A QoS Provisioned CIOQ Packet Switch Using Crossbar Structure with \( m \) Internal Links

Carlos Roberto dos Santos  
DEE-INATEL, Sta Rita Sapucai MG, Brazil  
Phone: +55-35-3471-9221  
Fax: +55-35-3471-9314  
carlos@inatel.br

Shusaburo Motoyama  
DT-FEEC-UNICAMP, Campinas SP, Brazil  
Phone: +55-19-3788-3765  
Fax: +55-19-3289-1935  
motoyama@dt.fee.unicamp.br

Abstract.

A QoS provisioned CIOQ switch using crossbar structure with \( m \) parallel lines per output port is proposed in this paper. The packets at input buffers are transferred to the output buffers by means of \( m \times N \) internal lines. Since all internal lines have the same speed as external links, no internal clock speedup is required so that the proposed structure is suited for high-speed switches. The switch models for analysis are proposed and its performances are evaluated by means of queuing theory. The results show that only 2 internal links for each output port are sufficient for quick packet transfer from the input buffers to the output buffers. The proposed switch has also the feature that facilitates the choice of scheduler in order to satisfy the QoS of each class of service.

1. Introduction

The intense increasing in Internet traffic is demanding high capacity packet switches. One of the approaches to obtain these switches is combining the switching speed of layer 2 with the routing features of layer 3. Such kind of switch is called IP switching. Many IP switching structures have been proposed and its performance analyzed in the literature [1] – [5]. In [1], [3] and [4], the used switch structures are based on a combination of ATM layer 2 with IP routing. In these switches the incoming packets are segmented and switched as ATM cells. At outputs the cells are reassembled as packets again and transmitted to another node. In structures proposed in [2] and [5] a label is added to a packet for switching purpose. This label has switching function of layer 2 and a packet is switched without segmentation. In both structures IP routing is used.

Recently the switch structures based on virtual output queuing (VOQ) and Combined Input-Output Queuing (CIOQ) have received much attention [6] – [8]. VOQ is a queuing scheme used to solve the HOL blocking problem associated with FIFO input queuing or to provide facility to satisfy quality of service (QoS) in input queuing. In this technique each input port maintains a virtual separate queue for each output port. CIOQ is a queuing scheme that combines input and output queuing. It is a good compromise between the performance and scalability of both output-queued and input-queued switches.

The structure presented in [6], uses a combination of VOQ and CIOQ. At input port \( N^2 \) buffers are provided and a two-stage approach for scheduling purpose is implemented. The arbiters at the input side perform input contention resolution, while the output-buffered switch element performs classical output contention resolution. The two main bottlenecks in this switch structure are the shared memory interconnection complexity (each input and each output must be able to reach each memory location) and the output queue bandwidth (up to \( N+1 \) accesses per queue per packet cycle).

The necessity to speedup internal links to match input ports to output ports in CIOQ switches is discussed in [7]. In [8] the internal link speedup based on space division multiplexing is proposed and analyzed.

In this paper we propose a simple practical CIOQ switch based on a crossbar structure. The proposed crossbar structure has only one line at each input port but each output port has \( m \) parallel lines. All internal lines have the same speed as external links. Thus, there is no need to speedup internal clock. This simple provisioning permits to the switch a parallel transfer of packets, thus giving a very high throughput at the output port.

In [9], [10] and [11] we have proposed and analyzed CIOQ switch based on crossbar structure with \( m \) internal links but the analysis was done using fixed size ATM cells. In this paper we propose a CIOQ switch using crossbar structure and dealing with any packet size.
The paper is organized as follows. In section 2, the switch structure is proposed. It is discussed two types of structures: one using FIFO scheduling and the other using priority scheduling. In the following section, the switch models for analysis are proposed, the switch performances are analyzed using queuing theory and the results are discussed. Finally, the conclusions are presented in the last section.

2. Proposed Structure

2.1. FIFO Scheduling Structure

Figure 2.1 shows the proposed structure using FIFO scheduling. Each input port has a single buffer, whereas at each output port a set of \( m \) buffers is provided.

![Figure 2.1. Crossbar switch with single input buffer and \( m \) output buffers](image1)

Each input buffer is connected to the \( m \) output buffers through the \( m \) physical links. A control unit (CTR) at each input buffer and a scheduler (SCH) at each output are also provided for control purposes. Each Control Unit (CTR) is connected to each Scheduler (SCH) through the separate reserved lines in which the request (REQ) information is transported. The request information is sent from a CTR to the correspondent scheduler when a new packet arrives at an input buffer. The scheduler after receiving the REQs from CTRs uses a scheduling algorithm to choose until \( m \) packets and acknowledges until \( m \) CTRs to transmit packets to \( m \) output buffers through the separate reserved lines (ACK lines). Another type of information (End Of Transmission - EOT) is also provided so that a CTR can signal the SCH that a packet has been transmitted and is waiting for another acknowledgement for transmission. The EOT information can be sent using the same lines used to transmit REQ information.

As an example of switch operation let suppose that inputs ports \( i_1, i_2, i_3, i_4 \) and \( i_5 \) have at its buffers packets addressed to the same output port address \( j \) at the same time. Let also suppose that the number of internal links is three (\( m=3 \)). By using for example the round robin scheduling scheme, after these inputs have sent the REQ signals to the scheduler \( j \), only the inputs \( i_1, i_2 \) and \( i_3 \) will receive the ACK signals. Simultaneously, the scheduler \( j \) will choose internal links 1, 2 and 3, and will enable the respective gates at crosspoints. Soon after one of inputs \( i_1, i_2, i_3 \) ends the packet transmission and sends EOT information to SCH, the input \( i_4 \) receives ACK signal and will be served, using a free internal link. Figure 2.2 shows the basic operation described above. The scheduling algorithm used at output buffer can be independent of SCH, but combining both schedulers it is possible to use complex and flexible scheduling algorithm in order to guarantee QoS of each service.

![Figure 2.2. Basic operation of the proposed structure](image2)

2.2. Priority Scheduling Structure

Figure 2.3 shows the switch structure using priority scheduling as an example for satisfying QoS. Other type of scheduling algorithm could be used such as WFQ but for simplicity the priority scheduling was chosen.

The figure shows that \( r \) FIFO buffers for each flow class are provided at each input port. Two sets of REQ and ACK lines, connecting each input port control
(CTR) to each scheduler (SCH) to transport request and acknowledge information are also provided. Each output needs \( m \times N \) buffers, where \( m \) is the number of internal links. It is important to note that each internal link needs a physical buffer, but flow class separation can be done by a logical buffer (shared memory). The figure shows that a priority scheduler (SCH) for each output port is also provided.

Figure 2.3. QoS provisioned packet switch structure

The incoming packets in each input port are discriminated and stored to the appropriate flow class queues. The flow class \( f_1 \) has highest priority followed by \( f_2 \) and so on, whereas \( f_r \) is the lowest flow class priority. This discrimination information is also sent to SCH using the REQ line by CTR. Thus, a SCH has all information to choose the packet to transmit in a priority order. If there are more than \( m \) requests, each SCH selects \( m \) packets on a priority order and sends ACK signals to the chosen CRTs through the reserved lines (ACK bus in Figure 2.3), so that a CRT can transmit a packet to the output buffer. At each output port, the scheduler transmits the packets in priority order.

As an example of operation, let suppose that input ports \( i_1, i_2, i_3, i_4 \) and \( i_5 \) have in its buffers, HOL packets with flow classes \( f_4, f_1, f_2, f_3 \) and \( f_1 \), respectively, addressed to the same output port \( j \). Also, let suppose that the number of internal links is three (\( m=3 \)). By priority criteria the SCH of output port \( j \) will choose the input ports \( i_2, i_3 \) and \( i_5 \) and will send the ACK signals to them. This operation is shown in Figure 2.4.

Figure 2.4. Basic operation of QoS provisioned packet switch

3. Performance Analysis

3.1. FIFO Switch Model for Performance Analysis

Since the crossbar structure is symmetric, for switch modeling, only a generic output \( j \) can be considered as shown in Figure 2.2. The input queues are distributed queues for the output port \( j \) and it can be modeled as only one queue and \( m \) parallel lines are \( m \) servers. At output port the set of \( m \) queues can also be considered as only one queue. Thus, the whole switch model can be considered as open network of queues as is shown in Figure 3.1.

Figure 3.1. Switch model for performance analysis

3.1.1. Queuing Analysis For analysis the following assumptions are made. The number of input or output links is \( N \), the packet arrival in each input link obeys Poisson distribution with average arrival rate \( \lambda \) and the packet length is exponentially distributed with mean \( 1/\mu \). The probability of a packet at input port being routed to a particular output port is \( 1/N \). The packets are served in a FIFO (first in first out) basis. With the above assumptions the open network of queues of Figure 3.1 can be modeled as an M/M/m queue at input
and as an M/M/1 queue at output. By using Jackson theorem for open network of queues, the input queue can be considered independent of output queue. The output queue can also be considered independent of input queue.

For M/M/m queuing system, the calculation of average queue size $E(N_q)$ and average packet waiting time in queue $E(T_q)$ are well-known and are given by:

$$E(N_q) = \frac{\rho^{m+1}}{m!(1-\rho/m)^2} p_0$$  \hspace{1cm} (3.1)

where $\rho = \frac{\lambda}{\mu}$ is the load per input or output link and

$$p_0 = \left[ \sum_{n=0}^{m-1} \frac{\rho^n}{n!} + \frac{m \rho^m}{m!(m-\rho)} \right]^{-1}

The average packet waiting time in queue $E(T_q)$ can be calculated by using Little’s law.

$$E(T_q) = \frac{E(N_q)}{\lambda}$$ \hspace{1cm} (3.2)

The average number of packets in system, $E(N)$, and average packet waiting time in system (queue + service), $E(T)$, are given by:

$$E[N] = \rho + E[N_q] = \rho + \frac{\rho^{m+1}}{m!(1-\rho/m)^2} p_0$$

and

$$E[T] = \frac{E[N]}{\lambda} = \frac{1}{\mu} + E(T_q)$$ \hspace{1cm} (3.3)

Table 3.1 Average number of packets in the input buffer as a function of traffic load

<table>
<thead>
<tr>
<th>$\rho$</th>
<th>$m=1$</th>
<th>$m=2$</th>
<th>$m=3$</th>
<th>$m=4$</th>
<th>$m=5$</th>
<th>$m=6$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\rho=1$</td>
<td>1.1111e-02</td>
<td>2.5063e-04</td>
<td>5.3795e-06</td>
<td>9.9149e-08</td>
<td>1.5702e-09</td>
<td>2.1661e-11</td>
</tr>
<tr>
<td>$\rho=2$</td>
<td>5.0000e-02</td>
<td>2.0202e-03</td>
<td>8.3542e-05</td>
<td>3.0239e-06</td>
<td>9.4761e-08</td>
<td>2.5961e-09</td>
</tr>
<tr>
<td>$\rho=3$</td>
<td>1.2857e-01</td>
<td>6.9054e-03</td>
<td>4.1152e-04</td>
<td>2.1916e-05</td>
<td>1.0187e-06</td>
<td>4.1556e-08</td>
</tr>
<tr>
<td>$\rho=4$</td>
<td>2.6667e-01</td>
<td>1.6667e-02</td>
<td>1.2688e-03</td>
<td>8.8271e-05</td>
<td>5.4065e-06</td>
<td>2.9184e-07</td>
</tr>
<tr>
<td>$\rho=5$</td>
<td>5.0000e-01</td>
<td>3.3333e-02</td>
<td>3.0303e-03</td>
<td>2.5786e-04</td>
<td>1.9500e-05</td>
<td>1.3054e-06</td>
</tr>
<tr>
<td>$\rho=6$</td>
<td>9.0000e-01</td>
<td>5.9341e-02</td>
<td>6.1644e-03</td>
<td>6.1520e-04</td>
<td>5.5107e-05</td>
<td>4.3905e-06</td>
</tr>
<tr>
<td>$\rho=7$</td>
<td>1.6333e+00</td>
<td>9.7721e-02</td>
<td>1.1237e-02</td>
<td>1.2770e-03</td>
<td>1.3165e-04</td>
<td>1.2132e-05</td>
</tr>
<tr>
<td>$\rho=8$</td>
<td>3.2000e+00</td>
<td>1.5238e-01</td>
<td>1.8921e-02</td>
<td>2.3952e-03</td>
<td>2.7821e-04</td>
<td>2.9041e-05</td>
</tr>
<tr>
<td>$\rho=9$</td>
<td>6.1111e+00</td>
<td>2.2853e-01</td>
<td>3.0012e-02</td>
<td>4.1000e-03</td>
<td>5.3551e-04</td>
<td>6.2303e-05</td>
</tr>
</tbody>
</table>

Numerical Example

The average number of packets in an input queue as function of the number of internal links $m$, for several values of traffic load $\rho$ is shown in Table 3.1. As it can be observed, for $m\geq 2$ there is no practically packet waiting in the input buffer. Even under heavy load ($\rho=.9$) and $m=2$, the number of packet in input queue is small, about .23 packets in average. For traffic load less than .9 and $m=2$, the number of packet in input buffer is less than one in average. The result shown in the first column ($m=1$) in Table 3.1 is also the average number of packet waiting in output buffer.

In order to calculate the average packet waiting time, it assumed that the link capacity is 100 Mbits per second and average packet length is 8000 bits. Figure 3.2 shows the results obtained by using Equations 3.1 and 3.2. As it can be seen the average waiting time is very small for $m\geq 2$, even under heavy load ($\rho=.9$). In the worst case, for $m=2$ and $\rho=.9$, the average waiting time is about 10 $\mu$seconds and 720 $\mu$seconds for $m=1$. For M/M/1 queuing system, that is a particular case of M/M/m queuing system with $m=1$, the expressions for average number of packets in queue, $E(N_q)$, and average packet waiting time in queue, $E(T_q)$, are given by:

$$E(N_q) = \frac{\rho^2}{1-\rho}$$ \hspace{1cm} and \hspace{1cm} (3.4)

$$E(T_q) = \frac{E(N_q)}{\lambda}$$ \hspace{1cm} (3.5)

The average number of packets $E(N)$, and average packet waiting time $E(T)$ in the system can be calculated by:

$$E[N] = \frac{\rho}{1-\rho}$$ \hspace{1cm} and \hspace{1cm} (3.6)

$$E[T] = \frac{E[N]}{\lambda} = \frac{1}{\mu-\lambda} = \frac{1}{\mu} + E(T_q)$$ \hspace{1cm} (3.7)
In the case of input buffer, for traffic load equal to 0.9, the average waiting time is very small; it is varying from 10 µseconds for link capacity equal to 100 Mbps to 0.4 µseconds for link capacity equal to 2.4 Gbps. In output buffer the average waiting time is varying from 700 µseconds for 100 Mbps capacity to 30 µseconds for 2.4 Gbps capacity. The average time that packets spend in the switch, that is, the average time interval between packet arrival and departure, can be calculated as the sum of average waiting time spent in the input and output queuing and average service time spent in input and output servers. Considering that in the proposed structure the input and output servers have the same average service time, the main impact in average packet waiting time is given by output system. Figure 3.5 shows the total average waiting time spent by a packet in the proposed structure for \( m=2 \) internal links, and considering average packet length of 8000 bits and link capacity of 622 Mbps. Even under heavy traffic (\( \rho=0.9 \)) the total average waiting time is still small, about 0.14 mseconds.

In the case of input buffer, for traffic load equal to 0.9, the average waiting time is very small; it is varying from 10 µseconds for link capacity equal to 100 Mbps to 0.4 µseconds for link capacity equal to 2.4 Gbps. In output buffer the average waiting time is varying from 700 µseconds for 100 Mbps capacity to 30 µseconds for 2.4 Gbps capacity. The average time that packets spend in the switch, that is, the average time interval between packet arrival and departure, can be calculated as the sum of average waiting time spent in the input and output queuing and average service time spent in input and output servers. Considering that in the proposed structure the input and output servers have the same average service time, the main impact in average packet waiting time is given by output system. Figure 3.5 shows the total average waiting time spent by a packet in the proposed structure for \( m=2 \) internal links, and considering average packet length of 8000 bits and link capacity of 622 Mbps. Even under heavy traffic (\( \rho=0.9 \)) the total average waiting time is still small, about 0.14 mseconds.

3.2. Priority Switch Model for Performance Analysis

The model considers that five input buffers are provided for each flow class. In each flow class buffer the packets arrive from \( N \) inputs at a rate \( \lambda_i \), assuming that the packets arrival at the inputs obeys Poisson Distribution. To study the performance analysis of proposed switch in relation to packet delay at input and output buffers, since the switch structure is symmetric, the analysis model can be simplified for only one queue for each flow class and \( m \times 5 \) queues can be also considerate as only one queue for each flow class, as is shown in Figure 3.6. The server \( j \) looks first for a packet to transmit in the \( f_i \) buffer that is the highest

---

**Figure 3.2.** Average packet waiting time as function of traffic load

**Figure 3.3.** Average waiting time in output queuing

**Figure 3.4.** Average waiting time in input queuing

**Figure 3.5.** Total average waiting time spent in the switch
priority flow and can serve until m packets simultaneously. If there is no packet or there are less than m packets in f1 buffer the server goes to the next buffer (f2, f3, f4, and f5, in this sequence) until completes m packets. The output buffer at each of internal links is buffer (f

The model also considers that the number of input or output port links is N and the traffic percentage of service class i is S_i (ΣS_i = 1). The probability that a packet in an input port will send to the particular output port is 1/N. Then the rate at each flow class buffer is N.S_i.ρ_i/1/N.

The output buffer at each of internal links is divided into five flows classes and the packets are transmitted to the output j in a priority order. The packets of same priority are served in a FIFO (first in first out) basis. Then, the open network of output. It is also assumed that there are r different flows and a generic flow of priority p can assume p=1,2,3,...r, where r is lowest priority.

For priority queuing system non preemptive service discipline, the calculation of average packet waiting time E(T_p) is given by

\[ E(T_p) = \frac{\left( \frac{m}{\mu} \right) E_{2,m}}{(m - \sigma_p)(m - \sigma_p)} \]

where

\[ \sigma_p = \sum_{k=1}^{p} \rho_k, \quad \sigma_0 = 0, \quad \sigma_r < 1 \]

\[ \rho_k = \frac{\lambda_i S_k}{\mu}, \quad \lambda_i S_k \quad \text{and} \quad \frac{1}{\mu} \quad \text{are average arrival rate and average packet length of flow k, respectively.} \]

\[ \rho = \frac{\lambda}{\mu} \sum_{k=1}^{r} S_k = \frac{\lambda}{\mu} \] is the total load per link, and

\[ E_{2,m} = \frac{\rho^m m!}{m! (m - \rho)} \]

is well-known Erlang’s C formula.

Numerical Example

It is assumed five classes of flows, f1, f2, f3, f4 and f5, where f1 is highest priority. The percentages of flows are S1=30%, S2=30%, S3=20%, S4=10% and S5=10% for f1, f2, f3, f4 and f5, respectively. The average packet length for all flows is 8000 bits and the link capacity is 150 Mbits per second.

Figure 3.7 shows the numerical results of Equation 3.8, for m=1, that means, the average packet waiting time at output buffer. It can be observed that up to 80% of load all flows have insignificant waiting time at output buffer. For load equal to 90% only f5 flow class has significant packet waiting time, about 2.5 ms.
Figure 3.8 shows the average packet waiting time at input buffer considering only two servers \((m=2)\).

It can also be observed that even under heavy traffic \((\rho=.9)\) the average packet waiting time for all flow classes at input buffer is insignificant. Even using only two internal links practically there is no packet waiting in input buffer, in average.

The Table 3.2 shows the switch performance in terms of average waiting time as function of number of internal links \((m)\), considering a traffic load equal to 90%.

<table>
<thead>
<tr>
<th>flow</th>
<th>(m=1)</th>
<th>(m=2)</th>
<th>(m=3)</th>
<th>(m=4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(f_1)</td>
<td>6.5753e-005</td>
<td>8.6107e-006</td>
<td>1.3681e-006</td>
<td>2.0488e-007</td>
</tr>
<tr>
<td>(f_2)</td>
<td>1.4294e-004</td>
<td>1.1796e-005</td>
<td>1.6684e-006</td>
<td>2.3686e-007</td>
</tr>
<tr>
<td>(f_3)</td>
<td>3.7267e-004</td>
<td>1.5942e-005</td>
<td>1.9777e-006</td>
<td>2.6935e-007</td>
</tr>
<tr>
<td>(f_4)</td>
<td>9.0226e-004</td>
<td>1.9560e-005</td>
<td>2.2440e-006</td>
<td>2.9215e-007</td>
</tr>
<tr>
<td>(f_5)</td>
<td>2.5263e-003</td>
<td>2.2760e-005</td>
<td>2.4363e-006</td>
<td>3.0911e-007</td>
</tr>
</tbody>
</table>

4. Conclusion

A QoS provisioned CIOQ switch using crossbar structure with \(m\) parallel lines per output port was proposed in this paper. The packets at input buffers are transferred to the output buffers by means of \(mxN\) internal lines. Since all internal lines have the same speed as external links, no internal clock speedup is required so that the proposed structure is suited for high-speed switches.

To verify the switch performance, two switch structures were proposed. One of structures was based on FIFO scheduling and another was based on priority scheduling. By using queuing theory both of structures were analyzed. The results showed that only 2 internal links for each output port are sufficient for quick packet transfer from input buffers to the output buffers.

For switch using FIFO scheduling the results have shown that for \(N = 32\) input or output links with each link capacity of 622 Mbps and packet size of 8000 bits, the switching capacity for load equal .8 and \(m=2\) is about 1.99 millions of packets per second with packets spending about 78 \(\mu\)seconds in average at the switch.

For the switch using priority scheduling the same high capacity of FIFO switch is obtained but in this case it is convenient to provide QoS for flow classes.

5. References