of proposed CDA. (a) Incoming frames before queueing and compressing. (b) TST = $O_{A1}+N_{B1}+N_{B2}$ when $O_{A2}+N_{A1} < O_{A1}+N_{B1}$. (c) TST = $O_{A2}+N_{A1}+N_{B2}$ when $O_{A2}+N_{A1} \geq O_{A1}+N_{B1}$

we obtain the upper bound on the throughput of proposed switch architecture with average CR r under $N \to \infty$ is

$$\rho = \frac{(r+1) - \sqrt{3r^2 - 2r + 1}}{r \times (2 - r)}.$$

Table I illustrates the analyzed saturated throughput as a function of r when $N \to \infty$. To provide a more accurate observation, a simulation for the saturated throughput is implemented. In the simulation, we consider the case of input saturated. This is the case when the input queue always has a cell waiting. The generated source-destination distribution is uniform distribution. That is, each cell is directed to one of the N outputs with probability $1/N$. Since the network throughput is affected by the network size N, different network sizes are investigated. The simulation results and the estimated saturated throughput under

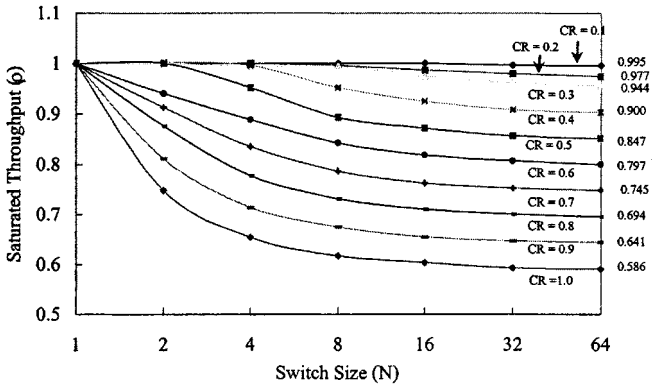


Fig. 8. The obtained throughput of proposed switch architecture strategy under different switch sizes (N) and compression ratios (CR) when $\Lambda = 1.0$.

TABLE I
THE SATURATED THROUGHPUT UNDER DIFFERENT COMPRESSION RATIOS

Compression Ratio r	Saturated Throughput ρ
0.1	0.995
0.2	0.976
0.3	0.944
0.4	0.899
0.5	0.845
0.6	0.788
0.7	0.731
0.8	0.678
0.9	0.630
1.0	0.586

different CR and different network sizes are shown in Fig. 8. We can see that as the network size becomes larger, a lower network throughput will be derived. For higher level of CR, we obtain a better network throughput gain. We note that the demonstrated results are obtained under the case without allocating expansion area for compressed data. In other words, the expanding area can overlap with the following incoming cell entirely. Therefore, the obtained result can be referred as the upper bound throughput of proposed CDA. The obtained throughput when $N = 64$ is very close to the analytic throughput, as shown in Table I.

V. SIMULATION MODEL AND SIMULATION RESULTS

A. Simulation Models

To investigate the performance of proposed architecture and algorithm, three simulation models are investigated. For each simulation run, the average queueing delay (AQD) of incoming cells are collected and analyzed. The AQD is measured as the average time interval between a packet arrives and the timing of switching it. In these simulation models, we consider an $N \times N$ smart photonic ATM switch. The packet arrival rate of input port i is a Poisson distribution with a mean λ_i, and the packet length (frame size) is an Exponential distribution with a mean of L cells. Therefore, the input link load (ILL) of i can be calculated as follows:

$$\text{ILL}_i = \lambda_i \times L.$$

Thus, the normalized total network load (denoted as Λ) can be easily obtained by the following equation:

$$\Lambda = \left(\sum_{i=1}^{N} \lambda_i \times L \right) \Big/ N.$$

Without losing generality, the source-destination distribution is considered as a uniform distribution. In the first simulation, we investigate how the AQD of the CDA is affected under different numbers of CR and different network loaded (Λ). The CR is considered from 0.1 increases to 1.0 by a step of 0.1. The other assumptions are listed as follows:

1) $N = 2$ input/output ports in ATM switch;
2) $L = 50$ cells;
3) $\Lambda = \{0.1$ (light load), 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0 (heavy load)$\}$;
4) CR $= \{0.1$ (well compressed), 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0 (without compression)$\}$.

In the second simulation model, two kinds of network parameters are considered: the switch size (N) and the network load (Λ) under a specified CR. The assumptions are given as follows:

1) $N = \{2, 4, 8, 16, 32\}$ input/output ports in ATM switch;
2) $L = 50$ cells;
3) $\Lambda = \{0.1$ (light load), 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0 (heavy load)$\}$;
4) CR $= 0.5$.

In the last simulation model, we investigate how the AQD of the CDA is affected under different frame sizes (L) and different network loaded under a specified CR. The other assumptions are listed as follows:

1) $N = \{2, 4, 16\}$ input/output ports in ATM switch;
2) $L = \{10, 50, 100\}$ cells;
3) $\Lambda = \{0.1$ (light load), 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0 (heavy load)$\}$;
4) CR $= 0.8$.

B. Simulation Results

Fig. 9 shows the AQD obtained by the proposed strategy under different CRs and network load when $N = 2$ and $L = 50$. The results obtained by the traditional switching method is the case for CR $= 1.0$ (without compression). We can see that a smaller CR is given, a lower AQD will be obtained. We note that the obtained AQD without compression is very high and the AQD will become infinite when only providing 55% traffic load ($\Lambda = 0.55$) into the switch. However, in the case of CR $= 0.6$, we can find that the infinite queueing delay occurs when network load is $0.92(\approx 0.55/0.6)$. This implies that if all incoming data can be compressed in a ratio of 60%, the system throughput will be improved significantly. We also can see that the obtained queueing delay is bounded within a small value when CR ≤ 0.5. This is because under such condition, the incoming traffic load will be reduced as CR $*\Lambda$. For example, the saturated throughput of CR $= 0.1$ becomes $5.5(=0.55/0.1)$. As a result, the infinite queueing delay will occur when network load is Λ/CR in our proposed architecture. This remarkable improvement is apparently caused by the concept of employing compression strategy.

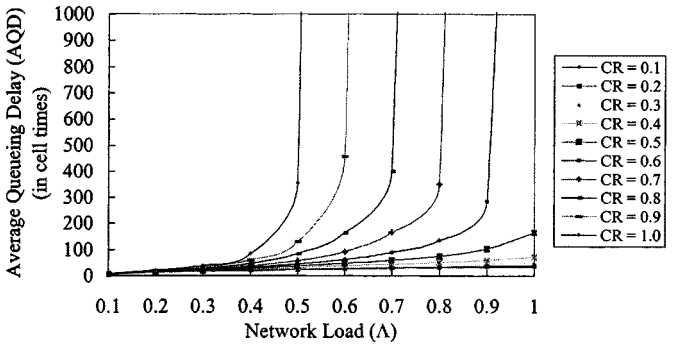


Fig. 9. Simlation results obtained by the CDA under different network loads (Λ) and CR.

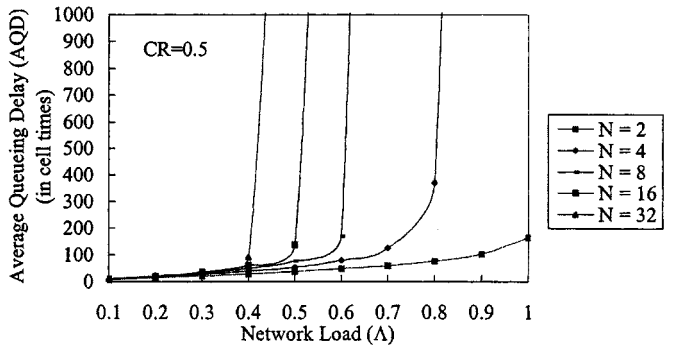


Fig. 10. The AQDs obtained by the CDA under different switch sizes (N) and network load (Λ) when CR $= 0.5$.

Since the CR is extremely depending on the bit pattern of incoming data stream. It is possible that all incoming data cannot be compressed at all (i.e., CR $= 1.0$). In this worse case, the proposed compression strategy is useless.

In Fig. 10, it shows that the relation between AQD and CR under different switch sizes N and different network loads Λ. Intuitively, a lager size of switch has a chance to occur seriously destination contention (under a same normalized network loaded). As a result, most of incoming cells will be queued for a longer time and a higher AQD will be obtained. However, if the CR is good and the frequency of compression is high, it will release the congestion situation and obtain an acceptable AQD. In Fig. 10, we can find that the affordable traffic loaded is less than the analytical throughput listed in Table I. This is because that the reserved expansion area for each compressed frame is considered in this simulation. Recall that reserving the expansion area during scheduling/switching will efficiently reduce both cost and complexity of CP.

Fig. 11 demonstrates the AQDs obtained by proposed strategy under different switch sizes, different network loads and different frame sizes $L = 10, 50, 100$ cells when CR $= 0.8$. It is cleat that if a larger frame is queued, it will cause the later incoming frames to be queued also. Therefore, the ADQ will be increased significantly. Comparing the results shown in Fig. 11, we can see that the AQDs obtained are almost identical when the network loaded is saturated. This is because that almost all frames are queued in spite of how large a frame size is. However, the obtained AQDs under different frame sizes in light network

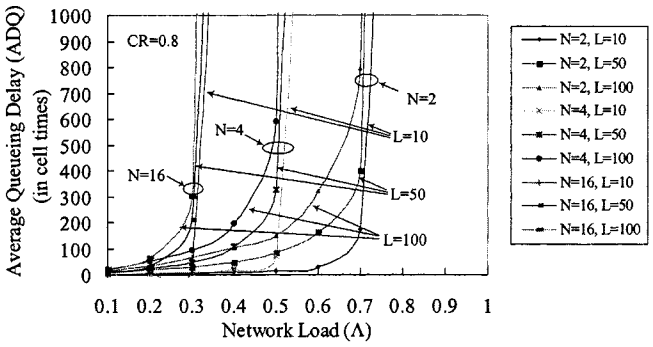


Fig. 11. Comparion of obtained AQDs under different switch sizes (N), different frame sizes (L) and network load (Λ) when CR $= 0.8$.

load is obviously different from each other. We can see that a smaller frame size is, a lower AQD will be derived.

From these simulation results, we can say that the proposed switch architecture with compression strategy will work well even when the CR is 0.8 only. This makes a possibility to build an inexpensive and high switching performance photonic ATM switch for backbone networks.

VI. Conclusion

In this paper, we introduced a novel photonic ATM switch architecture with compression strategy to release the degree of congestion. Because such congestion is mostly caused by the destination contention, the proposed architecture will compress the loser frames which consists of a number of consecutive cells with the same output port. Therefore, it not only shortens the required switching time but also releases the degree of congestion. In this paper, the timing of employing compression strategy was analyzed and the performance of the proposed CDA was analyzed and investigated by simulation. Simulation results show that the proposed strategy is better than the traditional approach under general traffic demands and different switch sizes.

References

[1] CCITT, "B-ISDN ATM cell transfer performance," *IEEE Commun. Mag.*, vol. 30, no. 4, Apr. 1992.

[2] J.-Y. L. Boudex, "The aynchronous transfer mode: A tutorial," *Computer Networks and ISDN Systems*, vol. 24, pp. 279–309, 1992.

[3] W. T. Chen, H. J. Liu, and Y. T. Tsai, "High-throughput cell scheduling for broadband switching systems," *IEEE J. Select. Areas Commun.*, vol. 9, no. 9, pp. 1510–1523, Dec. 1991.

[4] Y. Choi *et al.*, "Performance study of the multi-cell transfer scheme in the optical backbone network," *IEICE Trans. Commun.*, vol. E78-B, no. 5, pp. 729–735, 1995.

[5] Y. Choi *et al.*, "Photonic ATM switch architecture for WDM optical networks," *IEICE Trans. Commun.*, vol. E78-B, no. 9, pp. 1333–1335, 1995.

[6] J. W. Lockwood *et al.*, "Scalable optoelectronic ATM networks: The IPOINT fully functional testbed," *J. Lightwave Technol.*, vol. 13, no. 6, pp. 1093–1103, 1995.

[7] M. Prycker, *Asynchronous Transfer Mode: Solution for Broadband ISDN*. London, U.K.: Ellis Horwood, 1993.

[8] Y. Shimazu *et al.*, "Ultrafast photonic ATM switch with optical output buffers," *J. Lightwave Technol.*, vol. 10, no. 2, pp. 265–272, 1992.

[9] W. D. Zhong *et al.*, "Terahipas: A modular and expandable terabit/second hierachically multiplexing photonic ATM switch architecture," *J. Lightwave Technol.*, vol. 12, no. 7, pp. 1307–1315, 1992.

[10] W. D. Zhong *et al.*, "Optical bit-interleave multiplexing ATM switch," *Electron. Lett.*, vol. 31, no. 12, pp. 995–997, 1995.

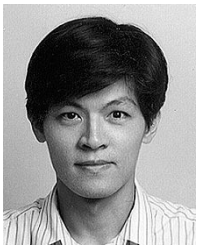
[11] K. L. Deng *et al.*, "Optical packet compressor for ultra-fast packet-switched optical networks," *Electron. Lett.*, vol. 33, no. 14, pp. 1237–1239, July 1997.

[12] P. Toliver *et al.*, "Simulaneous optical compression and decompression of 1000-Gb/s OTDM packets using a single bidirectional optical delay line lattice," *IEEE Photon. Technol. Lett.*, vol. 11, no. 9, pp. 1183–1185, Sept. 1999.

[13] K. L. Hall and K. A. Rauschenbach, "All-optical buffering of 40-Gb/s data packets," *IEEE Photon. Technol. Lett.*, vol. 10, no. 3, pp. 442–444, Mar. 1998.

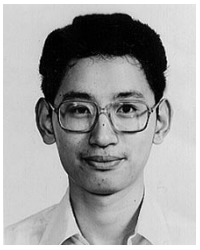
[14] Y. Yamada *et al.*, "Demonstration of 30 circulations in a transparent optical-loop buffer for 2-channel FDM packets at a data rate of 2.5 Gb/s," in *Tech Dig. OFC'96*, vol. WD2, San Jose, CA, 1996, pp. 107–108.

[15] G. D. Bartolini *et al.*, "All-optical storage of a picosecond-pluse packet using parametric amplification," *IEEE Photon. Technol. Lett.*, vol. 9, pp. 1020–1022, July 1997.



Shiann-Tsong Sheu was born in Taiwan in 1968. He received the B.S. degree in applied mathematics from National Chung Hsing University, Taiwan, R.O.C., and the Ph.D. degree in computer science from National Tsing Hua University, Taiwan, R.O.C., in 1990 and 1995, respectively.

Since 1995, he has been as Associate Professor in the Department of Electrical Engineering at Tamkang University, Taiwan, R.O.C. His current research interests include ATM networks, WDM networks, personal communication networks, and design and analysis of protocols and algorithms.



Yang-Han Lee was born in Taipei, Taiwan, R.O.C., in 1964. He received the B.S., M.S., and Ph.D. degrees in electrical engineering from National Taiwan University, Taipei, Taiwan, in 1987, 1989, and 1992, respectively.

Since 1994, he has been an Associate Professor in the Department of Electrical Engineeering at TamKang University, Taipei, Taiwan. His research interests are in the area of integrated circuits and systems for communication, optical fiber communication systems, and wireless communications systems.

Chih-Chiang Wu received the B.S. degree in electronica engineering from Feng Chia University, Taiwan, R.O.C., in 1997.

He is currently working toward the Ph.D. degree in electrical enegineering at Tamkang University, Taiwan, R.O.C. His main research interests are ATM networks, mobile networks, and network bandwidth management systems.