Flow Control in a Multi-Plane Multi-Stage Buffered Packet Switch

H. Jonathan Chao
Department of Electrical and Computer Engineering
Polytechnic University, Brooklyn, NY 11201
chao@poly.edu

Jinsoo Park
Division of Engineering Technologies and
Computer Sciences
Essex County College, Newark, NJ 07102
jpark@essex.edu

Abstract—A large-capacity, multi-plane, multi-stage buffered packet switch, called the TrueWay switch [1], was previously proposed by us. It can achieve hundreds of terabits/sec capacity. A small-scale TrueWay switch was prototyped to demonstrate the concept and feasibility. Three different load balancing schemes were investigated to achieve high throughput and low average delay with a moderate speedup. In this paper, we focus on the study of one of the load balancing schemes, window-based re-sequencing scheme, without a speedup. It is the most promising one among the three in terms of performance. Buffered switch modules are used in different stages to eliminate the need of centralized scheduling. However, packet out-of-subsequence is inevitable as packets are distributed to different paths that have various queuing delays. By applying flow control between the input and output ports and limiting the re-sequencing window size (similar to TCP/IP flow control), we are able to keep the implementation cost to an acceptable level while still providing high throughput. Link-level flow control between the switch stages is required to prevent the downstream queues from being overflowed. The interaction between link flow control at switch stages and end-to-end flow control at switch ports is an interesting problem. We show by simulations that the TrueWay switch can be engineered to achieve high throughput without an internal speedup even under bursty non-uniform traffic distributions.

Index Terms—packet switch, multi-plane switch, Clos network

I. INTRODUCTION

Multi-plane multi-stage buffered switch architectures have been used to build a large-capacity switch in the range of hundreds terabits per second [1][2][3]. Multiple planes are used to accommodate higher line rates, e.g., 40 Gbps. As the data rate of electronic switch device I/O is at range of a few Gbps (e.g., 2.5 Gbps), multiple switch planes are used to achieve an aggregated rate of, for instance, 80 Gbps, providing a moderate internal speedup for higher performance and accommodating the overhead carried in the cell header for routing purpose and the waste of unfilled cell payload resulted from packet segmentation. Multiple switch planes also provide better fault tolerance.

Multi-stage networks, such as Clos [4] or Beneš network, are used to interconnect switch modules to build a large-scale switch fabric. Each input-output pair has multiple paths to provide load balancing for better performance. A multi-path switch fabric can be further classified into a memory-less switch fabric or a buffered switch fabric. A memory-less multi-path switch fabric has no packet out-of-sequence problem because the propagation delays through different paths are deterministic.

However, it requires resolving internal links and output port contention by a centralized packet schedule with global information available, thus limiting switch fabric’s scalability. A buffered multi-path switch fabric does not need centralized packet scheduling, but distributed/local scheduling because the buffer in each switch module can temporarily store cells due to internal link or output port contention. However, it could cause a buffer overflow in the switch fabric and cell out-of-sequence.

Buffer overflow in the buffered multi-path switch can be prevented by using link-level flow control between stages. A downstream queue will periodically inform the upstream queues about its queue status. To maximize link-level throughput, each downstream queue size should be able to store cells worth of at least round trip time (RTT) between the switch stages. In a large scale router, racks that hold the line cards and switch fabric can be separated in a distance as far as a couple of hundred meters, or several time slots, pressing the constraint of the on-chip memory. With today’s advanced CMOS technology, it is feasible to have 16Mbits on-chip memory, making the buffered multi-stage switch feasible. Cells from the same flow are distributed to multiple paths to achieve better load balancing and thus higher performance. However, they may experience different delays while traversing different paths and cause out of sequences. This problem can be resolved by using a re-sequencing buffer to re-order the cells at the output port. To limit the degree of out of sequence a window-based port-to-port flow control scheme (similar to TCP/IP flow control) can be used. We show that with a moderate re-sequencing buffer high throughput is achieved.

In this paper, we study the performance of the TrueWay switch in details, especially the interaction between the link flow control at switch modules and end-to-end flow control at switch ports. We study the impact of several design parameters to the switch's performance, such as the crosspoint queue size (q), re-sequencing window size (w), link-level credit update period (CUP), and port-level acknowledgment period (AP).

The remainder of this paper is organized as follows. The TrueWay switch architecture is described in Section II. Stage-to-stage and port-to-port flow control is presented in Sections III and IV, respectively. Section V analyzes the switch performance and Section VI shows the simulation results. Section VII gives the conclusion.
II. TrueWay Switch Architecture

Figure 1 shows the TrueWay switch architecture. The ingress traffic manager (TMI) receives packets from an external line, segments the packets into fixed-size cells, and distributes cells to different paths of different switch planes. The egress traffic manager (TME) collects cells from different planes and reassembles them to packets and sends the packets to the external line.

![Figure 1 TrueWay switch architecture](image)

The switch fabric consists of $p$ switch planes, and each has a three-stage Clos network. The modules in the first, second, and third stages are denoted as input modules (IMs), center modules (CMs), and output modules (OMs). Each module can be logically considered a cross-point buffered switch. The first stage of the switch plane consists of $k$ IMs, each with $n$ inputs and $m$ outputs. The second stage consists of $m$ CMs, each with $k$ inputs and $k$ outputs. The third stage consists of $k$ OMs, each with $m$ inputs and $n$ outputs.

A. TMI/TME

Each TMI has two types of queues, Virtual Output Queue (VOQ) and Virtual Path Queue (VPQ). As shown in Figure 2, when a packet arrives at a TMI, it is put into the corresponding VOQ according to their destined TME. Each cell is assumed to be 64 bytes with a payload of 52 bytes. The head-of-line (HOL) cells of the VOQs are moved to the VPQs in a round-robin fashion. Each VPQ corresponds to a routing path determined by a switch plane and a CM in that switch plane. Each output link of the TMI selects a VPQ and sends its HOL cell to the IM in round robin.

Figure 3 illustrates the TME structure. When cells arrive at the TME, they are put into a corresponding Virtual Input Queue (VIQ) corresponding to the TME. Each cell carries a sequence number (SN), which ranges from 0 to $w-1$. If the cells in the VIQ are in order, they are sent to the Reassembly Queue (RAQ). Once the RAQ has a complete packet, the pointers of the linked cells are moved from the RAQ to the Class of Service Queue (COSQ). Since more than one RAQ may have a complete packet, packets can be served in a round-robin fashion to achieve fairness. The COSQ chooses a packet for transmission based on a service policy, such as Strict Priority (SP), Weighted Round Robin (WRR), or others.

![Figure 2 Ingress Traffic Manager (TMI) structure](image)

The TME periodically generates an SN Acknowledgment (ACK) and sends it to the corresponding TMI through the switch fabric. When an ACK is generated, it is stored in the ACK buffer in the TME until it is sent. It will be described in more detail in Section IV.

![Figure 3 Egress Traffic Manager (TME) structure](image)

B. Switch Modules

The term Switch Modules (SMs) refers to the IM, CM, or OM. Here, we assume $n=m=k$ to simplify the discussion. Each module has $n^2$ crosspoints and each crosspoint has a logical queue. Therefore, each SM has $n^2$ crosspoint queues. Figure 4 shows an example of 9 queues of a 3x3 switch module. Each queue can receive and send at most one cell in each time slot when there is no internal speedup. An incoming cell is stored at one of the queues before being routed to the next stage.

By sharing the space of the $n$ queues on each input line (e.g., 3 horizontal queues in the CM of Figure 4), a higher throughput can be achieved. However, the input-shared memory needs to operate at $(n+1)$ times the line speed. With an assumption of 16Mbit on-chip memory for a $64\times64$ SM, each queue can hold 8 cells. Some queue lengths can grow beyond 8 cells when they are more occupied and other queues are less.

III. STAGE-TO-STAGE FLOW CONTROL

It is a common practice to have a large buffer to hold packets/cells at line cards and a small buffer inside the switch fabric. This is because the latter usually operates at a much higher speed than the former and thus is normally implemented on the switching devices. For the multi-stage buffered switch, a
flow control mechanism is used between the stages to prevent the downstream buffers from being overflowed. Two flow control schemes, Back-Pressure and Credit-Based Flow Control [5], are considered here. We adopted the latter one with modification and call it the DQ scheme.

To simplify the system design, we adopt the same cell scheduling policy for each output link of the TMI, IM, CM, and OM. Figure 4 shows that an output link of IM(1) is responsible for scheduling cells in the 3 source queues (SQs in vertical) to the 3 destination queues (DQs in horizontal) of CM(2). A cell is transferred from an SQ at the upstream switch module to a DQ at the downstream switch module only if the SQ has credits. Cells in the SQs are sent to the DQs in round robin. DQs periodically send credits (CRT) to the SQs as cells have been sent to the next stage and spaces are free up.

Each SQ maintains three counters: Queue Outstanding Cell Counter (QOC), RTT Outstanding Counter (ROC), and Buffer Outstanding Counter (BOC). The QOC keeps track of the number of cells in the DQ, and those on the link destined for the DQ. The ROC keeps track of the number of cells transmitted during the RTT just before the CRT is received. The BOC is the sum of all outstanding cells destined for the same input link buffer at the downstream. A crosspoint queue size (q) is defined to be the maximum number of cells that a DQ can hold. It is an engineering parameter that will affect the performance of the TrueWay switch. Initially QOC is set to 0. Whenever a cell is transmitted, the QOC is increased by one. As long as the value of the QOC is less than q, the SQ is able to forward cells to the DQ. Once the value of QOC reaches q, the SQ stop sending cells until it receives a CRT.

The DQ periodically sends a CRT to the SQs, which contains the number of cells remaining in the DQ. Then the SQ can update QOC as the sum of the CRT and ROC (i.e., QOC = CRT + ROC). The summation counts the cells in transition while the CRT is delivered. When the QOC is updated, the ROC is reset to 0. From the SQ's point of view, the eligibility of sending cells depends on the value of the QOC.

Figure 4 shows an example of the DQ scheme. Cells in the 3 SQs of IM(1) are scheduled to transmit to the 3 DQs of CM(2). The output link of IM(1) that is connected to CM(2) keeps track of the QOC and ROC for each of the 3 DQs. A HOL cell of an SQ is defined to be eligible if its destined DQ has a free space. The output link serves the eligible cells in round robin. If we assume that q is 15 and the buffer size of the 3 DQs is 64 cells, a cell is eligible for transmission when the QOC is smaller than 15 and the BOC is smaller than 64.

IV. PORT-TO-PORT FLOW CONTROL

When building a packet switch, it is a common practice to segment each arriving packet into multiple fixed-length cells, route them through the switch fabric, and reassemble them into packets with the reassembly queues at the output ports. Packets that are routed through different paths may experience different queuing delays and become out of sequence. We use window-based re-sequencing scheme to tackle the packet out-of-sequence issue in the multi-plane, multi-stage buffered switch. A re-sequencing buffer at the TME is used to re-sequence the cells before delivering them to the external line.

A flow is defined as a TMI-TME pair. More specifically, all sessions/connections arriving at the same TMI and destined for the same TME are considered to be the same flow. If there are N input ports and N output ports, the number of flows is N².

The window-based re-sequencing uses a sequence number (SN) for each flow. The SN ranges from 0 to w-1, where w is the window size. Therefore, the number of outstanding cells in the switch fabric for a flow cannot exceed w.

With reference to Figure 5, when an input port sends a cell to the switch fabric, it attaches an SN to the cell header. The SN is given by TMI. The TMI maintains two pointers for each VOQ, the sequence head pointer (SEQ_head) and the sequence tail pointer (SEQ_tail). The SEQ_head is the SN to be attached to the cell sent from the input port. Right after a cell is sent, the SEQ_head is increased by one.
The ACK packets are delivered to the corresponding TMI through the same link used for cell delivery and CRT delivery. Figure 6 shows how they are delivered to their destinations. For instance, CRT from OM to CM is delivered through IM. There is a link between OM and IM to pass the CRT. CRT from CM to IM goes through OM. CRT from IM to TMI goes through OM and TME. Note that there is no flow control between OM and TME, thus no CRT from TME to OM. ACK originated from a TME goes through its associated TMI, the switch fabric (IM, CM, and OM), to destination TME, and then TMI. Each IM, CM, and OM has a separate buffer to store the ACK packet.

![Figure 6 How flow control signals are transferred in the switch](image)

### V. PERFORMANCE ANALYSIS

The TrueWay switch can achieve high throughput under both uniform and circuit-switched traffic. The two main design parameters that affect the throughput of the TrueWay switch are the queue size \( q \) and the window size \( w \). The queue size must be greater than the sum of RTT stages, where RTT stage is the propagation delay between switch stages, and link-level acknowledgment period (AP) and the product of RTT port and the number of planes, where RTT port is the propagation delay between TMI and TME and does not include the queuing delay in the switch fabric. The window size must also be smaller than the product of the queue size and the number of planes divided. To summarize:

(i) \( q > \text{RTT\_stage} + \text{CUP} \)

(ii) \( w > \alpha \times \text{RTT\_port} \times p + \text{AP} \)

(iii) \( w < \beta \times p \times q \)

where based on our performance study \( \alpha \) is a constant between 1.1 and 1.2 and \( \beta \) is a constant between 4.2 and 4.3.

#### A. HOL Blocking

Under uniform traffic distribution, the TrueWay switch can suffer throughput degradation due to HOL blocking. HOL blocking exists because of the FIFO structure of the SQ. It contains cells destined for different DQs and its HOL cell may be stopped from transmission due to the lack of credits. If we had a parallel queue structure for each SQ, we would have been able to eliminate HOL blocking.

As shown in Table 1, as each OM has \( n^1 \) queues, each CM should have \( n^2 \) queues to eliminate HOL blocking. Likewise, if each CM has \( n^3 \) queues, each IM should have \( n^4 \) queues to eliminate HOL blocking. However, to keep the number of queues to a manageable level, the TrueWay switch uses only one FIFO queue at each crosspoint and the total number of queues of each switch module is \( n^5 \), resulting in HOL blocking at the IM and CM.

<table>
<thead>
<tr>
<th>No. of required queues to avoid HOL blocking</th>
<th>IM</th>
<th>CM</th>
<th>OM</th>
</tr>
</thead>
<tbody>
<tr>
<td>n^4</td>
<td>n^3</td>
<td>n^2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>No. of queues in the TrueWay switch</th>
</tr>
</thead>
<tbody>
<tr>
<td>n^3</td>
</tr>
</tbody>
</table>

Table 1 Number of required queues to avoid HOL blocking

There is no HOL blocking at the OM because all cells destined for the output link are destined for the same TME. Therefore, its output link will never be idle whenever there is at least one cell destined for the TME (i.e., work-conserving). But OM can be congested when more than one cell arrives at the OM in the same time slot are destined for the same TME.

At the CM, there could be HOL blocking. The output link of the CM can be idle if the corresponding DQs of all HOL cells are full while the DQs of those cells behind of the HOL cells leave free spaces.

At the IM, HOL blocking may also occur in the same manner as at the CM. The DQs at a CM can be filled when many cells destined for the same OM are routed to the same CM.

At the TMI, HOL cells at VOQ can be held back in two cases. One is due to the congestion of the DQ at the IM. The other is due to flow control between the input and output ports. The larger the window size, the more outstanding cells in the switch fabric.

The HOL blocking probability depends on the queue size \( q \), the number of planes \( p \), and the window size \( w \). The larger the queue size, the more sharing effect, and thus the smaller the HOL blocking probability. The more switch planes, the larger the memory of the switch fabric, and thus the smaller the HOL blocking probability. As the window size is sufficiently large to keep traffic routing, the larger the window size is, the more cells in the switch fabric, and thus the larger the HOL probability. The ratio of the memory space in the switch fabric \( x \) to the total possible number of outstanding cells \( y \) shows the resource availability of a cell in the switch fabric. \( x \) is the sum of all memory space in IM, CM, and OM, which is \( 3p^4b \). Where \( b \) is the buffer size. If we allow one queue to occupy up to 50% of the buffer size (i.e., \( b = 2q \)), \( y \) is \( N^w \), or \( N^w \). The ratio is \( 6pq^Nw \). Therefore, the resource of each cell, which reflects to the overall throughput, increases as the number of planes or the queue size increases, and decreases as the window size increases.

Figure 7 shows an example of HOL blocking. At TMI(0), there are four cells A, B, C, and D, destined for TME(0), TME(1), TME(2), and TME(3), respectively. At TMI(1), cells W, X, Y, and Z are destined for each TME. There are two planes, plane(0) and plane(1). Cells are sent in round robin but it is possible that one path is more congested than the other. Cell W at IM(00) is held back because its DQ at CM(00) is filled with cells A and B. Cell Y at IM(00) is blocked due to its HOL cell W, although its DQ is not filled.

Authorized licensed use limited to: Polytechnic Inst of New York Univ. Downloaded on October 10, 2008 at 15:42 from IEEE Xplore. Restrictions apply.
Figure 7 An example of cell flows under uniform traffic

B. Cell Out-Of-Sequence

The window size is the same as the VIQ size. Let the window size be equal to w. Then each TMI/TME pair can send up to w cells before it receives an ACK packet. That is, the outstanding number of cells between an input and an output port must be equal to or less than w. Because we allow cells belonging to the same flow to take different paths in the switch fabric, the window size determines the degree of out-of-sequence.

Under circuit-switched traffic, the TrueWay switch achieves 100% throughput if the window size exceeds a certain threshold. The CM will never be congested because the input and the output links of each connection are determined at call setup and the cell arrives in one-time slot is guaranteed to be sent in the next-time slot. IM and OM can be congested because more than one cell destined for the same output link may arrive in the same-time slot.

Figure 8 An example of cell flows under circuit-switched traffic

Figure 8 shows an example how cells are routed through the switch fabric for circuit traffic. At TMI(0), there are four cells A, B, C, and D, destined for TME(0). At TMI(1), cells W, X, Y, and Z are destined for TME(1). Cells A and W arrive at TMI(00) in the same time slot. Therefore, one of the two cells must wait until the other cell is transferred to CM(00), which causes a queuing delay at CM. At CM(00), at most one queue for an output link can have a valid cell and all other queues for the output link will be empty because the traffic is circuit-switched. Therefore, there is no queuing delay at CM for circuit-switched traffic. At OM(00), cells A and C arrive in the same time slot, which causes a congestion at OM. Now the question is how large the window size (w) must be in order to achieve high throughput. The answer would be that w must be greater than the minimum window size in Table 2.

The RTT_port is composed of the propagation delay between the input and the output ports and the queuing delay in the switch fabric. The propagation delay is fixed, including the propagation delays between the TMI and IM, IM and CM, CM and OM, and OM and TME. For example, if the link speed is 2.5 Gbps and the cell size is 64 bytes (i.e., 512 bits), each time slot (TS) is 204.8 nsec. If the distance between IM and CM is 100 m, the propagation delay is 500 nsec (i.e., 100 m/2x10^8 m/s), which is equivalent to 3 TS. Let us assume that the RTT stage between IM and OM is 6 TS and that between TMI and IM is 2 TS. Then the RTT_port is 16 TS.

As shown in Table 2, the minimum window size to achieve full throughput is proportional to the number of switch planes. Since we assumed that the round-trip propagation delay is 16 TS, the window size of 4 planes must be greater than 64 (i.e., 16^4). The AP and the queuing delay in the switch fabric should also be considered when determining the window size.

Table 2 Minimum window size to achieve 100% throughput for circuit-switched traffic with different number of planes

<table>
<thead>
<tr>
<th>w</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>26</td>
<td>44</td>
<td>64</td>
<td>80</td>
<td>102</td>
<td>120</td>
<td>140</td>
<td>156</td>
<td></td>
</tr>
</tbody>
</table>

VI. COMPUTER SIMULATIONS

This section presents the TrueWay switch's throughput by extensive computer simulations. The system parameters used in computer simulations are as follows. There are 64 TMs. Each TM receives up to 4 cells per TS and sends up to 4 cells per TS, which is equivalent to an internal speedup of one. There are four planes and the IM, CM, and OM have 8 input links and 8 output links.

The RTT stage between the TM and the IM/OM is assumed to be 2 TS. The RTT stage between the IM/OM and CM is assumed to be 6 TS. The CUP is 8 TS. The VOQ size is 8,191 cells and the total memory size at each TME is 524,224 cells. The VIQ size is 256 cells and the total memory size at the TME is 16,384 cells. The queue size is 1.5 cells and the buffer size is 64 cells. The total memory size at each IM, CM, and OM is 512 cells.

The OM sends CRT signals to the CM every 8 TS. The CM sends CRT signals to the IM every 8 TS. The IM sends CRT signals to the TMI every 8 TS. However, the TME does not send the CRT signals to the OM because the VIQ never overflows due to the port-to-port flow control. Instead, the TME creates ACK packets in every 8 TS. It is assumed that all TMs have the same traffic model. The simulations assume a single priority level and only unicasting traffic. Each simulation generated about 20 million cells.

We considered two packet size distributions. The first distribution is random traffic with all packets having a size of one cell (1=1). The second is bursty traffic with a geometrically distribution between 1 and 192 cells with an average packet size of 10 cells (1=10).
The destinations of incoming packets are independent of each other. Therefore, it is possible that a large portion of the traffic is destined for the same TME at the same time. The destination distribution is determined by a so-called non-uniform parameter \( u \). For uniform traffic \( (u = 0.0) \), the destination is distributed evenly over all the TMEs. For circuit-switched traffic \( (u = 1.0) \), all cells at one TME are destined for one TME. In between these extremes, \( u 	imes 100\% \) of the traffic is assigned for one TME and the remaining \( (1-u) 	imes 100\% \) of the traffic is uniformly distributed over all TMEs.

A. Queue size

The TrueWay switch achieves high throughput (>94%) when the queue size is greater than the sum of RTT stage and the CUP. For instance, if the RTT stage is 6 TS and the CUP is 8 TS, the queue size must be greater than 14 cells. Figure 9 shows that the maximum throughput of TrueWay switch exceeds 94% if the queue size is greater than 14 under various traffic patterns including bursty non-uniform traffic.

![Figure 9 Maximum throughput with different queue sizes (w=256, p=4)](image)

B. Window size\( w \)

From the simulation results, we observe that the TrueWay switch achieves high throughput when \( w \) is greater than a certain threshold. For example, if the number of planes is equal to 4, the minimum window size to achieve full throughput for circuit-switched traffic is 80, as shown in Table 2. In Figure 10, we observe that the Trueway switch has a poor throughput when the window size is less 80.

![Figure 10 Maximum throughput with different window sizes (q=15, p=4)](image)

One interesting observation is for the bursty uniform traffic, the throughput slightly decreases when the window size increases. This is because the DQs become more congested as more outstanding cells exist in the switch module due to a larger window size. As a result, the HOL blocking probability increases, which in turn degrades the throughput.

C. Number of switch planes

As the number of switch planes increases, the total memory in the switch fabric proportionally increases because we assume that each SM has the same memory size (e.g., 16 Mbits). For one plane, the DQ is more likely to be congested due to its smaller memory space. As expected, the throughput significantly degrades with smaller number of planes for bursty traffic, as shown in Figure 11.

![Figure 11 Maximum throughput with different numbers of switch planes (w=256, q=15)](image)

VII. CONCLUSION

The Trueway switch is a multi-plane multi-path buffered switch. It is highly scalable because packet scheduling is distributed to each individual crosspoint buffered switch module. With today’s CMOS technology, each crosspoint can have several cells in average. Crosspoint queues of the same input link share their space to achieve high throughput.

The TrueWay switch achieves high throughput without internal speedup when it meets two requirements. The first is that the crosspoint queue size is greater than the sum of the round trip time (RTT) between stages and the credit update period. The second is that the window size must be greater than the sum of product of the number of planes and RTT between ports, and the acknowledgment period. Two factors that affect the performance of the Trueway switch are HOL blocking and cell out of sequence. Under uniform traffic, HOL blocking adversely affects the throughput more than cell out of sequence as the switch fabric becomes more congested.

REFERENCES


