Copyright Warning & Restrictions

The copyright law of the United States (Title 17, United States Code) governs the making of photocopies or other reproductions of copyrighted material.

Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or reproduction is not to be "used for any purpose other than private study, scholarship, or research." If a, user makes a request for, or later uses, a photocopy or reproduction for purposes in excess of "fair use" that user may be liable for copyright infringement,

This institution reserves the right to refuse to accept a copying order if, in its judgment, fulfillment of the order would involve violation of copyright law.

Please Note: The author retains the copyright while the New Jersey Institute of Technology reserves the right to distribute this thesis or dissertation

Printing note: If you do not wish to print this page, then select "Pages from: first page # to: last page #" on the print dialog screen

ABSTRACT

DESIGN AND STABILITY ANALYSIS OF HIGH PERFORMANCE PACKET SWITCHES

by
Zhen Guo

With the rapid development of optical interconnection technology, high-performance packet switches are required to resolve contentions in a fast manner to satisfy the demand for high throughput and high speed rates. Combined input-crosspoint buffered (CICB) switches are an alternative to input-buffered (IB) packet switches to provide high-performance switching and to relax arbitration timing for packet switches with high-speed ports.

A maximum weight matching (MWM) scheme can provide 100% throughput under admissible traffic for IB switches. However, the high complexity of MWM prohibits its implementation in high-speed switches. In this dissertation, a feedback-based arbitration scheme for CICB switches is studied, where cell selection is based on the provided service to virtual output queues (VOQs). The feedback-based scheme is named round-robin with adaptable frame size (RR-AF) arbitration. The frame size in RR-AF is adaptably changed by the serviced and unserviced traffic. If a switch is stable, the switch provides 100% throughput. Here, it is proved that RR-AF can achieve 100% throughput under uniform admissible traffic.

Switches with crosspoint buffers need to consider the transmission delays, or round-trip times to define the crosspoint buffer size. As the buffered crossbar switch can be physically located far from the input ports, actual round-trip times can be non-negligible. To support non-negligible round-trip times in a buffered crossbar switch, the crosspoint buffer size needs to be increased. To satisfy this demand, this dissertation investigates how to select the crosspoint buffer size under non-negligible round trip times and under uniform traffic. With the analysis of stability margin, the relationship between the crosspoint buffer size and round-trip time is derived.

Considering that CICB switches deliver higher performance than IB switches and require no speedup, this dissertation investigates the maximum throughput performance that these switches can achieve. It is shown that CICB switches without speedup achieve 100% throughput under any admissible traffic through a fluid model. In addition, a new hybrid scheme, based on longest queue-first (as input arbitration) and longest column occupancy first (as output arbitration) is proposed, which achieves 100% throughput under uniform and non-uniform traffic patterns.

In order to give a better insight of the feedback nature of arbitration scheme for CICB switches, a frame-based round-robin arbitration scheme with explicit feedback control (FRE) is introduced. FRE dynamically sets the frame size according to the input load and to the accumulation of cells in a VOQ. FRE is used as the input arbitration scheme and it is combined with RR, PRR, and FRE as output arbitration schemes. These combined schemes deliver high performance under uniform and nonuniform traffic models using a buffered crossbar with one-cell crosspoint buffers. The novelty of FRE lies in that each VOQ sets the frame size by an adjustable parameter, $\Delta_{i,j}$, which indicates the degree of service needed by $VOQ(i,j)$. This value is adjusted according to the input loading and the accumulation of cells experienced in previous service cycles.

This dissertation also explores an analysis technique based on feedback control theory. This methodology is proposed to study the stability of arbitration and matching schemes for packet switches. A continuous system is used and a control model is used to emulate a queuing system. The technique is applied to a matching scheme. In addition, the study shows that the dwell time, which is defined as the time a queue receives service in a service opportunity, is a factor that affects the stability of a queuing system. This feedback control model is an alternative approach to evaluate the stability of arbitration and matching schemes.

DESIGN AND STABILITY ANALYSIS OF HIGH PERFORMANCE PACKET SWITCHES

by
Zhen Guo

A Dissertation
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy in Computer Engineering

Department of Electrical and Computer Engineering

January 2006

APPROVAL PAGE

DESIGN AND STABILITY ANALYSIS OF HIGH PERFORMANCE PACKET SWITCHES

Zhen Guo

Dr. Roberto Rojas-Cessa, Dissertation Advisor — Date
Assistant Professor, Department of Electrical and Computer Engineering, New Jersey Institute of Technology

---

Dr. Nirwan Ansari, Committee Member — Date
Professor, Department of Electrical and Computer Engineering, New Jersey Institute of Technology

---

Dr. Aleksandar Kolarov, Committee Member — Date
Technical Leader, Cisco Systems

Dr. Edwin Hou, Committee Member — Date
Associate Professor, Department of Electrical and Computer Engineering, New Jersey Institute of Technology

---

Dr. Jie Hu, Committee Member — Date
Assistant Professor, Department of Electrical and Computer Engineering, New Jersey Institute of Technology

BIOGRAPHICAL SKETCH

Author: Zhen Guo

Degree: Doctor of Philosophy

Date: January 2006

Undergraduate and Graduate Education:

- Doctor of Philosophy in Computer Engineering,
  New Jersey Institute of Technology, Newark, NJ, January 2006
- Master of Science in Computer Engineering,
  New Jersey Institute of Technology, Newark, NJ, August 2003
- Master of Science in Computer Engineering,
  Chinese Academy of Space Technology, Beijing, China, July 1997
- Bachelor of Science in Computer Science,
  Institute of Astronautics, Beijing Union University, Beijing, China, July 1991

Major: Computer Engineering

Presentations and Publications:

Guo, Z. Rojas-Cessa, R. and Ansari, N.
"Packet Switches with Internally-Buffered Crossbars"
to appear as a book chapter

Guo, Z. and Rojas-Cessa, R.
"A Control Theoretic Analysis of Scheduling and Arbitration Schemes for Packet Switches,"
accepted by *IEEE Sarnoff Symposium'06*

Guo, Z. and Rojas-Cessa, R.
"Framed Round Robin Arbitration with Explicit Feedback Control for Combined Input-Crosspoint Buffered Packet Switches,"
submitted to *IEEE International Conference on Communications (ICC'06)*

Rojas-Cessa, R. Guo, Z. and Ansari, N.
"On the Maximum Throughput of a Combined Input-Crosspoint Queued Packet Switches,"
under review in *IEICE Transactions on Communications*

Rojas-Cessa, R. and Guo, Z.
"Round-Robin Selection with Adaptable Frame-Size for Combined Input-Crosspoint Buffered Packet Switches,"
accepted by *IEICE Transactions on Communications*

Rojas-Cessa, R. Dong, Z. and Guo, Z.
"Load-Balanced Combined Input-Crosspoint Buffered Packet Switch and Long Round-Trip Times,"
IEEE Communications Letters, Vol. 4, No. 7, pp. 661-663, July 2005

Rojas-Cessa, R. Guo, Z. and Ansari, N.
"Combining Distributed and Centralized Arbitration Schemes for Combined Input-Crosspoint Queued Packet Switches,"
IEEE International Conference on Networks (ICON'05), Nov. 2005

Guo, Z. and Rojas-Cessa, R.
"Analysis of a Flow Control System for a Combined Input-Crosspoint Buffered Packet Switch,"
IEEE Workshop on High Performance Switching and Routing (HPSR'05), pp. 336-340, May 2005

Guo, Z. and Rojas-Cessa, R.
"Stability Analysis of a Flow Control System for a Combined Input-Crosspoint Buffered Packet Switch,"
Conference on Information Sciences and Systems (CISS'05), March 2005

Guo, Z. and Savir, J.
"Analog Circuit Test using Transfer Function Coefficient Estimates,"
accepted and to appear in *IEEE Transactions on Instrumentation and Measurement*, Vol. 55, No. 1, Feb. 2006

Guo, Z. and Savir, J.
"Analog Circuit Test using Transfer Function Coefficient Estimates,"
IEICE Transactions on Information and Systems Special Issue on Test and Verification of VLSI, Vol. E87-D, No. 3, pp. 642-646, March 2004

Savir, J. and Guo, Z.
"Test Limitations of Parametric Faults in Analog Circuits,"
IEEE Transactions on Instrumentation and Measurement, Vol. 52, No. 5, pp. 1444-1454, October 2003

Guo, Z. and Savir, J.
"Analog Circuit Test using Transfer Function Coefficient Estimates,"
IEEE International Test Conference (ITC'03), pp. 1155-1163, October 2003

Guo, Z. and Savir, J.
"Coefficient-Based Test of Parametric Faults in Analog Circuit,"
IEEE Instrumentation and Measurement Technology Conference (IMTC'03), Vol. 1, pp. 71-75, May 2003

Savir, J. and Guo, Z.
"Test Limitation of Parametric Faults in Analog Circuits,"
IEEE Asian Test Symposium (ATS'02), pp.39-44, Nov. 2002

Savir, J. and Guo, Z.
"On the Detectability of Parametric Faults in Analog Circuits,"
IEEE International Conference on Computer Design (ICCD'02), pp. 273-276, Sept. 2002

Guo, Z. and Savir, J.
"Observer-Based Test of Analog Linear Time-Invariant Circuits,"
IEEE Electronic Design, Test and Applications (DELTA'02), pp. 13-17, 2002

Guo, Z. , Zhang, X. , Savir, J. and Shi, Y.
"On Test and Characterization of Analog Linear Time-Invariant Circuits using Neural Networks,"
IEEE Asian Test Symposium (ATS'01), pp. 338-343, 2001

Guo, Z. and Savir, J.
"Algorithm-Based Fault Detection of Analog Linear Time-Invariant Circuits,"
IEEE Instrumentation and Measurement Technology Conference (IMTC'01), Vol. 1, pp. 49-54, 2001

To my parents who made me realize anything can be achieved if you put forth the effort.

To my beloved wife for her always believing in me.

ACKNOWLEDGMENT

First and foremost, I would like to express my sincere appreciation to my advisor and mentor, Professor Roberto Rojas-Cessa, for his understanding, encouragement, invaluable instructions. I deeply appreciate his advice, guidance and academic insight. I treasure the opportunities he created for me to continue and complete my doctoral research. His advice is essential to the completion of this dissertation. I will never forget his help.

I am deeply grateful to Professor Nirwan Ansari, Associate Chair and Academic Advisor, for his tireless help, encouragement, guidance and support. I respect him and I am fortunate to get help from him.

I am deeply grateful to Dr. Ronald Kane, Dean of Graduate Studies, for his timely, continuous and invaluable support and encouragement. Without his help, it would be harder for me to complete my doctoral research.

I am deeply grateful to Professor Atam Dhawan, Chair of ECE Department, for his understanding and encouragement.

I acknowledge the valuable comments and discussion with my committee members: Dr. Aleksandar Kolarov (Technical Leader, Cisco Systems), Professor Edwin Hou and Professor Jie Hu. I would like to give thanks for their comments and helpful reviews on my dissertation. I also extend my special thanks to Professor Kenneth S. Sohn for his help, Professor Timothy N. Chang and Professor Durga Misra for their encouragement.

The friendship of Zhiyun Yang, Ying Li, Kun Li, Puttiphong Jaroonsiriphan, Qiming He, Yingqin Yuan, Chuanbi Lin, Ziqian Dong, Zhen Qin, Chen Fang, Hong Zhang, Jun Jiang, Li Zhu, and Amey B. Shevtekar is much appreciated.

Finally, I express my special gratitude to my parents for their dedicated and endless love to me, without their love and guidance I would be lost. I would like to thank my beloved wife, Yueling Li, for her patient understanding, help, support and love. I cherish and enjoy the happiness which my lovely daughter brings to me during these years. Last, but not least, I would like to thank my brother for his faith in me.

TABLE OF CONTENTS

TABLE OF CONTENTS
(Continued)

Chapter **Page**

LIST OF FIGURES

LIST OF FIGURES
(Continued)

CHAPTER 1

INTRODUCTION

1.1 Introduction to Packet Switches

The exchange of information within the Internet is made possible by the switches and routers interconnecting networks. These routers facilitate networks to communicate by using a shared language or protocol. An example of such protocols employed at different network layers are Ethernet, asynchronous transfer mode (ATM), or the popular TCP/IP suite. These protocols determine the packet formats and the way to find a route from the source host to the destination host.

In the remainder of this dissertation, the IP protocol is considered to define the packet term. Therefore, switches and routers are required to process IP packets. However, a large number of packet switches base their architecture on ATM technology, where different from IP packets, ATM packets have fixed lengths, called cells. IP packets, however, can be handled by cell-based switches as variable-length packets are segmented at the input ports, and switched from input to output in a cell-based fashion. Variable-length IP packets are re-assembled at the output ports, before they depart to another switches. Here, fixed-length packets are referred as cells, which are not necessarily ATM cells.

Packet switches identify the destination of packets at the input ports and forward them to the appropriate output ports, completing in this way the packet processing at layer 2 of the open system interconnection (OSI) model. Those switches that find out information about the connectivity between networks and paths to reach different possible destinations are referred to as routers. This information is summarized in a forwarding table that is used to determine the output port of the switch according the destination of the traversing packet. These routers are devices performing tasks of layers 2 and 3. Once output ports are defined in a forwarding table, a switch performs the scheduling and forwarding of packets.

As interconnection technologies mature, such as those base on optical technology, data rate increases and routers need to keep up with that by processing packets fast. The functions that require high performance in routers is identifying the packet type, so that the packet can be processed accordingly (including forwarding), and switching the packet from an input to an output port. Here, switching a packet means the packet is transferred from the input port to an output port. The task seems simple; however, it gets complex as there is the possibility that several packets need to go from different inputs to the same output. Therefore, this creates the necessity of interconnecting input and output ports, buffering packets and scheduling the packet switching time. A scheduler selects the time a packet is switched to the output. The complexity of the scheduler depends on the buffering strategy and on the selection scheme used.

A packet switch is comprised of input port cards and a switch fabric. An input port card, also known as a line interface card (LIC), determines the processing the switch performs on each packet. Some of these functions are: packet classification, destination lookup (IP lookup for IP packets), buffering, packet modification, and packet interfacing for internal switching. The switch fabric is an interconnection network used by the LICs. The way a switch operates internally depends on the switch fabric used. A plethora of switch fabrics have been developed for packet switches. Many of them have been incepted for the telephone network and are now applied to packet networks.

1.2 Development of Combined Input-Crosspoint Buffered Packet Switches

The internet continues to experience extraordinary growth. By any measure, the growth is remarkable on all fronts: the number of hosts, the number of users, the amount of traffic, the number of links, the bandwidth of individual links. In order to keep pace with the growth of Internet usage, higher capacity packet switches are needed with aggregate data rates of multiple terabits per second, and forwarding rates of billions of packets per second. However, after 10-years' booming development on switch, some people wander

how far can switch go forward in face of this big challenge? Novel switch architectures and arbitration schemes might provide the answer [10, 63].

Crossbar switching fabrics are very popular for switch implementation due to their non-blocking capability, simplicity, and their market availability. The performance of a switch can be analyzed according to the adopted buffering strategy.

An OB switch has no queues at the input ports. All arriving cells must be immediately delivered to their outputs. A major disadvantage is that simultaneously delivery of all arriving cells to the outputs require high internal interconnection bandwidth and memory bandwidth. For an $N \times N$ OB switch, the memory has to support N write accesses (to write N cells into output buffer) and one read access (to send one cell to the outgoing link) in one-cell time. This means that an OB switch must operate $N + 1$ times faster than the line rate. This requirement is known as internal speedup of a switch which is defined as the number of times that the switch core works faster than the input line rate. To ensure that there are no packets queued at the input ports, it is widely believed that an OB switch has to have an internal speedup of N. Unfortunately, the increase in the line rate and/or switch size makes it extremely difficult and impractical to build memories with adequate capacity for such high-bandwidth. However, this architecture has been used as a comparison reference for any switch model because of its high throughput and low delay. A switch with buffers* at the inputs, named input-buffered (IB) switch. IB switches are desirable because of their scalability and low hardware requirement. The IB switch has an internal speedup of 1 (also considered as no speedup) because the crossbar fabric has the same speed as that of the external line. It is well-known that, if first-in first-out (FIFO) input queues are used to hold arriving packets, head-of-line (HOL) blocking problem limits the throughput of only 58.6%. To eliminate HOL blocking, virtual output queuing (VOQ) can be used, each input buffer is partitioned into N queues with one queue for each output

*This dissertation uses the terms queue and buffer interchangeably.

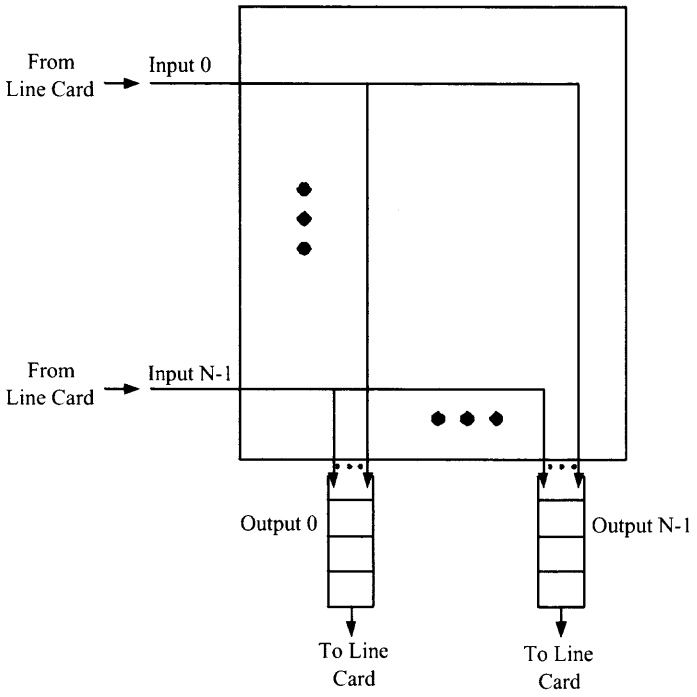

Figure 1.1 Output-buffered crossbar switch.

port (hence the name virtual output queuing). Each arriving packet is classified and then queued into the appropriate VOQ according to its destination output port.

However, IB switches need to resolve input and output contentions before cells are forwarded to the outputs. Arbiters at input and outputs perform the contention resolution by means of a matching process. Furthermore, the switching performance of an IB switch requires complex matching schemes to provide high-switching performance. This high complexity limits the switch port speeds. The requirements for arbiters to be feasible and to provide a high performance are: (a) low complexity, (b) fast contention resolution, (c) fairness and, (d) high matching efficiency. As an example, the matching scheme must perform input or output arbitration within 8 ns in an IB switch with 40 Gbps (OC-768) ports and 80-byte cells, assuming that input and output arbitrations may use up to half of a time slot and that the transmission delays are decreased to negligible amounts (e.g., the arbiters are implemented in the same chip, in a centralized way).

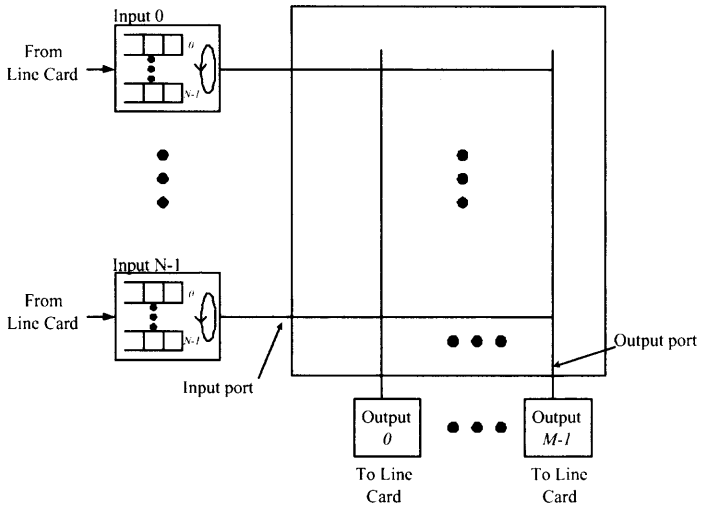

Figure 1.2 Input-buffered crossbar switch.

A matching can be classified as maximum or maximal. A maximum match is a maximum cardinality bipartite matching of input with packets queued to N outputs. A maximal match is a matching that cannot be improved without removing some input-output matches. A maximum weight matching (MWM) algorithm provides 100% throughput for any no-overbooking traffic theoretically. However, the scheme's complexity prevents its implementation for fast speeds. Maximal matching schemes have been considered as an alternative to maximum matching schemes [17]; iSLIP [14, 16], dual round-robin matching (DRRM), and longest output occupancy first algorithm (LOOFA) are some examples. To make up for the lack of efficiency that a maximal scheme has (compared to a maximum type), a number of iterations (where the number of iterations is the number of times that an algorithm is performed to obtain a cumulative result), speedup, or the number of both is used, as in LOOFA. iSLIP is a typical example of an iterative matching scheme. iSLIP provides 100% throughput for uniform traffic, but because of the arbitration time, it has been proposed for a small number of ports due to its centralized implementation. The transmission of phases such as request, grant and acknowledge are performed within a cell slot between input and output arbiters. This transmission of information reduces the

available time for arbitration because transmission phases are performed during the cell slot in serial with input and output arbitration, even when the transmission is done within a single chip.

Speedup is another approach to tackle the lack of efficiency by scheduling on IB switch, when the internal speedup is between 1 and $N-1$, buffering is required at both the inputs and outputs [82]. Hence, a combination of an input buffered and an output buffered switch is required, which is combined input and output buffered switch (CIOB). With a speedup of 2, it is proved that any maximal matching algorithm can achieve 100% throughput under any admissible traffic. However, speedup shortens the schedule time. Iterative-based algorithm may have no time to find a maximal matching.

The arbitration in a crosspoint buffered (CB) switch is only performed for input selection at each output of the buffered crossbar, where packets stored in the crosspoint buffers are considered. A CB switch is called a pure buffered crossbar because in this architecture buffering is only at the crosspoints. A large crosspoint buffer has been utilized to minimize cell loss rate. However, the number of buffers in a crossbar grows in the same order as the number of crosspoints, $O(N^2)$. This makes implementation costly for a large buffer size or large N. One way to keep the buffer complexity feasible is to use crosspoint buffers that are small in size.

An example of a CB switch was proposed in [1], where a 2×2 crossbar chip with a crosspoint memory of 16 Kbytes was implemented to provide an acceptable cell loss. In order to reduce the crosspoint buffer size, input buffers can be used.

Buffered crossbar switches need output arbiters to select a cell, which is stored in the crosspoint buffer, to be sent to the output port. Therefore, the performance of this switch depends on the output arbitration. The output arbitration scheme considers all crosspoint buffers and selects one according to a selection policy. Therefore, the time and computation complexity of an output arbitration scheme is in function of the number of crosspoint buffers (or inputs, N) at an output, or $O(N)$.

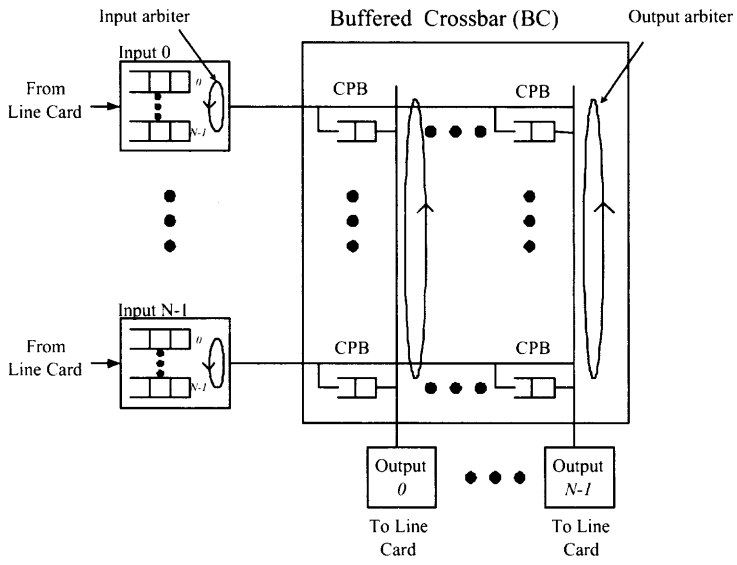

Figure 1.3 Combined input-crosspoint buffered crossbar switch.

An example of a BC switch that emphasizes on the output arbitration is found in the scalable distributed arbitration (SDA) switch [8]. The SDA switch reduces the arbitration time by using a distributed approach for the output arbitration in each output. Instead of considering N input at any given time, an arbiter is partitioned into $N - 1$ selectors, where each selector considers two buffers. This switch performs random selection of cells (crosspoints) at each output. The SDA switch has a crosspoint buffer, a transit buffer, an arbitration-control block (CNTL), and a selector at every crosspoint. A crosspoint buffer sends a request (REQ) to CNTL if there is at least one cell stored in the crosspoint buffer. A transit buffer stores several cells that are sent from either the upper crosspoint buffer or upper transit buffer. The transit buffer has a size of one or a few cells. The transit buffer size is determined by the round-trip delay of control signals between two adjacent crosspoints. The longest control signal transmission distance for arbitration within one cell time is the distance between two adjacent crosspoints. In a switch with an implementation of the output arbiters in a centralized fashion, the control signal for arbitration must pass through all the crosspoint buffers, belonging to the same output line to complete the selection

each time slot, and therefore, the arbitration time depends on the number of inputs (or crosspoints). In this way, the arbitration time in the SDA switch is independent of the number of input ports.

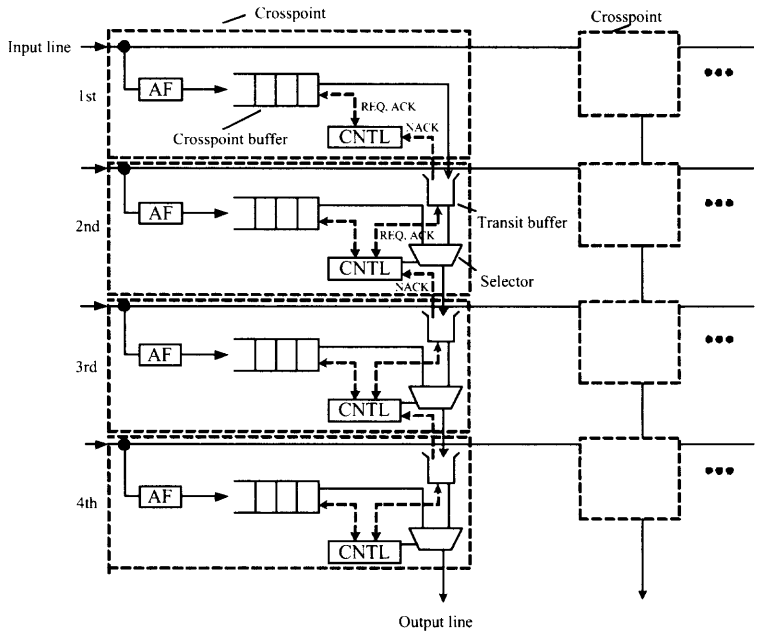

Figure 1.4 Scalable distributed-arbitration switch structure.

The SDA switch was tested under uniform traffic with Bernoulli arrivals with an input load of 0.95 for different switch sizes ($N = \{4, ..., 32\}$). This switch delivers a cell average delay of less than 100 time slots. Another feature of the SDA switch is fairness. This is achieved by the nature of the distribute selection scheme using different selection probabilities for different inputs at a crosspoint, such that the total selection probability by an output is the same for any input.

In order to reduce the crosspoint buffer size, input buffers can be used with larger capacity as these buffers are located in the input ports, and the amount of memory at the input ports (outside of the buffered crossbar) can be of large size (e.g., several memory

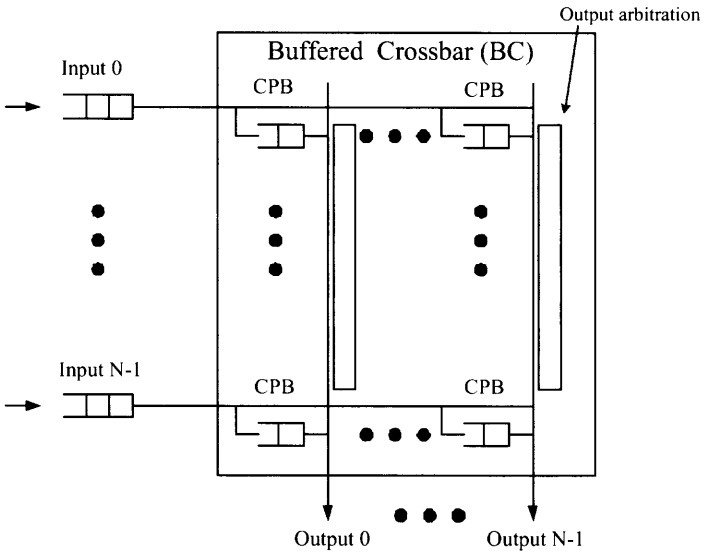

Figure 1.5 Combined input-crosspoint buffered crossbar switch with FIFO input buffers.

chips). CICB switches that use the FIFO policy in their input buffers are the simplest of them, and are called FIFO-CICB.

In [6], an input and crosspoint buffering matrix switching architecture with FIFO input buffers, or FIFO CICB switch, was proposed. In [7], a FIFO-CICB switch input buffers and random selection policy at the output was proposed and shown to provide high throughput. CICB switches with single-cell crosspoint buffers were proposed in [2, 3]. These switches also used FIFO input buffers at the input ports, or FIFO-CICB switches. The switches provide a throughput of 91%, where, however, the HOL blocking [5] was still present. The FIFO buffers at the inputs limit the maximum throughput in that switch because the HOL blocking can not be completely eliminated. These switches showed the need to remove the HOL blocking that was present in IB switch and then, inherited by FIFO CICB switches.

Another example of a FIFO-CICB switch, with a different architecture approach, a multiple-plane architecture, is the tandem-crosspoint (TDXP) switch [9]. The main purpose of the TDXP switch is to overcome HOL blocking by using the parallel switch technology.

This switch has multiple crossbar switch planes as shown in Figure 1.6. The switch planes are connected in tandem at each crosspoint. There is a one-cell buffer in the crosspoint. The internal speedup in each plane is the same as the input/output line speed. Each switch plane can transmit only one cell to each output port within one cell time slot. The HOL blocking phenomenon occurs at the input buffers, where the FIFO policy is used.

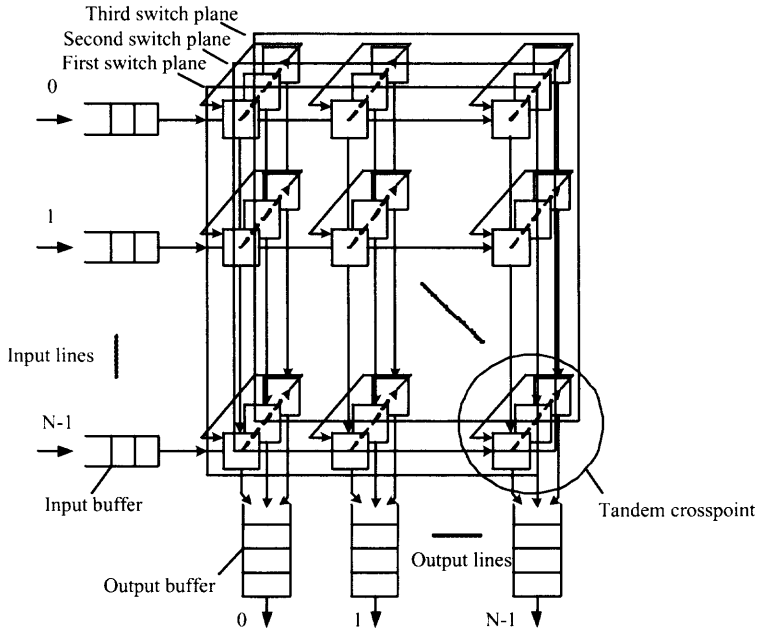

Figure 1.6 Tandem-crosspoint (TDXP) switch with three planes.

The TDXP switch improves the switching performance by using multiple planes. In this way, cells will not cause HOL blocking as there is room for a cell in each switch plane. This is similar to letting the first K cells of a FIFO participate in the output arbitration process, where K is the number of planes in the TDXP switch. In a 32×32 switch, the throughput provided is above 95%, which is a significant improvement over an IB switch with FIFO input buffers.

As in IB switches, the HOL blocking problem for FIFO buffers can be overcome in CICB switches by using VOQs, or VOQ-CICB switches. For the sake of brevity, VOQ-CICB switches are referred as CICB switches in the remainder of this dissertation.

CICB switches use time efficiently as input and output port selections are performed separately. For each input, there is one input scheduler which separately deals with input contention, i.e. to decide which VOQ in this input is allowed to transfer a cell into the switch core. In a similar way, there is an output scheduler which independently deals with output contention for each output, i.e. to decide which crosspoint buffer is allowed to transfer a cell out of the switch core. Back to the example of the stringent timing, a CICB switch with 40-Gbps and 80-byte packets can perform input (or output) arbitration within 16 ns, therefore, the timing for arbitration is extended.

1.3 Research Challenges and Motivations

In CICB switches, high matching efficiency is achieved with simpler arbitration schemes than those used in bufferless crossbars (i.e., IB switches) at the expense of having to accommodate buffers in the crosspoints.

A CICB switch with timestamp-based arbitration and VOQs at the input ports showed that the crosspoint-buffer size can be small if the VOQs are provided with enough storing capacity [12]. Furthermore, it has been shown that a CICB switch using one-cell crosspoint buffers (CIXB-1), a simple round-robin arbitration (RR) scheme for input/output arbitration, and a credit-based flow control provide 100% throughput for uniform traffic [42]. However, as actual traffic may present nonuniform distributions, it is necessary to provide arbitration schemes that provide 100% throughput for admissible traffic. Admissible traffic is defined as: Let us denote $\lambda_{i,j}$ to the cell arrival rate at input i for output j that is received in $VOQ_{i,j}$. Let us consider admissible traffic such that

$$\sum_{i} \lambda_{i,j} \leq 1 \tag{1.1}$$

and

$$\sum_{j} \lambda_{i,j} \leq 1. \tag{1.2}$$

One way to provide 100% throughput under nonuniform traffic patterns is by using weight-based arbitration schemes, where weights are assigned to input queues proportionally to their occupancy or HOL cell age. It has been shown that weight-based [46] and priority-based [48] schemes in buffered crossbars can provide high throughput under various traffic patterns. Two schemes were presented in [46]: one is based on the selection of the longest VOQ occupancy at inputs and round-robin selection at the outputs; the other scheme is based on the selection of the oldest cell first (OCF) instead of VOQ occupancy. However, weight-based schemes need to perform comparisons among all contending queues, which can be a large number, thus increasing the implementation complexity. Moreover, weight-based schemes (e.g., queue-occupancy based) may starve some queues for very long time to provide more service to the congested ones, presenting unfairness. On the other hand, RR algorithms have been shown to provide fairness and implementation simplicity, as no comparisons are needed among queues, and high-performance under uniform traffic [14]. However, schemes based on round-robin selection have not been shown to provide nearly 100% throughput under nonuniform traffic patterns with a buffered crossbar that have crosspoint buffers of small size. It has been shown that a switch using RR needs a large crosspoint buffer to provide high throughput under admissible unbalanced traffic [15], where the unbalanced traffic model is a nonuniform traffic pattern [42]. This large buffer can make the implementation of a switch costly.

A question arises: is it possible to provide an arbitration scheme based on round-robin selection for buffered crossbars such that a switch can deliver high throughput under admissible traffic with nonuniform distributions, such as unbalanced traffic, with a small crosspoint buffer size?

Frame-based matching have been shown to have improved switching performance under different traffic scenarios [59]. However, how to set the frame size is a complex issue. In Chapter 2, an arbitration scheme is introduced for buffered crossbars, it is based on round-robin selection, which uses the concept of adaptable-size frame. The studied scheme is named round-robin arbitration with adaptable frame size (RR-AF) selection [60], [61]. The frame size is called adaptable as it is determined by the amount of service that a queue receives. In this chapter, it is shown that this arbitration scheme can achieve nearly 100% throughput under several nonuniform traffic patterns with one-cell crosspoint buffers. The performance results presented in this chapter shows that this switch retains the high performance, 100% throughput, of simple round-robin arbitration under uniform traffic.

In Chapter 3, it is investigated how to select the crosspoint buffer size under non-negligible round trip times. With the analysis of stability margin, the relationship between the crosspoint buffer size and round trip time (R_0) is derived.

A flow control mechanism is considered, where the parameter in observation is the queue occupancy. In this case, control theory can be applied to analyze and design queue management schemes. Most of AQM work focuses on supporting congestion management for the transmission control protocol (TCP) flows. However, only a few of the previous works are applied to switches. The main focus of this chapter is to apply control theory to analyze a flow control mechanism for a CICB switch. It is considered that credit-based flow control is used for avoiding buffer overflow [21]. This flow control mechanism has been applied to CICB switches with a negligible transmission delay [42], where the transmission delays are the propagation delays of sending a cell from the input port to a crosspoint, and of sending the flow control information from the crosspoint buffer to the input port. The sum of the transmission delays plus the selection delays at inputs (VOQ selection) and outputs (crosspoint buffer selection) is called round-trip time. As the buffered crossbar switch can be physically located far from the input ports, actual round trip times can be non-negligible. To support non-negligible round-trip times in a buffered-crossbar switch,

the crosspoint-buffer size needs to be increased, such that up to R_0 cells can be buffered. Non-negligible round trip delays have been considered for practical implementations [35, 36].

However, in a credit-based flow control mechanism, the stability of the switch, as a product of the round-trip times and crosspoint-buffer size, might be difficult to analyze. Therefore, a proportional (P) controller is used for a flow control mechanism for a CICB switch. The relationship between the round-trip times and crosspoint buffer size and the effect on stability is analyzed.

Stability is of utmost importance in switch design. Stability of a switch determines its throughput. If a switch is stable, it can provide 100% throughput. It has been shown that CICB switches are stable under uniform admissible traffic. In Chapter 4, it is shown that a CICB switch achieves 100% throughput without speedup under any admissible traffic and prove its stability through a fluid model. In addition, a new hybrid scheme with longest queue first as input arbitration and longest column occupancy first as output arbitration is proposed, which can achieve 100% throughput under uniform and non-uniform traffic patterns.

The intuition of this result lies on the knowledge that CICB switches provide higher performance than IB switches [42, 46], and that IB switches can provide 100% throughput under admissible traffic with no speedup [44], although with a high-complexity matching scheme.

In RR-AF scheme, the frame size increases by a constant number, independently of the actual size needed, each time a frame is completely served. The frame size decreases each time a VOQ misses an opportunity to be served. Although this scheme provides high performance, it is difficult to analyze. To give a better insight into the feedback nature of arbitration schemes for CICB switches, Chapter 5 introduces a frame-based round-robin arbitration scheme with explicit feedback control (FRE) for CICB packet switches. This scheme adopts a frame-based arbitration scheme that dynamically sets

the frame size according to the input load and to the number of cells accumulated in the input queues. The scheme is analyzed with control theory and the switching performance is studied by computer simulations. The combination of FRE (as the input arbitration scheme) with other weightless arbitration schemes (as output arbitrations) are tested. It is shown that FRE provides high throughput under several admissible traffic patterns using a CICB switch with one-cell crosspoint buffers.

The existent stability analysis approaches concentrate on the evolution of the queue length. If no overflow occurs to any queue, then the switch is stable. In fact, each queue is one subsystem of the whole switch, one important factor in the switch system is the arbitration scheme as this decides which queue can obtain the opportunity to transfer packets and also determines how often and how soon the queue can be served.

A question arises: what is the degree in a scheduling scheme affects the stability of the whole system, and how to analyze these issues using a control-theoretical technique?

In Chapter 6, a queuing system is modeled and it is shown that the dwell time, defined as the time a queue receives service in a service opportunity, is a factor that affects the stability of a queuing system. Furthermore, a case study of an arbitration scheme on a 2×2 switch is used to show that the feedback control model is an alternative approach to evaluate the stability of an arbitration scheme.

CHAPTER 2

STABILITY ANALYSIS OF FRAME-BASED ARBITRATION WITH ROUND-ROBIN SELECTION FOR A COMBINED INPUT-CROSSPOINT BUFFERED PACKET SWITCH

2.1 Introduction

Here, a frame is related to VOQ. A frame is the set of one or more cells in a VOQ that are eligible for matching in successive time slots. However, how to set the frame size is a complex issue [19, 20, 77]. The studied scheme is named round-robin arbitration with adaptable frame size selection (RR-AF) [60, 61]. There are two novelties associated with RR-AF. First, it is an arbitration scheme for a CICB switch. Secondly, it is based on a round-robin selection and uses the concept of adaptable-size frame. RR-AF is different from a common frame-based scheme. The frame size is called adaptable as it is determined by the amount of service that a queue receives. Stability of a switch determines its throughput. If a switch is stable, it can provide 100 % throughput.

In this chapter, we present the stability analysis of the RR-AF arbitration scheme and prove that under admissible uniform traffic, a CICB switch can provide 100% with the RR-AF arbitration scheme. The appeal of RR-AF scheme lies on that RR-AF as a round-robin based scheme can achieve 100% throughput.

This chapter is organized as follows. Section 2.1.1 presents the switch model under study, Section 2.1.2 introduces the RR-AF arbitration scheme and shows a simulation study of throughput under uniform and nonuniform traffic patterns. Section 2.2 presents a stability analysis of the RR-AF arbitration scheme. Section 2.3 presents the conclusions.

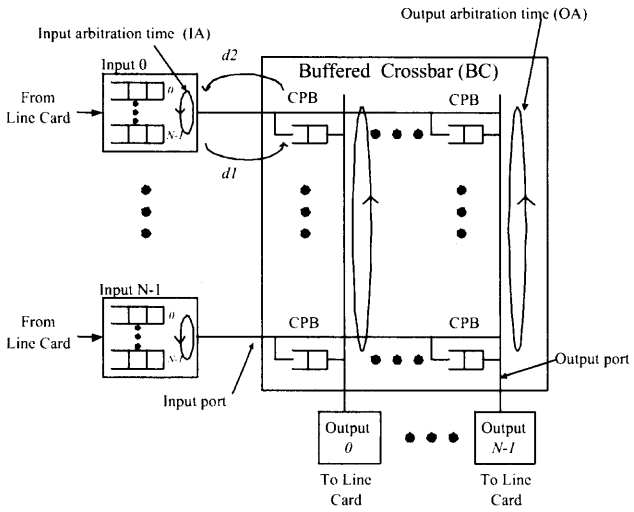

Figure 2.1 $N \times N$ buffered crossbar with VOQs.

2.1.1 Combined Input-Crosspoint Buffered Switch Model

Consider a CICB switch with N inputs and outputs. In this switch model, there are N VOQs at each input. A VOQ at input i, where $0 \leq i \leq N-1$, that stores cells for output j, where $0 \leq j \leq N-1$, is denoted as $VOQ_{i,j}$. A crosspoint (CP) element in the CICB that connects input port i to output port j is denoted as $CP_{i,j}$. The buffer at $CP_{i,j}$ is denoted as $CPB_{i,j}$. The size of $CPB_{i,j}$, k, is indicated by the number of cells that can be stored. A credit-based flow-control mechanism indicates to input i whether $CPB_{i,j}$ has room available for a cell or not, as described in [42]. $VOQ_{i,j}$ is said to be eligible for selection if the VOQ is not empty and the corresponding $CPB_{i,j}$, at buffered crossbar (BC), has room to store a cell.

The round trip (RT) time, as in [42], is defined as the sum of the delays of the input arbitration (IA), the transmission of a cell from an input to the crossbar ($d1$), the output arbitration (OA), and the transmission of the flow-control information back from the crossbar to the input ($d2$). Figure 2.1 shows an example of RT for input 0 by showing the transmission delays for $d1$ and $d2$, and arbitration times, IA and OA. Cell and bit alignments are included in the transmission times. The condition for this switch to avoid

underflow, is such that:

$$RT = d1 + OA + d2 + IA \leq k \tag{2.1}$$

where k is the crosspoint buffer size, in time slots, which is equivalent to the number of cells that can be stored. In other words, the crosspoint buffer must be able to store a number of cells to keep the buffer busy (i.e., transmitting cells) during at least one RT time.

2.1.2 Round-Robin with Adaptable-Size Frame (RR-AF) Arbitration Scheme

The studied arbitration scheme is round-robin based. Each time a VOQ (or a CPB at an output) is selected by the arbiter, the VOQ gets the right to forward a frame, where a frame is formed by one or more cells. Each cell of a frame is dispatched in one time slot. The frame size is determined by the serviced and unserviced traffic, such that no intervention is needed to select the frame size. We call this arbitration round-robin with adaptable-size frame (RR-AF). The amount of serviced (and unserviced) traffic depends on the experienced load by queues.

In each VOQ (and CPB) there are two counters: a frame-size counter, $FSC_{i,j}(t)$, and a current service counter, $CSC_{i,j}(t)$. The value of $FSC_{i,j}(t)$, $|FSC_{i,j}(t)|$, indicates the frame size; that is, the maximum number of cells that $VOQ_{i,j}$ can send in back-to-back time slots to the buffered crossbar, one cell per time slot. The initial value of $|FSC_{i,j}(t)|$ is one cell (i.e., its minimum value). It is considered that $|FSC_{i,j}(t)|$ can be as large as needed, although practical results have shown that its value does not reach large numbers. $CSC_{i,j}(t)$ counts the number of serviced cells at time slot t in a frame corresponding to a VOQ, where the frame size is indicated by FSC, in a regressive fashion. A regressive-fashion count is used in CSC as CSC only considers FSC at the end of a serviced frame. The initial value of $CSC_{i,j}(t)$, $|CSC_{i,j}(t)|$, is one cell (i.e., its minimum value).

The input arbitration process is as follows. An input arbiter selects an eligible $VOQ_{i,j'}$ in round-robin fashion, starting from the pointer position, j. For the selected $VOQ_{i,j'}$, if $|CSC_{i,j'}(t)| > 1$, $|CSC_{i,j'}(t+1)| = |CSC_{i,j'}(t)| - 1$, and the input pointer remains at $VOQ_{i,j'}$, so that this VOQ has the higher priority for service in the next time slot and the frame transmission can continue. If $|CSC_{i,j'}(t)| = 1$, the input pointer is updated to $(j'+1) \mod N$, $|FSC_{i,j'}(t)|$ is increased by f cells, and $|CSC_{i,j'}(t)| = |FSC_{i,j'}(t)|$. For any other $VOQ_{i,h}$, where $h \neq j'$, which is empty or inhibited by the flow-control mechanism, and it is positioned between the pointed $VOQ_{i,j}$ and the selected $VOQ_{i,j'}$: if $|FSC_{i,h}(t)| > 1$, $|FSC_{i,h}(t+1)| = |FSC_{i,h}(t)| - 1$. If there exist one or more VOQs that fit the description of $VOQ_{i,h}$ at a given time slot, it is said that those VOQs missed a service opportunity at that time slot. The increment of the frame size, done by f cells, is performed each time the previous complete frame of a VOQ has been serviced. The value of f has to be chosen as discussed in the following section.

For the sake of clarity, the following pseudo-code describes the input arbitration scheme, as seen at an input:

-At time slot t, starting from the pointer position j, find the nearest eligible $VOQ_{i,j'}$ in a round-robin fashion.

-Send the HOL cell from $VOQ_{i,j'}$ to $CPB_{i,j'}$ time slot $t+1$.

-If $|CSC_{i,j'}(t)| > 1$ *then*

$|CSC_{i,j'}(t+1)| = |CSC_{i,j'}(t)| - 1$,

the pointer points to j'.

-else $|FSC_{i,j'}(t+1)| = |FSC_{i,j'}(t)| + f$,

$|CSC_{i,j'}(t+1)| = |FSC_{i,j'}(t+1)|$,

the pointer points to (j'+1) modulo N.

-For VOQ(i, h), where $j \leq h < j'$ *for* $j < j'$, *or* $0 \leq h < j'$ *and* $j \leq h \leq N-1$ *for* $j > j'$:

$FSC_{i,h}(t+1) = FSC_{i,h}(t) - 1$.*

*Note that when $j' = j$, there is no $VOQ(i,h)$.

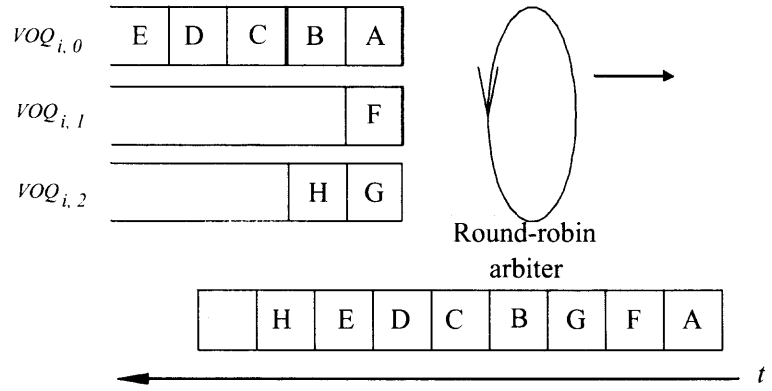

Figure 2.2 Example of RR-AF among three queues in a 3×3 switch.

- Go to the next time slot.

Note that f may be equal to a constant or a variable value. In general, f assumes the finite value of N, unless otherwise stated. The value of f affects the performance of RR-AF in different traffic scenarios. Note that when $f = 0$, RR-AF becomes RR.

The output arbitration works in a similar way to the input arbitration, considering $CPB_{i,j}$ and the corresponding counters in each crosspoint. Figure 2.2 shows an example of RR-AF at an input. Assume that the queues shown in the figure are the VOQs of input i in a 3×3 switch. Initially, all queues have three cells each, as Figure 2.2 shows. Assuming that the FSC for each queue has the initial value of one, a cell from each queue is served in a round-robin fashion. Then, each frame is increased by N cells; therefore, the remaining two cells in each queue are served back-to-back. The cells leave the input as the figure shows.

Figure 2.3 shows an example of the adjustment of $FSC_{i,j}$. In this example, $VOQ_{i,1}$ and $VOQ_{i,3}$ have cells, five and three, respectively, and no VOQ is inhibited by the flow-control mechanism. The highest priority is given to $VOQ_{i,0}$ at time slot t. During this time slot, the input arbiter selects $VOQ_{i,1}$ to be the next that sends a cell to the buffered crossbar. Then $VOQ_{i,0}$ misses an opportunity to send cells as it is empty, and its FSC value decreases by one cell. Since $VOQ_{i,1}$ has five cells and $FSC_{i,1} = 2$, the VOQ holds the

priority (or token) so that it receives service in the next time slot. Since the VOQ has three cells, it completes frame service and at time slot $t + 2$ its FSC value is increased by four, and the arbiter selects $VOQ_{i,2}$ as the next VOQ to receive service, but $VOQ_{i,2}$ misses an opportunity to send cells as it is empty and its FSC value decreases by one cell. Then, the priority is given to $VOQ_{i,3}$ at time slot $t + 3$.

The performance evaluations of two CICB switches, one using RR-AF arbitration and the other using RR arbitration, are presented. In addition, an OB switch is also considered in the evaluations for comparison purposes. The performance evaluations are produced by computer simulation. The traffic models considered have destinations with uniform and nonuniform distributions, the latter is called unbalanced. Both models use Bernoulli arrivals. The simulation does not consider the segmentation and re-assembly delays. The simulation results are obtained with a 95% confidence interval, not greater than 5% for the average cell delay.

Figure 2.4 shows simulation results of two 32×32 CICB switches with RR-AF, RR, and an OB switch under uniform traffic with Bernoulli arrivals ($l = 1$) and bursts with average lengths of 10 and 100 cells ($l = 10$ and $l = 100$). The burst length is exponentially distributed. The buffered crossbars have crosspoint buffers with a size of one cell each.

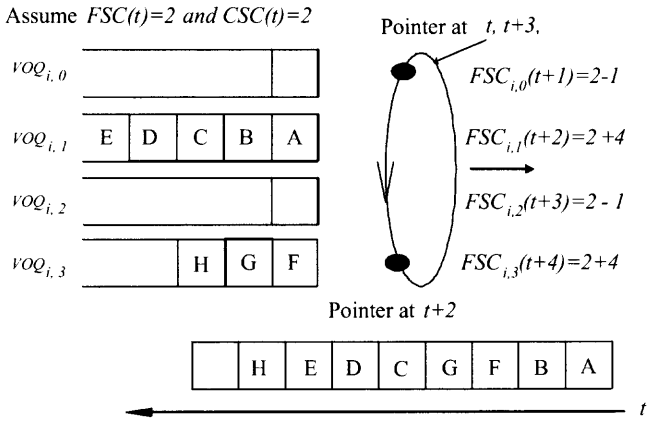

Figure 2.3 Example of VOQs missing opportunities for cell forwarding.

The simulation shows that the RR-AF arbitration scheme provides 100% throughput under uniform traffic.

This figure also shows that the average delay performance of RR-AF under Bernoulli arrivals is close to that of RR, and therefore, to that of an OB switch. The adaptable frame-size condition in the arbitration does not degrade the throughput performance, neither does it increase the average delay under this traffic model. As the RR-AF uses the history of serviced and unserviced traffic from the queues (i.e., VOQ and CPB), the switch practically adapts itself to uniform traffic.

RR-AF was simulated with different sizes of k. The result of the simulation shows that there is no measurable improvement by increasing the size of k. This result is expected as the average delay of RR-AF with $k = 1$ is close to that of an OB switch. Therefore, the increasing of k negligibly affects the results. As in [42], the size of k needs to be determined by the RT time. As the size of k does not affect the performance of RR-AF, k is assigned the value of one cell, (i.e., $k = 1$), in the remainder of the chapter, unless otherwise stated.

RR-AF and RR arbitrations were simulated under a nonuniform traffic model, the unbalanced traffic model [42]. The unbalanced traffic model uses a probability, w, as the

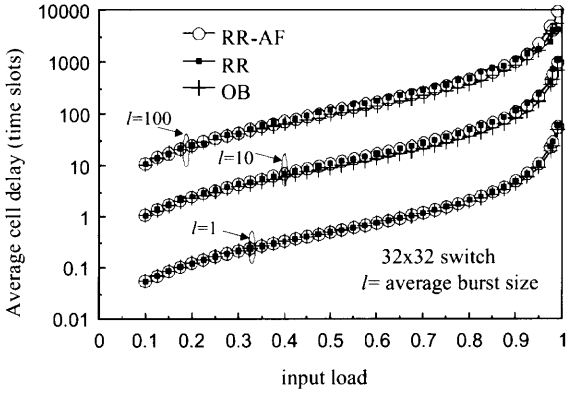

Figure 2.4 Average delay of RR-AF arbitration under Bernoulli and bursty uniform traffic.

fraction of input load directed to a single predetermined output, while the rest of the input load is directed to all outputs with uniform distribution. Let us consider input port s, output port d, and the offered input load for each input port ρ. The traffic load from input port s to output port d, $\rho_{s,d}$ is given by,

$$\rho_{s,d} = \begin{cases} \rho\left(w + \frac{1-w}{N}\right) & \text{if } s = d \\ \rho\frac{1-w}{N} & \text{otherwise.} \end{cases} \tag{2.2}$$

When $w = 0$, the offered traffic is uniform. On the other hand, when $w = 1$, it is completely directional, from input s to output d, where $s = d$.

Two combined input-crosspoint buffered switches of size $N = 32$, one with RR-AF and the other with RR, were simulated under unbalanced traffic. The switch with RR-AF uses $k = 1$ and for comparison, RR uses $k = 1$ and $k = N = 32$. Figure 2.5 shows that RR-AF, with $k = 1$ and $f = N$, provides well above 99% throughput under the complete range of w. It is considered that this throughput is nearly 100% for practical purposes. These results show that RR-AF with $k = 1$ outperforms RR with $k = 32$. This results in a feasible implementation of buffered crossbars as the size of the crosspoint buffer is reduced. In this example, RR, with $k = 32$ and a cell size of 64 bytes, would need 16 Mb of memory, while RR-AF, with $k = 1$, would need 512 Kb of memory. Furthermore, the switch with RR-AF can provide nearly 100% throughput under unbalanced traffic.

RR-AF with $f = N$, is also tested under other nonuniform traffic models: Chang's [51] and asymmetric [56].

Chang's traffic model can be defined as $\rho = 0$ for $i = j$ and $\rho = \frac{1}{N-1}$, otherwise. Figure 2.6 shows the average cell delay experienced by a 32×32 switch using RR-AF under these traffic models. As the figure shows, the throughput of RR-AF is 100% under Chang's and Asymmetric traffic models. The average delay under Chang's traffic is larger than that of the asymmetric's traffic; however, the difference is small. RR-AF adapts the frame size to the different loads offered to each input and output.

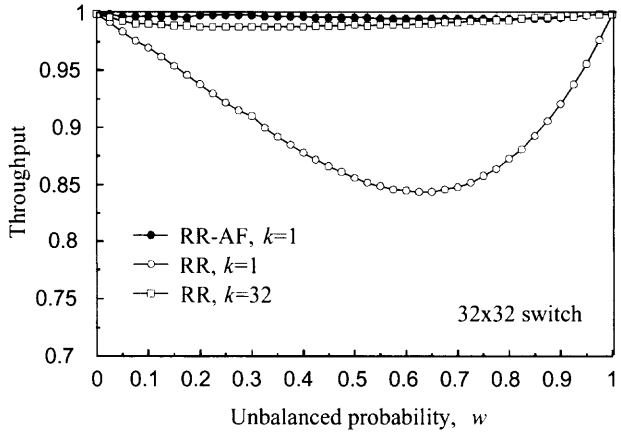

Figure 2.5 Throughput performance of RR-AF under unbalanced traffic.

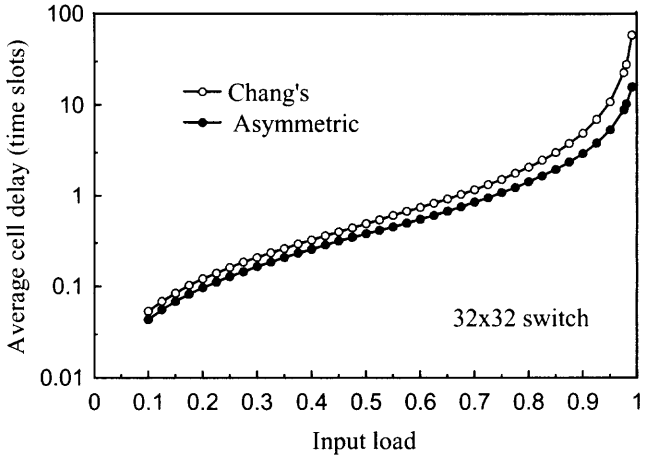

Figure 2.6 Average cell delay of RR-AF under Chang's and asymmetric traffic.

2.2 Stability Study

RR-AF arbitration is based on round-robin and it aims to improve the throughput under non-uniform traffic. The attractiveness of RR-AF lies on keeping the property of round-robin based schemes to deliver stability, and therefore, 100% throughput under uniform traffic. We use the definition of stability as presented in [18].

The high throughput of RR-AF is the product of increasing or decreasing service for a queue in proportion to its received and missed service, respectively. RR-AF ensures

service to the queues with high load by increasing the frame size, and to the other queues by using round-robin selection. In addition, the decreasing policy (i.e., FSC is decremented by one unit each time the VOQ misses service) for the frame-size counter ensures that the counter does not increase infinitely, as observed experimentally.

In this section, we prove that RR-AF, with f in a general sense, provides 100% throughput under admissible traffic, despite the use of the adaptable-size frame concept. We focus this proof on the input arbitration and VOQs. However, the result applies to the output arbitration and crosspoint buffers [64].

In order to analyze the stability of RR-AF, the queue length evolution by every time slot is not enough in analyzing the changing of the frame size.

In this analysis, the following definitions are used.

Definition 1. *A cycle is the service opportunity given to a VOQ where the number of cells that can be sent in consecutive time slots to the crosspoint can be up to the frame size. The cycle length is given in the number of time slots that the VOQ receives service. The start of a cycle is determined when a VOQ is selected to receive service at time slot t if that VOQ received service at time $t-1$.*

Definition 2. *The completion service rate $R^c_{i,j}$ is the rate at which $VOQ_{i,j}$ finishes frame service per cycle.*

Definition 3. *The miss service rate $R^m_{i,j}$ is the rate at what $VOQ_{i,j}$ misses service per cycle, including the following two reasons of the service miss: i) when the number of cells in a VOQ is smaller than the frame size, and ii) when a VOQ cannot send cells to the crosspoint for lacking of room in the crosspoint buffer. Therefore, $R^m_{i,j} = 1 - R^c_{i,j}$.*

In addition, we use the following notations:

$m_{i,j}$ denotes the accumulative total number of time slots that $VOQ_{i,j}$ receives service from t_0 to any time t, where t_0 is the starting time and t is any time slot such that $t > t_0$.

$\sigma_{i,j}$ is the cumulative number of opportunities a VOQ receives for service from cycle n_0, the time before VOQ receives any service during the switch working time, to cycle n.

$C_{i,j}^{inc}$ is the cumulative number of cycles where FSC increases until cycle n.

$C_{i,j}^{min}$ is the cumulative number of cycles where FSC has no changed because it has reached the minimum of one cell.

In this section, we denote the value of FSC at the end of cycle n as $FSC_{i,j}(n)$. Notice that $FSC(n)$ is different from $FSC(t)$ in section 2.1.2.

In addition, let $E[FSC_{i,j}(n)]$ denote the expected frame size of any VOQ at the end of n^{th} cycle. Since the average arrival rate is $\lambda_{i,j}$, let $\lambda_{i,j}E(x)$ be the number of cell arrivals per cycle (based on Little's theorem), where $E(x)$ is the average number of time slots that a VOQ receives service. Also, we denote the occupancy of a VOQ at the end of cycle n as $L_{i,j}(n)$.

Under traffic with uniform distribution among all outputs, the stability of the switch is directly related to the stability of the frame size of each queue. The stability of RR-AF is then based in the proof of the following claim:

Theorem 1. *A CICB switch using RR-AF scheduling algorithm is stable under traffic with uniform distribution.*

Proof. We assume that all inputs receive traffic independently and identically distributed. Therefore, identical service should be expected in each VOQ.

Since the service that a VOQ (or CPB) receives is determined by FSC, then we define the following lemma.

Lemma 1. *In a CICB packet switch using RR-AF as input arbitration, $VOQ_{i,j}$ is stable if $FSC_{i,j}$ is stable, under uniform traffic.*

Proof. When $FSC_{ij}(n)$ is stable, $L_{i,j}(n)$ can be either cases:

(*i*) $\lim_{n\to\infty} L_{i,j}(n) = \infty$.

(*ii*) $\lim_{n\to\infty} L_{i,j}(n) = a$, where a is a finite value and $a \geq 1$.

Let's consider the case (i) first: in a cycle, the service to $VOQ_{i,j}$ always complete because $\lim_{n\to\infty} L_{i,j}(n) = \infty$. $FSC_{i,j}(n)$ will be increased by f each time. Therefore $FSC_{i,j}(n)$ can not be bounded by a finite value, which contradicts with the assumption that $FSC_{ij}(n)$ is stable.

Now let's consider the case (ii): $\lim_{n\to\infty} L_{i,j}(n) = a$ means that $VOQ_{i,j}$ receives service all the time and $L_{i,j}(n)$ will never go to infinity. Since we have already proved that case (i) is impossible, so only case (ii) stands.

Summing up the cases above, if $FSC(n)$ is stable then $L(n)$ is stable.

□

For completeness, we state the following corollary:

Corollary 1. *Under uniform traffic, if $FSC_{i,j}(n)$ is unstable then $L_{i,j}(n)$ is unstable.*

Proof. We prove that the following state is false: if $lim_{n\to\infty} FSC_{i,j}(n) = \infty$ then $L_{i,j}(n)$ is stable. Let's assume that the statement is true. There must be that $L_{i,j}(n)$ is bounded by a finite value b (i.e., $\lim_{n\to\infty} L(n) = b$) and therefore $FSC_{i,j}(n)$ increases its value by f each cycle until it reaches the value of b. At this point $FSC(n)$ cannot continue increasing its value at each cycle, and therefore $FSC_{i,j}(n)$ converges to a finite value b, which contradicts the initial assumption. Therefore, if $FSC_{i,j}(n)$ is unstable, and $L_{i,j}(n)$ cannot be stable under uniform traffic.

□

Now, it remains to prove that $FSC_{i,j}(n)$ is stable. For this, let's consider the behavior of $FSC_{i,j}(n)$, and by stating the following lemma:

Lemma 2. *A CICB switch using RR-AF and under traffic with uniform distribution has $R^m_{i,j} > \frac{f}{f+1}$.*

Proof. The accumulated FSC value from cycles n_0 to n, where $n > n_0$, is

$$FSC_{i,j}(n) = FSC_{i,j}(0) + fC^{inc} - (\sigma_{i,j} - C^{inc}_{i,j} - C^{min}_{i,j}), \tag{2.3}$$

where $FSC_{i,j}(0)$ is the initial FSC value at n_0.

Let's assume that a frame is completely served at this cycle. The inequality involving the stationary state follows:

$$FSC_{i,j}(0) + fC^{inc}_{i,j} - (\sigma_{i,j} - C^{inc}_{i,j} - C^{min}_{i,j}) \leq \lambda_{ij}E(x) + \delta_{i,j}, \tag{2.4}$$

where $\delta_{i,j}$ is the discrepancy between the actual and the expected values. Then, we can express C^{inc} as:

$$C^{inc}_{i,j} \leq \frac{\lambda_{ij}\frac{m}{\sigma_{i,j}} + \sigma_{i,j} + \delta_{i,j} - C^{min}_{i,j} - FSC_{i,j}(0)}{f+1}. \tag{2.5}$$

Recalling that $R^c_{i,j} = \frac{C^{inc}_{i,j}}{\sigma_{i,j}}$ and using (2.5), we have:

$$\frac{C^{inc}_{i,j}}{\sigma_{i,j}} \leq \frac{1}{f+1} + \frac{\lambda_{ij}\frac{m}{\sigma_{i,j}} + \delta_{i,j} - C^{min}_{i,j} - FSC_{i,j}(0)}{\sigma_{i,j}(f+1)}. \tag{2.6}$$

Let's consider that the switch has been functioning for a very long period of time, such that $\sigma_{i,j}$ has a very large value. Therefore, we have:

$$R^c_{i,j} \leq \frac{1}{f+1}, \tag{2.7}$$

or

$$R^m_{i,j} > \frac{f}{f+1}. \tag{2.8}$$

□

Now, with Lemma 2 proved, the dynamics of FSC are used to define the value of the frame size at time cycle $n+1$, $FSC_{i,j}(n+1)$, as:

$$E[FSC_{i,j}(n+1)] = (FSC_{i,j}(n) + f)(1 - R^m_{i,j}) + (FSC_{i,j}(n) - 1)R^m_{i,j}, \quad (2.9)$$

where $E[FSC_{i,j}(n+1)]$ is the expected value of FSC at cycle $n+1$. This equation considers an increment and a decrement of the FSC with probabilities $1 - R^m_{i,j}$ and $R^m_{i,j}$, respectively, at time slot n.

Considering that $FSC_{i,j}(n+2) = FSC_{i,j}(n+1+1)$:

$$E[FSC_{i,j}(n+2)] - E[FSC_{i,j}(n+1)] = f - R^m_{i,j}(f+1). \quad (2.10)$$

According to the definition of stability in the sense of Lyapunov [62], if $E[FSC_{i,j}(n+l+1)] - E[FSC_{i,j}(n+l)] = -\varepsilon < 0$, which indicates that the expected FSC drift is negative, the FSC exhibits an overall downward drift, and will not become unbounded, therefore FSC is stable.

Recalling the value of R_m from Lemma 2, and substituting $R_m = \frac{f+\mu}{f+1}$ in (2.8), where $0 < \mu < 1$, it is clear that:

$$R_m = \frac{f+\mu}{f+1} > \frac{f}{f+1}. \quad (2.11)$$

Considering that $l = 1$ and n can be any service cycle, we substitute (2.11) in (3.4):

$$E[FSC_{i,j}(n+2)] - E[FSC_{i,j}(n+1)] = f - (\frac{f+\mu}{f+1})(f+1), \quad (2.12)$$

which is:

$$E[FSC_{i,j}(n+2)] - E[FSC_{i,j}(n+1)] = -\mu, \quad (2.13)$$

for any cycle n during steady state. This equality shows the stability of FSC of any VOQ. Therefore, a packet switch using RR-AF arbitration under uniform traffic is stable.

□

2.3 Conclusions

The RR-AF scheme use the concept of round-robin selection and adaptable-size frame. The frame size depends on the service received by a queue. This chapter proved that the round-robin scheme with adaptable-size frame arbitration delivers 100% throughput under uniform traffic.

The analytical result can be extended to nonuniform traffic patterns. The result also shows that a buffered crossbar with one-cell crosspoint buffers is sufficient to provide high throughput with the proposed round-robin based arbitration.

This chapter shows that it is possible to provide an arbitration scheme based on round-robin selection for buffered crossbars such that a switch can deliver high throughput under admissible traffic with nonuniform distributions with a small crosspoint buffer size.

CHAPTER 3

ANALYSIS OF A FLOW CONTROL SYSTEM FOR A COMBINED INPUT-CROSSPOINT BUFFERED PACKET SWITCH

3.1 Introduction

Active queue management (AQM) has been a very active research area in recent years. A lot of research work on AQM mechanisms has been done, e.g. random early detection (RED) [22], random exponential marking (REM) [23], proportional-integral (PI) controller [25, 26], adaptive virtual queue (AVQ) [28, 29, 30, 31] and state feedback control (SFC) [32]. The objective of AQM mechanism is to provide early congestion notification to the sources so they can reduce the sending rate to avoid packet loss produced by buffer overflow. The TCP congestion-avoidance control model was proposed by [27].

We can consider a flow control mechanism where the parameter in observation is the queue occupancy. In this case, control theory can be applied to analyze and design queue management schemes. Most of AQM work focuses on supporting congestion management for the transmission control protocol (TCP) flows. Few of the previous works are on the switching applications. This chapter focuses on applying control theory to analyze a flow control mechanism for a CICB switch.

Credit-based flow control is used for avoiding buffer overflow [21]. This is basically a feedback control problem. The feedback is usually based on the amount of buffer space available or in use in the switch. Credit provides precise control over the buffer use, and can stop transmission automatically to avoid buffer overflow. This flow control mechanism has been applied to CICB switches with a negligible transmission delay in switches [42], where the transmission delays are the propagation delays of sending a cell from the input port to a crosspoint, and the flow control information from the crosspoint buffer to the input port. The sum of the transmission delays plus the selection delays at inputs (VOQ selection)

and outputs (crosspoint buffer selection) is called round-trip time. As the buffered crossbar switch can be physically located far from the input ports, actual round trip times (R_0) can be non-negligible. To support non-negligible round-trip times in a buffered-crossbar switch, the crosspoint-buffer size needs to be increased, such that up to R_0 cells can be buffered. Non-negligible round trip delays have been considered in [35, 36, 45, 67], for practical implementations.

However, in a credit-based flow control mechanism, the stability of the switch, as a product of the round-trip time and crosspoint-buffer size, might be difficult to analyze. In this chapter, we use a proportional (P) controller for a flow control mechanism for a CICB switch as a case study. We analyze the relationship between the round-trip time and crosspoint buffer size and the effect on stability [65, 66].

This chapter is organized as follows. Section 3.2 briefly introduces the AQM-based flow control model in a CICB switch and presents the analysis on the relationship between the system stability and crosspoint buffer size as well as round-trip time. Section 3.3 discusses the design of an input shaper to obtain short transient response time of a flow control system. Section 3.4 presents the conclusions.

3.2 Flow Control Mechanism and Stability Analysis

Figure 3.1 shows a buffered crossbar switch with M inputs and outputs. In this switch model, there are M VOQs at each input. A VOQ at input i that stores cells for output j is denoted as $VOQ_{i,j}$. A crosspoint element in the buffered crossbar that connects input port i, where $0 \leq i \leq M-1$, to output port j, where $0 \leq j \leq M-1$, is denoted as $CP_{i,j}$. The buffer at $CP_{i,j}$ is denoted as $CPB_{i,j}$, and it is considered of k-cell size,where $k \geq 1$.

We consider that in a CICB switch each VOQ and its corresponding CPB comprise a closed loop as shown in Figure 3.1. In order to avoid overflow of $CPB_{i,j}$, feedback is needed to inform the $VOQ_{i,j}$ to control the sending data rate. Based on the similar concept

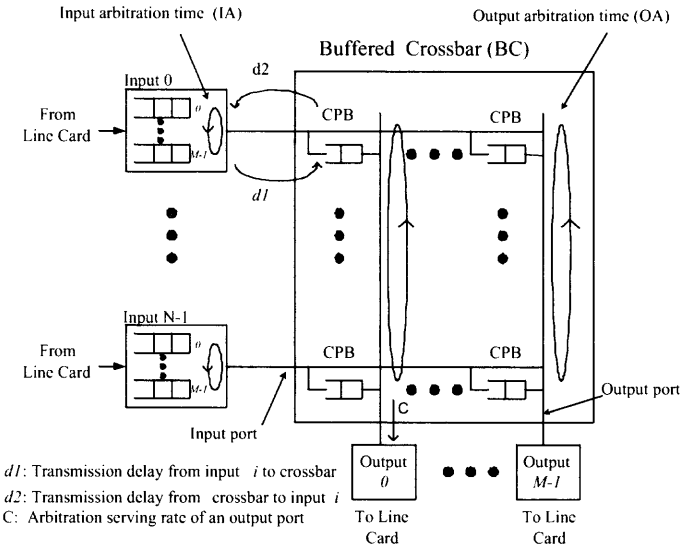

Figure 3.1 Combined input-crosspoint buffered crossbar switch.

of TCP windowing, we use a frame size to control the VOQ's sending rate. The frame size is the amount of packets transferring into the switch fabric [59],[60].

In TCP, the congestion window size, $W(t)$, is increased by one packet every time an acknowledgment is received in a round trip time if no congestion is detected, and is halved upon congestion detection. This additive-increase multiplicative-decrease (AIMD) behavior of TCP has been modeled by the equation (4.7) and (4.8) in [27]:

$$\dot{W}(t) = \frac{1}{R(t)} - \frac{W(t)W(t - R(t))}{2R(t - R(t))} p(t - R(t)), \tag{3.1}$$

where W is the average TCP window size (packets), $R_0(t)$ is the round-trip time (seconds).

In a network topology of N homogeneous TCP sources and one router, the equation for the queue dynamics is given as:

$$\dot{q}(t) = \frac{W(t)}{R(t)} N(t) - C, \tag{3.2}$$

where q is the average queue length (packets), C is link capacity (packets/sec) and N is the load factor. After linearization of (4.7) and (4.8), we have the transfer function of the target plant:

$$P(s) = \frac{\frac{C^2}{2N}}{(s + \frac{2N}{R_0^2 C})(s + \frac{1}{R_0})}. \tag{3.3}$$

To use this analogy in a CICB switch, some parameters need to be adapted. W is the average VOQ frame size, C is the arbitration serving rate of an output port (packets/sec). Since in each closed loop of VOQ and its corresponding CPB there is only one session, so the load factor is N=1. The minimum service rate to provide 100% throughput that one crosspoint buffer receives under uniform traffic is $\frac{C}{M}$, when R_0 can be negligible and CPB size $k \geq 1$. In this chapter, we consider a large R_0 value, where $\frac{k}{R_0}$ is not necessarily 1. The minimum data rate of a flow that can be handled by this switch under uniform traffic is $\frac{C}{M}\frac{k}{R_0}$.

We discuss the feedback control on the following equation. The plant transfer function $P(s)$ is:

$$P(s) = \frac{\frac{C^2 k^2}{2M^2 R_0^2}}{(s + \frac{2M}{R_0 C k})(s + \frac{1}{R_0})}. \tag{3.4}$$

We use P control for the flow control in a CICB switch. Figure 3.2 shows the block diagram of the P control in the VOQ-CPB closed loop. The feedback signal is the regulated output (crosspoint buffer occupancy) multiplied by a gain factor K_p.

The nominal loop transfer function of the proportional controller case is:

$$L(s) = \frac{K_p \frac{C^2 k^2}{2M^2 R_0^2} e^{-sR_0}}{(s + \frac{2M}{R_0 C k})(s + \frac{1}{R_0})}. \tag{3.5}$$

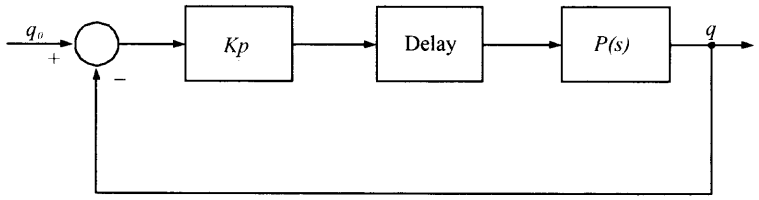

Figure 3.2 Block diagram of P control in a VOQ-CPB closed loop.

We can take the loop's unity-gain crossover frequency as the geometric mean of corner frequency:

$$w_g = \sqrt{\frac{2M}{R_0^2 Ck}} \tag{3.6}$$

and choose K_p to make $|L(jw_g)| = 1$. The goal of the P controller design is to provide a stable closed-loop system. Beyond an acceptable transient response, the system is required to have a margin of safety to be robust to variations in model parameters. The margin is called stability margin (gain margin and phase margin). In this chapter, we consider the phase margin. The phase margin (PM) * is

$$PM = 180 - \arctan w_g R_0 - \arctan \frac{w_g R_0 Ck}{2M} - \frac{180}{\pi} w_g R_0. \tag{3.7}$$

Let's consider the following example.

Example 1. *Consider the following setup in a CICB switch.* C*=3750 packets/s,* M*=32,* R_0*=0.0246 s, and the value of* k *is the crosspoint buffer size in time slots.*

Since C*=3750 packets/s, one time slot is 0.267 ms. For example,* k *is 10-packet long, or* k*=2.67 ms.*

*Phase margin, as a relative stability indicator for both discrete-time and continuous time system, is defined in terms of the system open-loop frequency response [37].

Table 3.1 Phase margin and the ratio of crosspoint buffer size and round-trip time.

k	k/R_0	PM
10	$\frac{1}{10}$	$-54.85°$
20	$\frac{1}{5}$	$-12.43°$
25	$\frac{1}{4}$	$-1.62°$
30	$\frac{1}{3}$	$6.36°$
40	$\frac{2}{5}$	$17.59°$
50	$\frac{1}{2}$	$25.23°$
100	1	$44.19°$

Assume that the round-trip time is fixed. When the crosspoint buffer size increases, the system's phase margin increases as Table 3.1 shows. However, when the crosspoint buffer size decreases to a value where $\frac{k}{R_0} \leq \frac{1}{4}$, the system becomes unstable. †

To support large R_0 in a buffered crossbar switch, the crosspoint-buffer size needs to be large enough, such that up to R_0 cells can be buffered. However, the memory amount that can be allocated in a chip is limited. In this example, when $\frac{k}{R_0} \geq \frac{1}{3}$, the flow control system is stable.

3.3 Design of an Input Shaper

For the purpose of the flow control in a CICB switch, the controller must have an acceptable transient response such that the crosspoint buffer occupancy can converge to a target value q_0 in a short time. A short transient response time is important to the flow control in a CICB switch [38].

†In most cases, a positive phase margin will ensure stability of the closed-loop system [37].

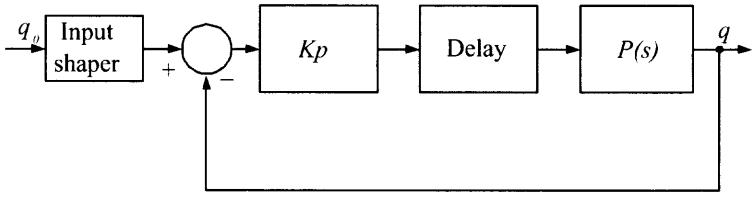

Figure 3.3 Block diagram of P control with input shaper in a VOQ-CPB closed loop.

In order to obtain a short transient response, we add an input shaper to the closed loop system. The input shaper is a feed-forward pole-zero cancelation method. Ideally, the input shaper uses its zeros to cancel the poles of a target system. Therefore, a good performance is obtained [39].

The input shaper might be designed as a step function. In our case, the step function is used to set up the value of q_0 and also the time to apply q_0 into the closed-loop system. Each step function has a step response. We use two step functions to cancel the two-pole system's overshoots. The key point is how to put these two step functions together. We show this combination with the following example.

Example 2. *Consider the following setup in a CICB switch. If* C*=3750 packets/s,* M*=32,* R_0*=0.00801 s, and the value of* k *is 0.00267 s, the ratio* $\frac{k}{R_0}$ *becomes* $1/3$.

The closed-loop system can be regarded as a second-order system by approximation [71]. The transient response is shown in Figure 3.5. Because the system is underdamped, there are some oscillations before the crosspoint buffer occupancy converges to q_0. The maximum overshoot value q_{max} is measured as $q_{max} = 15$, the steady state value of queue occupancy q_{ss} is measured as $q_{ss} = 10$.

With the following equations, we can obtain the step value and step time for each step function.

$$V = \frac{q_{max} - q_{ss}}{q_{ss}} = 0.5 \tag{3.8}$$

The step value for the first step function is:

$$A_1 = \frac{1}{1+V} \tag{3.9}$$

The step value for the second step function is:

$$A_2 = 1 - A_1 \tag{3.10}$$

$$exp(-\frac{\zeta\pi}{\sqrt{1-\zeta^2}}) = V. \tag{3.11}$$

From (3.11), the damping factor ζ is obtained as 0.2154. Since ζ is less than 1, the system is underdamped. Then, the time interval between the first step function and the second step function is obtained.

$$\Delta T = \frac{\pi}{\omega_n\sqrt{1-\zeta^2}} = 0.01573 \tag{3.12}$$

In summary, if the target crosspoint buffer occupancy is q_0, with a design of an input shaper, we first generate a step function whose step value is $q_0 \times A_1$, and its step time is 0. The step value of the second step function is $q_0 \times A_2$ and its step time is ΔT. Now, we get the input shaper. Before being input to the proportional control system as Figure 3.2 shows, q_0 is shaped by the two step functions. That is why we call the two step functions an input shaper. Figure 3.4 shows the block diagram of a P control with input shaper. As Figures 3.5 and 3.6 show, with an input shaper, the transient response time can be improved by almost 40%.

The simulation result for a PI control with an input shaper and its comparison with the PI control without input shaper. Figure 3.7 shows the block diagram of a PI control with an input shaper. As Figures 3.8 and 3.9 show, it is possible to obtain the similar result as the above P control such that the transient response time can be improved by almost 40%.

3.4 Conclusions

This chapter analyzed an AQM-based flow-control mechanism for CICB switches. This chapter also investigated the stability margin by analyzing the relationship between the crosspoint buffer size and the round-trip time. When $\frac{k}{R_0} \geq \frac{1}{3}$ in the experiments, the flow control system is stable. This may provide a guideline on how to choose the crosspoint buffer size under non-negligible round trip times. This chapter shows an improvement of the system's transient response time by almost 40% when adding an input shaper to the closed-loop system.

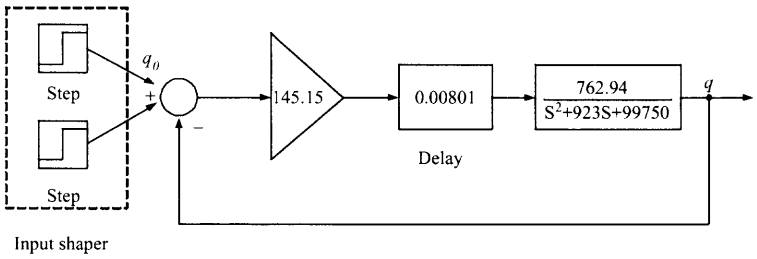

Figure 3.4 Diagram of a P control with input shaper.

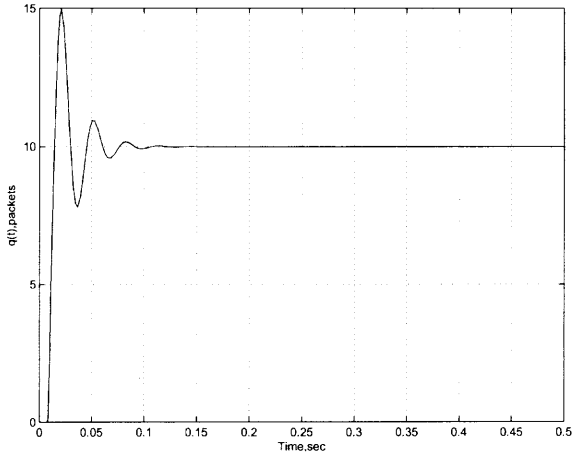

Figure 3.5 Simulation result on a P control without input shaper.

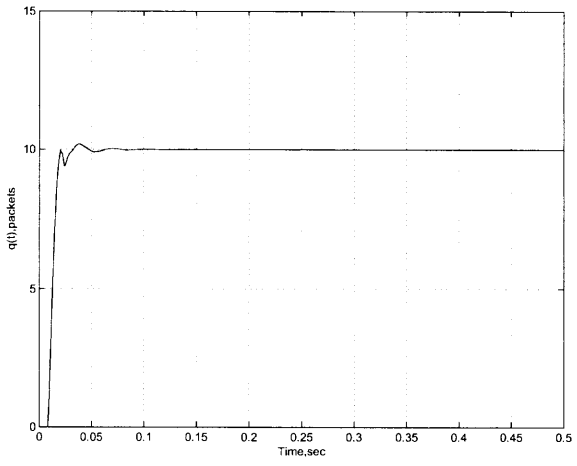

Figure 3.6 Simulation result on a P control with input shaper.

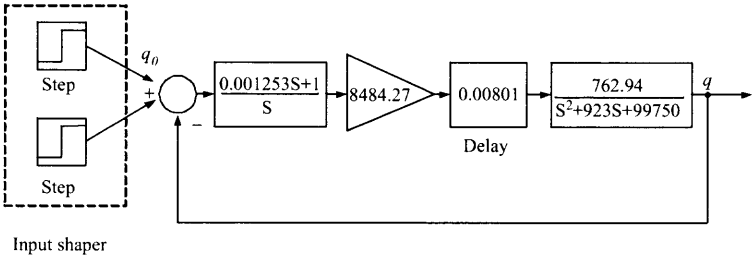

Figure 3.7 Diagram of a PI control with input shaper.

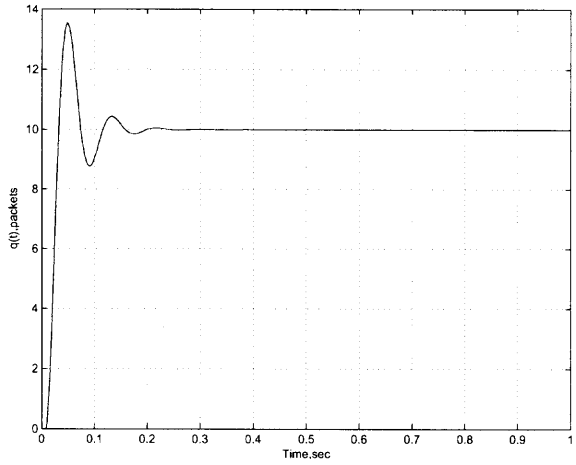

Figure 3.8 Simulation result on a PI control without input shaper.

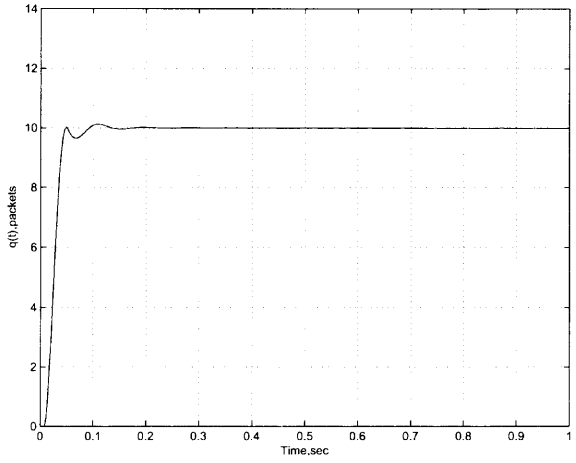

Figure 3.9 Simulation result on a PI control with input shaper.

CHAPTER 4

THROUGHPUT OF A COMBINED INPUT-CROSSPOINT BUFFERED PACKET SWITCH WITHOUT SPEEDUP

4.1 Introduction

Combined input-crosspoint buffered (CICB) switches are known to be a practical alternative to provide high-performance switching and to relax arbitration timing for packet switches with high-speed ports. CICB switches use time efficiently because input and output port selections are performed separately, and the working speed of the memory in a CICB switch is as relaxed as that of the memory in IB switches.

Buffered crossbar switches have simpler scheduling algorithms than bufferless crossbar switches to achieve equal or better performance. In an bufferless crossbar switch, the scheduler must find a matching between inputs and outputs [40, 41].

CICB switches with FIFOs as input queues have been used to reduce the crosspoint-buffer size and to reduce the packet loss ratio [1, 6]. However, a CICB switch with input FIFOs may have the throughput limited by the head-of-line blocking phenomena. CICB switches with VOQs at the inputs can provide 100% throughput under uniform traffic, using weightless [42] and weighted [12, 46] arbitration schemes. A VOQ-CICB switch is referred as a CICB switch for the sake of brevity in the remainder of this chapter.

CICB switches provide good throughput for admissible uniform traffic with simple algorithms. Until recently, there are a few of analytical results on the high throughput to explain or confirm the observation made by simulatons. In [46], it was proved that a buffered crossbar switch with no speedup can achieve 100% throughput under uniform traffic. In [47, 80], it was proved that a buffered crossbar switch with a speedup of two can mimic a first–come first–serve output buffered (FCFS-OB) switch with any arrival traffic pattern. In [49], it was proved that with a weighted round-robin scheduler, a buffered

crossbar can achieve 100% throughput and can mimic an OB switch with a speedup of two for admissible traffic. Notice that [47] and [49] require the buffered crossbar switches work with a speedup of 2, and the switches have output queues. Therefore the buffered crossbar switch becomes a combined input-crosspoint-output buffered (CICOB) switch.

A question arises: what is the throughput of a CICB switch with VOQs under any admissible traffic patterns while using no speedup [68]? As a response to this question, herein it is proved that CICB switches with one-cell crosspoint buffers and no speedup can provide 100% throughput under admissible traffic that complies with the strong law of large numbers (SLLN) [44]. The intuition of this result lies on the knowledge that CICB switches provide higher performance than IB switches [42, 46], and that IB switches can provide 100% throughput under admissible traffic with no speedup [44], although with a high-complexity matching scheme.

Previous research [18, 62] have shown the stability analysis and proofs are based on the consideration of the queue length evolution, and finding or building up a Lyapunov function, by which to prove the stability of the scheduling algorithm. In [18], a second order Lyapunov function, such as $L^T(n)L(n)$ and $W^T(n)W(n)$ were used to show stability of maximum weight matching (MWM) algorithms, such as longest queue first (LQF) and oldest cell first (OCF). Stability analysis and proof for maximal size matching (MSM) algorithm [62] can be very complex since maximal matchings can not guarantee a result as MWM does, and therefore, to find a suitable Lyapunov function.

Another approach is using fluid model in the analysis of switch stability. A fluid model technique was used to prove that 100% throughput on IB switches by MWM scheme and also 100% throughput on CIOB switches by MSM scheme with a speedup of 2 [44].

In this chapter, it is assumed that the common practices of having incoming variable-size packets, which are segmented into fixed-size cells at the ingress side of a switch and reassembled at the egress side before leaving the switch, and of using a crossbar, are also used.

This chapter is organized as follows. Section 4.2 introduces the switch and fluid models, and some preliminary definitions. Section 4.3 presents the throughput analysis of a CICB switch. Section 4.4 shows examples of input and output arbitration schemes that share the same properties required by the fluid model. Section 4.5 shows a performance evaluation. Section 4.6 presents our conclusions.

4.2 CICB Switch and Fluid Model

Consider a CICB switch with N inputs and outputs. There are N VOQs at each input. A VOQ at input i that stores cells for output j is denoted as $VOQ_{i,j}$. A crosspoint element in the buffered crossbar that connects input port i, where $0 \leq i \leq N-1$, to output port j, where $0 \leq j \leq N-1$, is denoted as $CP_{i,j}$. The buffer at $CP_{i,j}$ is denoted as $CPB_{i,j}$, and it is considered of one-cell size. Therefore, the transmission and arbitration delays are considered negligible, without loss of generality. A large CPB size will have higher throughput under this condition, and it would allow non-negligible transmission delays. $CPB_{i,j}^{Busy}$ denotes an occupied CPB.

The occupancy of $VOQ_{i,j}$ at up to time slot n is denoted as $Z_{i,j}(n)$. The cumulative number of packets that have arrived at $VOQ_{i,j}$ by time slot n is denoted as $A_{i,j}(n)$, and the cumulative number of packets that have departed from $VOQ_{i,j}$ by time slot n is denoted as $D_{i,j}(n)$.

In a CICB switch, the input arbitration at input i selects a cell from a non-empty VOQ, whose corresponding CPB is available, to be forwarded to the buffered crossbar. At the same time, the output arbitration selects a cell from an CPB among all those for output j to leave the buffered crossbar. It is considered that the output arbitration can adopt either a distributed or a centralized approach.

A fluid model [44] is used to analyze the properties of the VOQs in a CICB switch with no speedup and look at the stability property of this switch under a traffic model with the restrictions of being admissible and where the cell arrivals follow the strong law of large

numbers:

$$\lim_{n\to\infty} \frac{A_{i,j}(n)}{n} = \lambda_{i,j}, \tag{4.1}$$

where $\lambda_{i,j}$ is the average arrival rate at $VOQ_{i,j}$.

An input arbitration uses scheme m, such that the selected VOQs can be expressed by matrix $\pi^m_{i,j}(n) \in \Pi$ at time slot n. For π, let T^m_π be the cumulative amount of time that a combination π has been used by time slot n. Therefore, $D_{i,j}(n)$ is the number of departures from $VOQ_{i,j}(n)$ up to time slot n, where $D_{i,j}(0) = 0$ is defined.

Definition 4. *If* $\lim_{n\to\infty} \frac{D_{i,j}(n)}{n} = \lambda_{i,j}$, *the switch is said to be rate stable.*

It has been proved that a switch is rate stable if the corresponding fluid model is weakly stable [44].

For $n \geq 0$, the switch dynamics are represented as:

$$Z_{i,j}(n) = Z_{i,j}(0) + A_{i,j}(n) - D_{i,j}(n), \tag{4.2}$$

and

$$\sum_{\pi\in\Pi} T^m_\pi(n) = n. \tag{4.3}$$

where $T^m_\pi(.)$ is non-decreasing.

A switch under traffic that complies with SLLN can be represented through a fluid model [44].

Definition 5. *The fluid model of a switch is said to be weakly stable if for every fluid model solution* (D, T, Z) *with* $Z(0) = 0$, $Z(t) = 0$ *for almost every* $t \geq 0$ *[44].*

The dynamics of the fluid model of the switch can be expressed as

$$Z_{i,j}(t) = Z_{i,j}(0) + \lambda_{i,j}t - D_{i,j}(t), \tag{4.4}$$

and

$$\dot{D}_{i,j}(t) = \sum_{\pi \in \Pi} \pi_{i,j} \dot{T}_{\pi}^{m}(t), if Z_{i,j}(t) > 0, \tag{4.5}$$

where $T_{\pi}^{m}(.)$ is non-decreasing and $\sum_{\pi \in \Pi} T_{\pi}^{m}(t) = t$. Here, $\dot{g}(t)$ is the derivative of a function $g(t)$ at t.

From the switch dynamics in (4.2) to the fluid model in (4.4), it is a complex step associated with limiting procedure to obtain the fluid limits, which are proved to be the fluid model solutions. The fluid limit of a switch is Lipschitz continuous and therefore is absolutely continuous, which determines that the fluid model is continuous. That is why the fluid model approach is considered to change the analysis of a discrete stochastic system into the analysis of a model obeying deterministic differential equations.

Fact 1. *(Lemma 1 in [44]) Let $f : [0, \infty) \rightarrow [0, \infty)$ be an absolutely continuous function with $f(0) = 0$. Assume that $\dot{f}(t) \leq 0$ for almost every t such that $f(t) > 0$ and f is differentiable at t. Then $f(t) = 0$ for almost every $t \geq 0$.*

By the fluid behavior of the VOQs', for a weakly stable switch there must exist an $f(t)$, where $f(t) = 0$ implies $Z(t) = 0$ for every $t > 0$, and where $f(0) = 0$ implies $Z(0) = 0$.

4.3 Throughput Analysis of a CICB Switch

Theorem 2. *A CICB, with a VOQ structure at the inputs and using no speedup, provides 100% throughput under admissible traffic.*

Proof. The CICB switch is analyzed in two separated parts. The first part is concerned with the inputs, and the second part with the buffered crossbar.

As in [44], let's define

$$C_{i,j}(t) = L_i(t) + M_j(t), \tag{4.6}$$

where

$$L_i(t) = \sum_k Z_{i,k}(t)$$

denotes the total amount of fluid queued at the input i at time t, and

$$M_j(t) = \sum_k Z_{k,j}(t)$$

denotes the total amount of fluid destined for output j at time t. In other words, $C_{i,j}$ denotes the total amount of fluid at input i and the fluid destined to output j.

Since input and output arbitrations work separately in a CICB switch, if cell c is dispatched from $VOQ_{i,j}$ and is stored at $CPB_{i,j}$, then $L_i(t)$ and $M_j(t)$ decrease by one. Therefore $C_{i,j}(t)$ is reduced by two in a single time slot.

Compared with CICB switch, CIOB switches do not use buffers in the crosspoints. Since the speedup is 2 in CIOB switch, there are two scheduling phases in one time slot (as shown in Figure 4.1). Within one scheduling phase, $C_{ij}(t)$ can only be reduced by one. The reason is that if a match between input i and output j is found in the input arbitration, input i sends one cell to output j, because there is no buffer in the crosspoint, the coming out cell has no place to stay in and it can only go to the output following the matching path in the output arbitration, by which all of $L_i(t) + M_j(t)$ together can only reduce one. So with speedup of 2, in one time slot, the two scheduling phases together can make $L_i(t) + M_j(t)$ reduce by 2. For the purpose of comparison, Figure 4.2 presents the scheduling of a CICB switch.

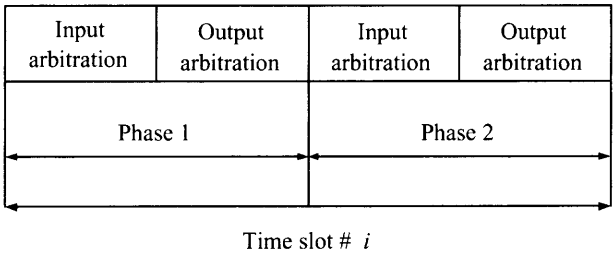

Figure 4.1 Scheduling in a CIOB crossbar switch.

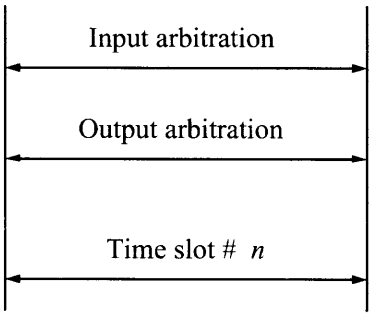

Figure 4.2 Scheduling in a one-cell buffered crossbar switch.

In a similar way as in [44], let Q be a $N \times N$ matrix with each entry being 1. Then

$$C(t) = QZ(t) + Z(t)Q, t \geq 0 \tag{4.7}$$

where $C_{i,j}$ is an element of $C(t)$.

$f(t)$ is defined as:

$$f(t) = \langle Z(t), C(t) \rangle = \sum_{i,j} Z_{i,j}(t) C_{i,j}(t). \tag{4.8}$$

It follows that $f(t) \geq 0$ for $t \geq 0$ and $f(0) = 0$. It is easy to see that $f(t) = 0$ implies $Z(t) = 0$. Next, it is shown that $f(t) > 0$ implies $\dot{f}(t) \leq 0$ for almost every t.

In this case, from [44], it follows that

$$\dot{f}(t) = 2 \sum_{i,j} Z_{i,j}(t) \dot{C}_{i,j}(t). \tag{4.9}$$

Therefore, $\dot{f}(t) \leq 0$ if and only if $\dot{C}_{i,j}(t) < 0$. As mentioned above,

$$\dot{C}_{i,j}(t) = \sum_{k} \lambda_{i,k} + \sum_{k} \lambda_{k,j} - 2 \tag{4.10}$$

where

$$\sum_{j} \lambda_{i,j} \leq 1$$

and

$$\sum_i \lambda_{i,j} \leq 1$$

makes $\dot{C}_{i,j}(t) \leq 0$. Therefore, from (4.9) and (4.10), $\dot{f}(t) \leq 0$ whenever $f(t) > 0$.

The results state that the existence of $f(t)$ and Fact 1 establish that the fluid model of a CICB switch with one-cell crosspoint buffer is weakly stable as long as no input is inhibited from sending a cell to the buffered crossbar in a time slot. Then, it remains to complete the proof of Theorem 2 with the following lemmas.

In (4.5), the arbitration scheme m selects a VOQ such that an CPB at j receives one cell, as expressed by (4.6). The following lemma is used for the non-inhibition of an input arbiter:

Lemma 3. *At any time slot, input i has at least an available $CPB_{i,j}$ under admissible traffic such that inhibition is avoided.*

Proof. Lemma 3 can be rephrased in terms of the output arbitration scheme, as follows:

Lemma 4. *There exists an output arbitration scheme such that the selection result causes $\sum_j CPB_{i,j}^{Busy} < N$ for admissible traffic, at any time slot.*

Consider the following propositions, presented in [50, 57, 75, 76]. *Von Neumann proposition: if a matrix $B = (B_{i,j})$ is doubly substochastic, then there exists a doubly stochastic matrix $\bar{B}$ such that $B_{i,j} < \bar{B}_{i,j}\ \forall\ i, j$.*

Birkhoff's proposition: for a doubly stochastic matrix $\bar{B}$, there exists a set of positive numbers ϕ_k and permutation matrices P_k, where $1 \leq k \leq K$, such that $\bar{B} = \sum_k \phi_k P_k$. Let e be the column vector with all its elements being 1. As $\bar{B}$ is doubly stochastic $e = \bar{B}e = \sum_k \phi_k(Pe) = (\sum_k \phi_k)e$, making $\sum_k \phi_k = 1$.

The occupancy of the one-cell crosspoint buffers in the buffered crossbar can be represented by a matrix

$$CPB^{Busy} = (CPB_{i,j}^{Busy}) \tag{4.11}$$

such that

$$\sum_j CPB_{i,j}^{Busy} \leq N \tag{4.12}$$

and

$$\sum_i CPB_{i,j}^{Busy} \leq N. \tag{4.13}$$

Normalizing CPB^{Busy} with respect to N, the matrix is doubly substochastic.

Therefore, CPB^{Busy} can be represented as doubly stochastic $\overline{CPB}^{Busy}$, such that there exist permutation matrices that indicate which $CPB_{i,j}^{Busy}$ is served at j in a time slot.

Therefore, the output arbitration scheme must select a set of CPBs such that, for a given P_k

$$\sum_j P_{i,j} > 0, \tag{4.14}$$

and therefore

$$\sum_j CPB_{i,j}^{Busy} < N \tag{4.15}$$

after the output arbitration. By using the permutation matrices as the set of CPBs that are selected by the output arbitration, input i has at least one CPB available at any time slot.

Furthermore, because $K \leq N^2 - 2N + 2$ [57], the smallest switch size of $N = 2$ has $K \geq 1$. Therefore, this result holds for all N values. As the permutations correspond to a time slot, the unserved cells are held by the CPBs for the following time slot.

Since there exists an output arbitration scheme that allows inputs to be uninhibited, then Lemma 4, and consequently, Lemma 3 are proved. □

As Lemmas 3 and 4 are true, then Theorem 2 is proved. □

Figure 4.3 shows an example of a decomposed matrix for a 4×4 switch. Figure 4.3.a shows a matrix CPB^{Busy} is doubly substochastic according to Equation (4.12) and (4.13). Based on Von Neumann proposition, the normalized matrix CPB^{Busy} can be converted to a

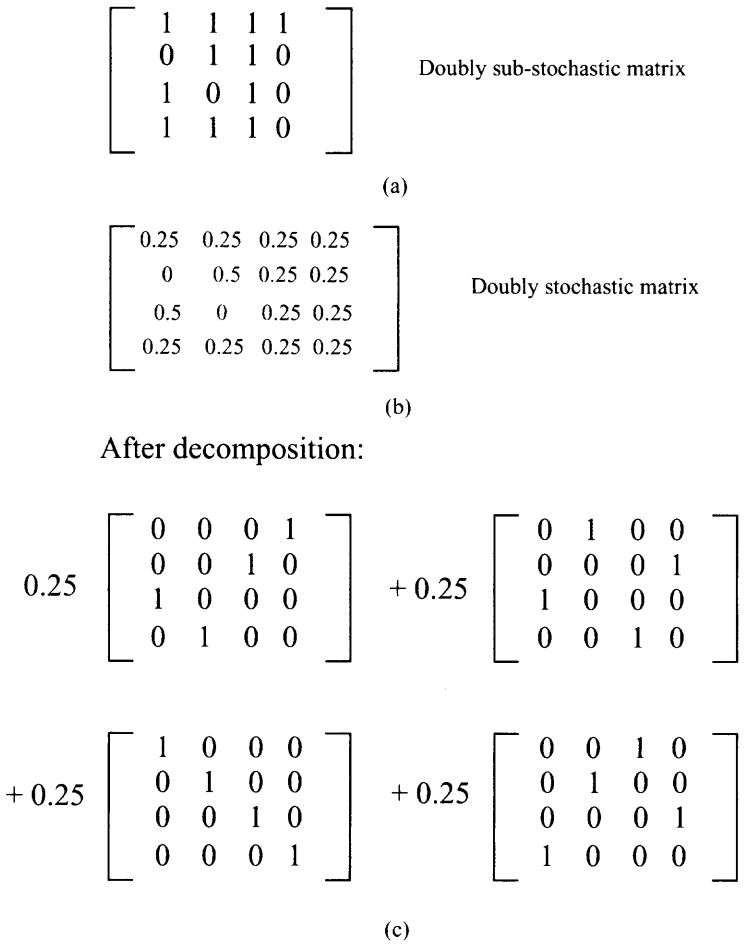

Figure 4.3 Example of a decomposed matrix for a 4×4 switch.

doubly stochastic matrix. Figure 4.3.b shows the doubly stochastic matrix. With Birkhoff's proposition, the decomposed matrix is obtained in Figure 4.3.c. After the decomposition, there are four permutation matrices, which indicate that each row (input) has only one occupied CPB, therefore input inhibition is avoided.

However, the set of permutation matrices may not be enough to keep all outputs active, which can be used to simplify the input arbitration scheme. As an example of an output arbitration scheme that grants at least one CPB from input i and one CPB from output j, the following scheme is introduced, called longest column occupancy (LCO), which is based on the output arbitration presented in [58].

4.4 Arbitration Scheme for 100% Throughput

Input Arbitration: a distributed input arbitration scheme, longest queue first (LQF), which can differentiate among flows that require extensive service, and that can be applied at any input port independently.

Output Arbitration: Since the output arbiters can be placed in-chip at the buffered crossbar, the LCO arbitration scheme as a centralized algorithm, as presented in Section 4.3. By using LCO, an output arbiter selects an input for each output of the buffered crossbar. LCO uses two steps:

Step 1: Select an output $\{j \mid max \sum_i CPB_{i,j}^{Busy}\}$ and an input $\{i \mid max \sum_j CPB_{i,j}^{Busy}$ $\}$. Set i and j as reserved and perform Step 1 with unreserved i, j pairs until no more can be found (i.e., the number of unreserved inputs or outputs becomes zero, or else, when the remaining occupied CPBs belong to reserved inputs or outputs). Then go to step 2.

Step 2: If there are unreserved outputs where at least one CPB is occupied, select an CPB^{Busy} arbitrarily from each unreserved output.

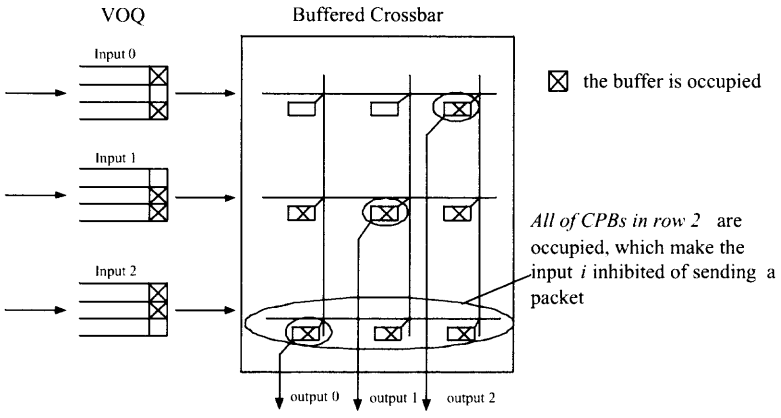

Figure 4.4 Illustration on the LCO scheme.

It is expected that Step 1 be enough to complete the selection of crosspoints equivalent to a doubly substochastic matrix. However, Step 2 may be required in some occasions. Nevertheless, these two steps are used for completeness.

Figure 4.4 shows that by using LCO scheme, occupied CPBs in different rows are selected such that no row can have all its CPBs being occupied at any time slot, no input can be inhibited.

Figure 4.5 shows an example of the selection process that LCO performs in a 4×4 buffered crossbar. In this matrix, rows represent the inputs and the columns represent the outputs. In Figure 4.5.a, the initial matrix have busy and idle crosspoints. Here, the column and row with larger number of busy crosspoint buffers are selected. Ties are broken by arbitrary selections. In Figure 4.5.b, the second largest column is selected. The row selection considers only those unselected and busy crosspoint buffers as shown. Figures 4.5.c and 4.5.d show the selection of the last two columns. The selected crosspoint buffers are marked with an X.

LCO maximizes the number of active outputs and therefore, relaxes the requirements for an input arbitration scheme. This scheme has a computation complexity of $O(N^2)$.

Here, we show why the most critical internal buffer first (MCBF) scheme cannot guarantee that a crosspoint buffer is available for any input at any time slot [58]. This scheme does not consider the state of a single crosspoint buffer, but rather the set of crosspoint buffers in the buffered crossbar.

In MCBF, each distributed output arbiter independently selects the crosspoint from the longest input occupancy as specified by the longest buffer first (LBF) output arbitration. Therefore, if an input has high load, the associated crosspoints could be selected during the same time slot, while leaving other inputs unserviced. Figure 4.6 shows an example of the selection process that the decentralized MCBF performs in a 4×4 buffered crossbar, which is represented as a matrix. In this matrix, rows represent the inputs and the columns represent the outputs. A CPB with a cell is represented by 1 (busy), and 0 (idle), otherwise.

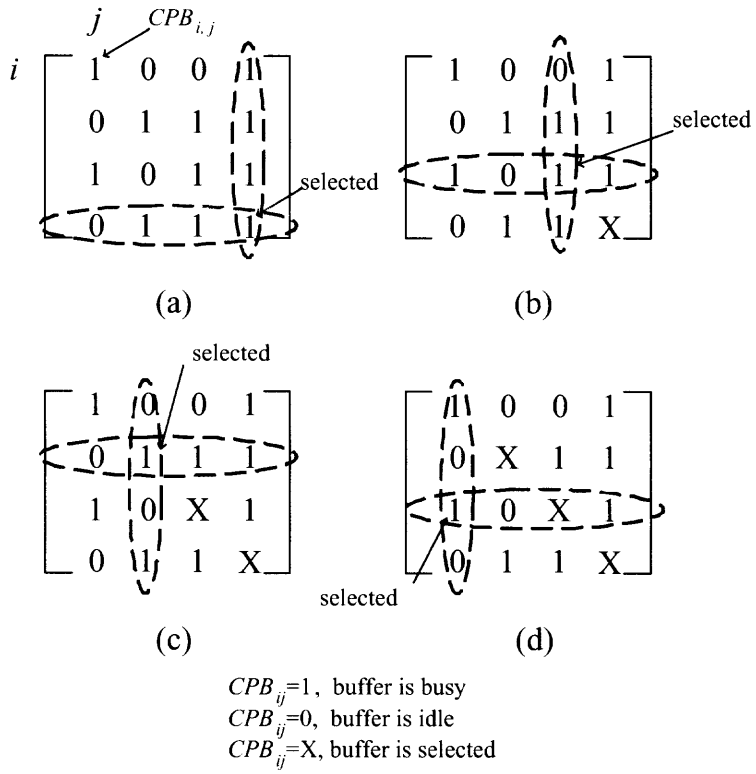

Figure 4.5 Example of selection by performing LCO in a 4×4 switch.

Figure 4.6.a shows the state of CPBs as busy and idle crosspoints. Figure 4.6.b shows the input occupancy seen by the output arbiters per CPB. For example, $CPB_{0,1}$ is 2 as there are only two cells from input 0 for all outputs. An idle CPB is indicated by a zero as it is ignored by the output arbiter. This figure also shows that this example has all CPBs from inputs 1 and 3 with the longest input occupancy, and the output arbiters, using the same selection policy, select all CPBs from input 3. Therefore, Figure 4.6.c shows that $CPB_{3,0}$ to $CPB_{3,3}$ are selected, marked with an X. These independent arbiters select CPBs from input 3 and leave input 1 without an available CPB in the next time slot. Therefore, the distributed MCBF scheme cannot guarantee having an available CPB at any time slot.

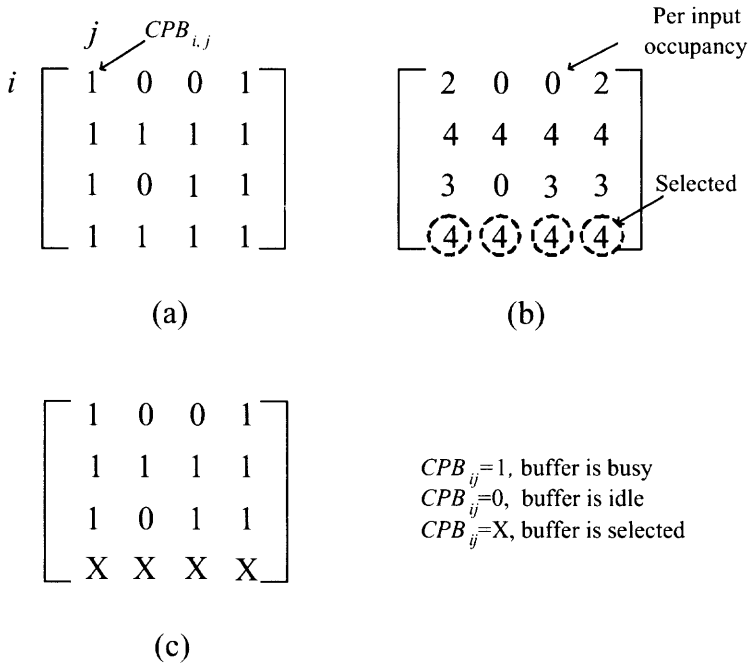

Figure 4.6 A counter example of crosspoint-buffer selection by MCBF in a 4×4 switch.

4.5 Simulation Study of LQF+LCO

As an example of arbitration schemes for achieving 100% throughput under admissible traffic, the LQF selection is used as input arbitration, and LCO selection is used as output arbitration in a CICB switch with a crosspoint buffer size of one cell. LCO follows the selection process as described in Section 4.3. The performance of the CICB switch using LQF and LCO is being compared with an output buffered (OB) switch.

It is shown that no performance degradation occurs for a CICB under uniform traffic. It is also considered that CICB under unbalanced, diagonal, and power of two (PO2) traffic patterns, where all these have nonuniform distributions at different degrees.

In the following, it is shown that the obtained graphs for all different traffic patterns present similar curves, although with different absolutes delay values, which are specially noticeable at loads closes to 0.99. Although seemingly repetitive, these graphs have the

objective to show the way the CICB follows the performance, in terms of the average cell delay, of an OB switch.

4.5.1 Uniform Traffic

Figure 4.7 shows the average cell delay of the OB and CICB switch. This average delay is close to that of an OB switch, as has been previously known for several distributed arbitration schemes in CICB switches (round-robin based arbitration).

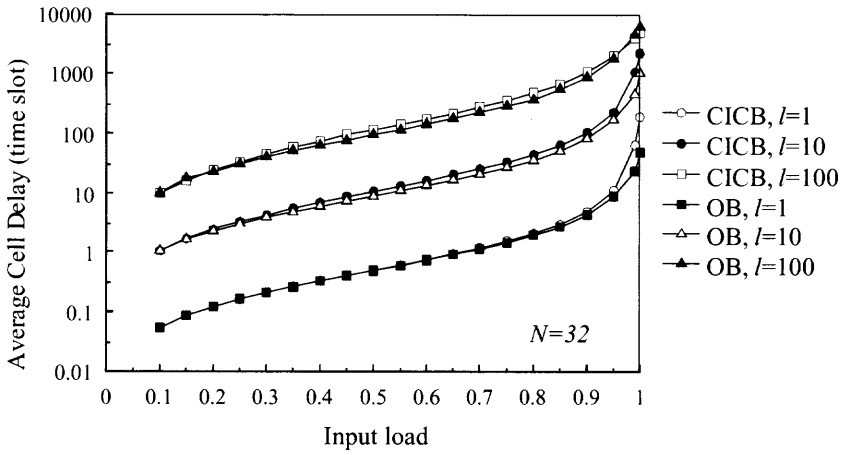

Figure 4.7 Average delay of a 32×32 switch under uniform traffic.

4.5.2 Nonuniform Traffic: Unbalanced

The unbalanced traffic load can be represented by matrix $\bar{\rho}$ as:

$$\bar{\rho} = \rho \begin{pmatrix} w + \frac{1-w}{N} & \cdots & \frac{1-w}{N} \\ \vdots & \ddots & \vdots \\ \frac{1-w}{N} & \cdots & w + \frac{1-w}{N} \end{pmatrix}$$

where ρ in the input load. This traffic model has a fraction of the input load directed to a single output (i.e., the output with the same index as the input) and the rest is informly distributed among all outputs.

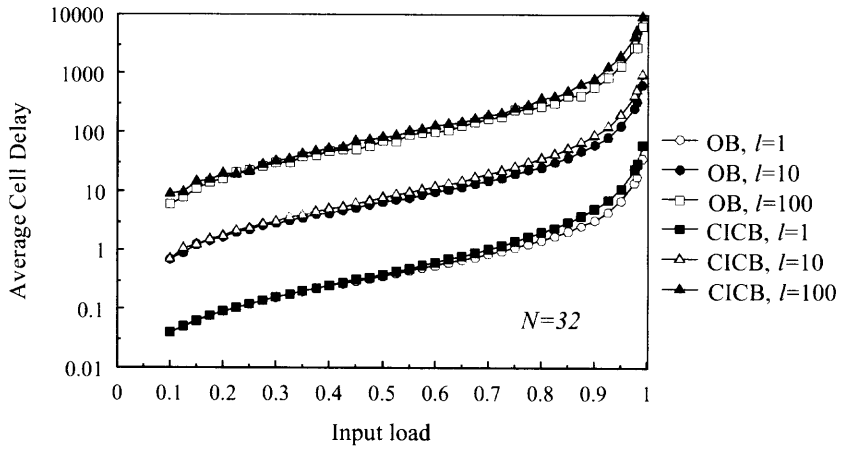

Figure 4.8 Average delay of a 32×32 switch under unbalanced traffic with $w = 0.5$.

4.5.3 Nonuniform Traffic: Diagonal

The diagonal traffic can be represented as $d\rho(i,j) = d\rho$ for $i = j$, $(1-d)\rho$ for $j = (i+1) \bmod N$, or by the matrix $\bar{\rho}$ as:

$$\bar{\rho} = \rho \begin{pmatrix} d & (1-d) & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & & \\ (1-d) & 0 & \dots & 0 & d \end{pmatrix}$$

This traffic model presents load distributions among two outputs per each input. The distributions are given by the diagonal degree probability, d.

Figure 4.9 shows the performance of OB and CICB switches for $d = 0.75$. The CICB switch has an average delay close to that of the OB switch. The figure presents a similar behavior as under uniform traffic. However, there is a small gap between OB and CICB for large input loads, where CICB has slightly higher delay. This figure shows that even when the input load approaches 1.0 the delay CICB continues to follow the delay of the OB switch, therefore, showing 100% throughput.

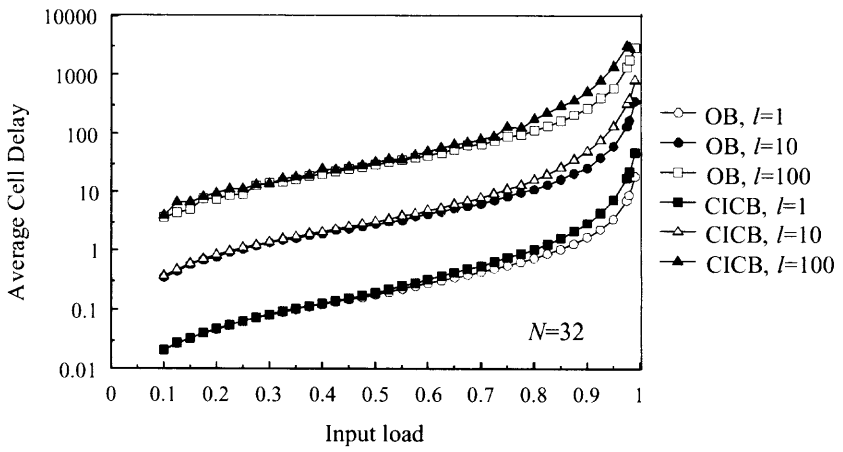

Figure 4.9 Average delay of a 32×32 switch under diagonal traffic.

4.5.4 Nonuniform Traffic: Power of Two (PO2)

In addition, the OB and CICB switches were simulated under power-of-two (PO2) traffic [?] for 30×30 switches. The PO2 traffic model can be represented by matrix $\bar{\rho}$ as:

$$\bar{\rho} = \rho \begin{pmatrix} \frac{1}{2^1} & \cdots & \frac{1}{2^N} \\ \vdots & \ddots & \vdots \\ \frac{1}{2^N} & \cdots & \frac{1}{2^{N-1}} \end{pmatrix}$$

This traffic model presents a large nonuniform distribution among all inputs and outputs. The distribution difference in an input changes along all N possible destinations.

Figure 4.10 the average cell delay of the OB and CICB switches under PO2 traffic. The average delay of the CICB switch also follows the average delay of the OB switch as in the other traffic patterns. This shows that the throughput of the CICB switch approaches to 100%.

4.6 Conclusions

In this chapter, it is shown that a combined input-crosspoint buffered, CICB, packet switch can provide 100% throughput with no speedup and under admissible traffic when VOQs

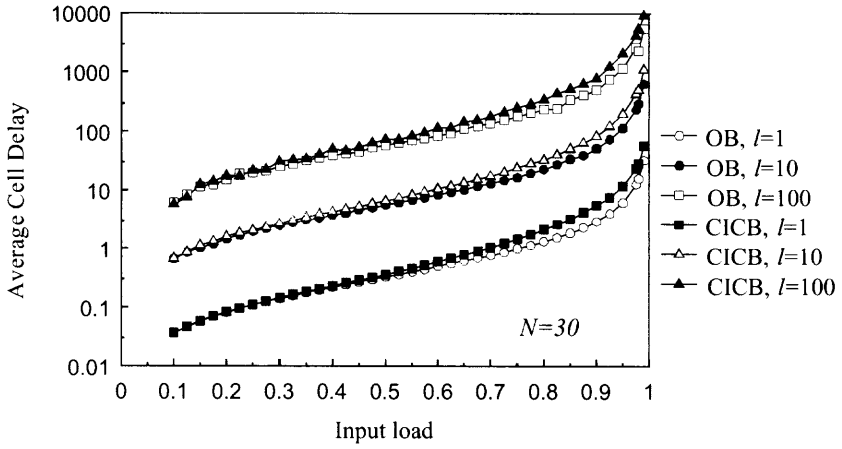

Figure 4.10 Average delay of a 30×30 switch under PO2 traffic.

are used at the inputs. This result is independent of the switch size, and it requires that the admissible input load follows the strong law of large numbers. The fact that input and output arbitrations in a CICB switch are performed separately allows us to analyze the queues at the inputs of the CICB switch while assuming an output arbitration at the buffered crossbar that keeps inputs uninhibited, and by analyzing the buffered crossbar while assuming that an input arbitration can select any VOQ that has a crosspoint buffer available. The proposed output arbitration scheme can avoid input inhibition and also keep outputs active.

It is also presented a simulation study of a hybrid set of arbitration schemes, LQF, which performs in a distributed setting at each input, and LCO, which performs in a centralized setting by considering the location of output arbiters closer to each other and to the crosspoint buffers. Although LCO consumes time to perform a suitable selection of crosspoints, it supports the existence of arbitration schemes that avoid input inhibition. The stability presented by LCO comes at the cost of having a complex algorithm where the timing relaxation may be lost. However, it provides a simple example of a weighted arbitration scheme equivalent to performing matrix decomposition.

CHAPTER 5

FRAMED ROUND-ROBIN ARBITRATION WITH EXPLICIT FEEDBACK CONTROL FOR COMBINED INPUT-CROSSPOINT BUFFERED PACKET SWITCHES

5.1 Introduction

Schemes based on round-robin selection have been shown to provide a high degree of fairness. Furthermore, it has been shown that a CICB switch using one-cell crosspoint buffers (CIXB-1), a round-robin arbitration (RR) scheme for input and output arbitration, and a credit-based flow control provides 100% throughput for uniform traffic [42]. In a CICB switch, the input arbiter selects a VOQ if there is at least a single eligible VOQ. The eligibility of VOQs is determined by the flow control mechanism and packet existence. In CIXB-1, the selection of a crosspoint per output is performed in a round-robin fashion, where only non-empty crosspint buffers are considered.

In order to improve the throughput of CIXB-1 for non-uniform traffic, a frame-based round-robin scheme with adaptable frame size, RR-AF, was proposed [60]. This scheme sends cells from a VOQ to a crosspoint buffer (or from a crosspoint buffer to the output) in a back-to-back fashion, where the number of cells sent continuously depends on the frame size. In this scheme, the frame size increases by a constant number, independently of the actual size needed, each time a frame is completely served. The frame size decreases each time a VOQ misses an opportunity to be served. This scheme, although it provides high performance, however, it is difficult to analyze.

To give a better insight of the feedback nature of arbitration schemes for CICB switches, this chapter introduces a frame-based round-robin with explicit feedback control (FRE) arbitration scheme for CICB packet switches. This scheme adopts a frame-based arbitration scheme that dynamically sets the frame size according to the input load and

to the number of cells accumulated in the input queues. The stability of this scheme is analyzed by using control theory and the switching performance is evaluated by computer simulations. The scheme is the combination of FRE as the input arbitration scheme with other weightless arbitration schemes as output arbitrations. It is shown that FRE provides high throughput under several admissible traffic patterns using a CICB switch with one-cell crosspoint buffers.

This chapter is organized as follows. Section 5.2 describes the switch architecture, introduces several notations used in this chapter, and shows the theoretic control analysis on the proposed scheme. Section 5.3 describes the arbitration algorithm in a weightless cell-based approach. Section 5.4 evaluates the performance on a 32×32 CICB switch through computer simulations. Section 5.5 presents the conclusions.

5.2 CICB Switch Model

The switch model considered in this section is the same as that presented in Chapter 2. The buffered crossbar switch has N inputs and outputs. In this switch model, there are N VOQs at each input. A VOQ at input i that stores cells for output j is denoted as $VOQ_{i,j}$. A crosspoint element in the buffered crossbar that connects input port i, where $0 \leq i \leq N-1$, to output port j, where $0 \leq j \leq N-1$, is denoted as $CP_{i,j}$. The buffer at $CP_{i,j}$ is denoted as $CPB_{i,j}$, and it is considered of k-cell size. In this chapter $k = 1$.

Different from previous approaches, the design of the FRE scheme is started by defining explicitly the feedback system needed to control the service rate to a VOQ. In a CICB switch, each VOQ and its corresponding service rate comprise a closed loop. The VOQ's sending rate is controlled by a frame size. The frame size is the number of packets transfered into the buffered crossbar [59, 60].

A fluid model system is used to analyze the input arbitration scheme in a CICB switch by making an analogy to a level-control system for a fluid container and represent the input arbiters as the controller of an actuator system that drains fluid from the container.

5.2.1 Controlling the Service Rate by Explicit Feedback

Let's consider that the minimum unit of data can be divided into an infinitesimal amount and packetization is not strictly required. Therefore, the switching of data can be performed at any part of a packet; This is, instead of switching packets by their boundaries, it is assumed that fluid data can be switched at any time and that the packets can be re-assembled at the outputs.

Consider a fluid container with a inflow rate V_i, the fluid level h is required to be kept constant by changing the fluid outflow rate through changing the drain area [84]. The inflow rate V_i is used to emulate the input traffic rate in a switch and V_o is the outflow rate, which is the serving rate provided by the arbitration scheme.

The equation governing the change in fluid level is that the rate of change of fluid level is equal to the inflow rate minus the outflow rate.

$$\frac{dh(t)}{dt} = V_i(t) - V_o(t) \tag{5.1}$$

The Laplace transform is derived:

$$H(s) = \frac{V_i(s) - V_o(s)}{s} \tag{5.2}$$

Therefore, the plant is a first-order system.

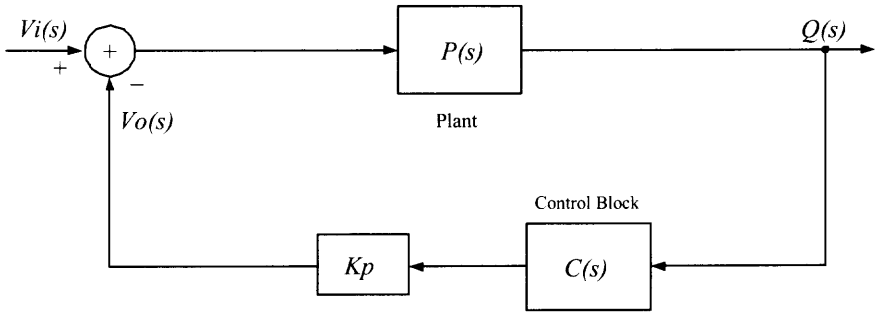

Figure 5.1 Block diagram of a feedback control system.

Figure 5.1 shows the block diagram of a fluid system with a feedback control that makes the outflow rate follow the inflow rate to keep the fluid level from raising indefinitely. We call it fluid control system.

It is assumed that the container's height can be large enough to avoid overflow (this is the analogy of a queue with large capacity) and to allow measuring the differences of fluid accumulation. If the inflow rate is greater than the outflow rate, the feedback controller is expected to increase the outflow rate in order to bring down the level. The integral plant $P(s)$ reflects the fact that the level depends on the difference between the inflow and outflow rate.

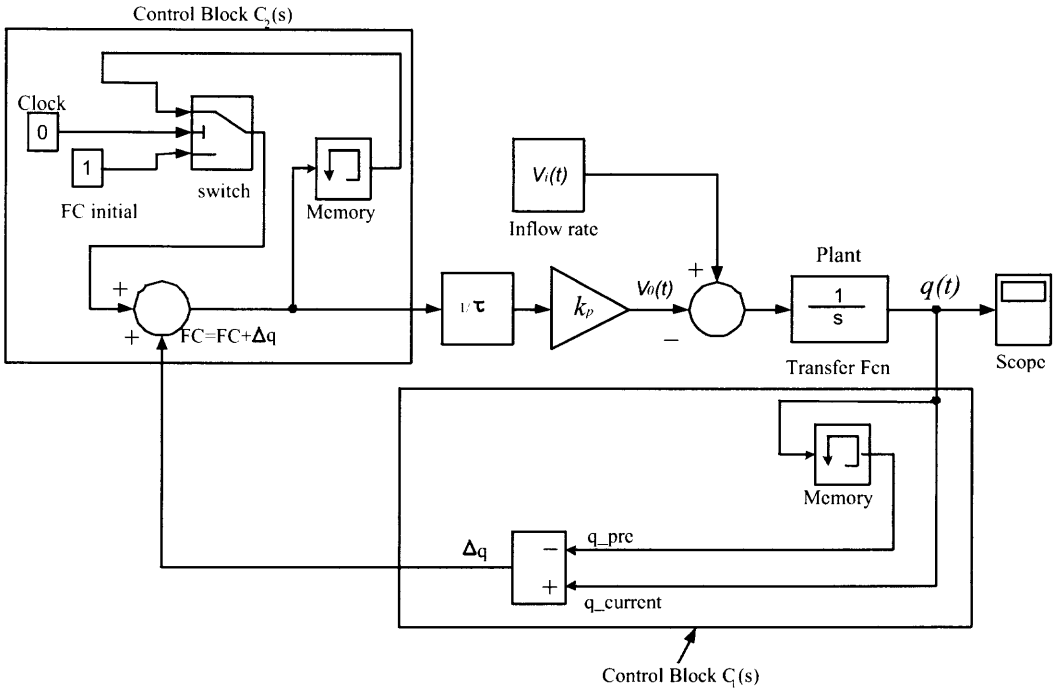

Figure 5.2 Block diagram of the explicit feedback control system.

The stability behavior of this feedback system is studied by control analysis and by using computer simulation (Simulink). Figure 5.2 shows the block diagram of a fluid system with a feedback control that implements the outflow rate to effectively track the

inflow rate, used in Simulink. Control block $C_1(s)$ is used to generate Δ_q, which is the difference between the actual current fluid level $q_{current}$ and the previous fluid level q_{pre}. The Control block $C_2(s)$ is used to generate FC, which is regarded as the amount of fluid to be drained out. FC is adjusted by the previous FC plus the updated Δ_q. With the block $\frac{1}{\tau}$ and proportional gain K_p, the outflow rate is obtained.

Figure 5.3 shows the simulation results in which the outflow rate effectively follows the inflow rate. In this figure, the top graph shows the inflow rate, and the bottom graph shows the outflow rate. In this example, we use a random number generator as the source of the inflow rate with mean of 100 and variance of 10. Figure 5.4 indicates that the fluid level is stable around the constant value of 9. Because the outflow rate effectively follows the inflow rate, the fluid does not accumulate so as to cause fluid overflow.

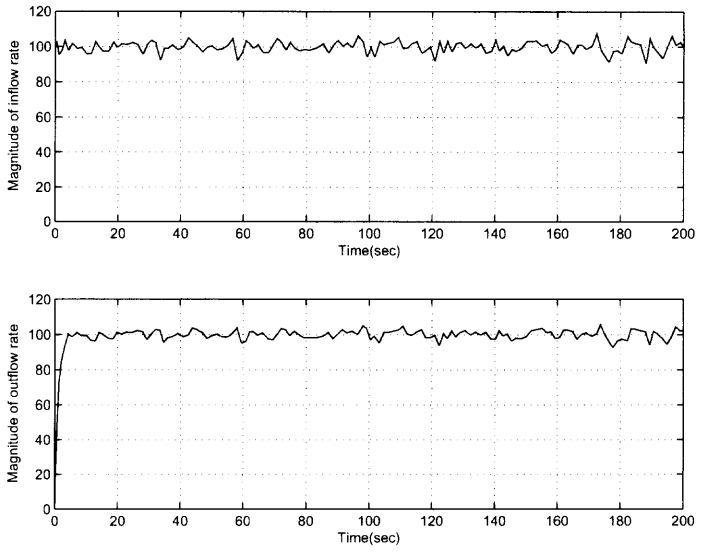

Figure 5.3 Simulink results of the outflow rate tracking the inflow rate.

The stability analysis on the feedback control system is shown below The block diagram in Figure 5.2 is redrawn as Figure 5.5 shows.

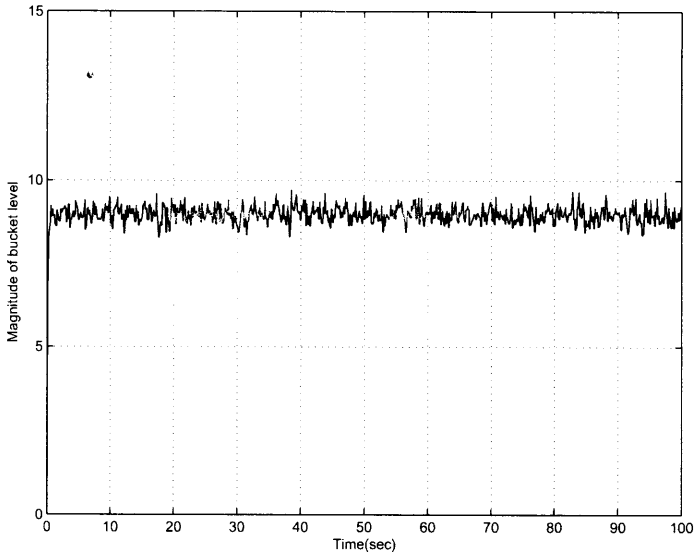

Figure 5.4 Simulation result on fluid level of the example.

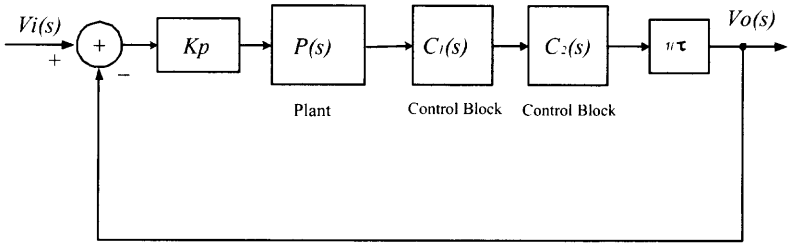

Figure 5.5 Block diagram of the fluid level control with a proportional control.

Here, $C_1(s)$ is used to generate Δ_q.

$$\Delta_q(t) = q(t) - q(t - \alpha_1). \tag{5.3}$$

Consider that α_1 is period of time to generate Δ_q.

The Laplace transform is derived:

$$\Delta_q(s) = q(s)(1 - e^{-\alpha_1 s}). \tag{5.4}$$

The transfer function for $C_1(s)$ is

$$C_1(s) = \frac{\Delta_q(s)}{q(s)} = 1 - e^{-\alpha_1 s}. \tag{5.5}$$

and $C_2(s)$ is used to generate FC as

$$FC(t) = FC(t - \alpha_2) + \Delta_q(t). \tag{5.6}$$

Here, α_2 is period of time to update FC, which is the amount of fluid. The Laplace transform is derived as:

$$FC(s) = FC(s)e^{-\alpha_2 s} + \Delta_q(s). \tag{5.7}$$

The transfer function for $C_2(s)$ is

$$C_2(s) = \frac{FC(s)}{\Delta_q(s)} = \frac{1}{1 - e^{-\alpha_2 s}}. \tag{5.8}$$

The transfer function of the system is derived from the block diagram shown in Figure 5.5:

$$H(s) = \frac{V_o(s)}{V_i(s)} = \frac{K_p R(s)}{1 + K_p R(s)}, \tag{5.9}$$

where

$$R(s) = \frac{C_1(s)C_2(s)}{s\tau} = \frac{1 - e^{-\alpha_1 s}}{1 - e^{-\alpha_2 s}} \cdot \frac{1}{s\tau} \tag{5.10}$$

If the pole of the system is in the left half plane (LHP), the feedback control system is stable. The denominator of the transfer function $H(s)$ is then:

$$1 + \frac{K_p}{s\tau} \frac{1 - e^{-\alpha_1 s}}{1 - e^{-\alpha_2 s}}.$$

There are two cases:

(1) when $\alpha_1 = \alpha_2$, the denominator of the transfer function $H(s)$ is $1 + \frac{K_p}{s\tau}$, then as long as $\frac{K_p}{\tau}$ is positive, the pole of the system is in the LHP, the feedback control system is stable.

(2) when $\alpha_1 \neq \alpha_2$, the term $e^{-\alpha s}$ can be approximated by Taylor series expansion of the exponential function or by first order $Padé$ approximation [71]. It is easy to see that

the poles of the system are in the LHP. Therefore, the feedback control system is stable.

□

As the control system shows stability, the feedback concept can be interpreted into an cell-based arbitration scheme for the discrete domain. The objective of the new scheme is that each VOQ sets the serving rate according to the input loading condition and the accumulation of cells in the VOQ. If the input traffic is heavy in one queue and the service rate is such that accumulation of cells occurs, the queue requests more service. Otherwise, the queue requests less service. Based on the analysis above, the difference in the VOQ occupancy between two consecutive service cycles is used to adaptively update the frame size according to the input loading condition.

5.3 FRE Arbitration Scheme

The proposed arbitration scheme is round-robin based. In each VOQ (and CPB), there are two counters: a frame-size counter, $FC_{i,j}(T)$, and a current service counter, $CC_{i,j}(t)$, where T is the service cycle that starts after the time slot when a frame is completely served, and t is any time slot when $VOQ_{i,j}$ receives service. The value of $FC_{i,j}(T)$, $|FC_{i,j}(T)|$, indicates the frame size; that is, the maximum number of cells that $VOQ_{i,j}$ can send in consecutive time slots to the CPB, one cell per time slot. The initial value of $|FC_{i,j}(T)|$ is one cell. $CC_{i,j}(t)$ counts the number of serviced cells of a given frame of $VOQ_{i,j}$ at time slot t. A regressive-fashion count is used in CC as CC only considers FC at the end of a serviced frame. The initial value of $CC_{i,j}(t)$, $|CC_{i,j}(t)|$, is one cell (i.e., its minimum value).

The input arbitration process is as follows. An input arbiter selects an eligible $VOQ_{i,j'}$ in round-robin fashion, starting from the pointer position, j, where $j' \geq j$, and $a > b$ means a is sorted after b in the round-robin schedule. A VOQ is considered eligible if the VOQ is not empty and the corresponding CPB is not full.

For the selected $VOQ_{i,j'}$, if $|CC_{i,j'}(t)| > 1$, $|CC_{i,j'}(t+1)| = |CC_{i,j'}(t)| - 1$, and the input pointer remains at $VOQ_{i,j'}$, so that this VOQ has the higher priority for service in the next time slot and the frame transmission can continue.

If $|CC_{i,j'}(t)| = 1$, a new service cycle is defined as T, the input pointer is updated to $(j'+1) \mod N$, $|FC_{i,j'}(T+1)|$ is increased by $\Delta_{i,j}$ cells, $|CC_{i,j'}(t+1)| = |FC_{i,j'}(T+1)|$, where $\Delta_{i,j}$ is defined as the current actual queue occupancy $Q(T+1)$ minus the previous queue occupancy $Q(T)$. Note that the number of time slots in each service cycle is not necessarily constant.

For the sake of clarity, the following pseudo-code describes the input arbitration scheme, as seen at an input:

-At time slot t, starting from the pointer position j, find the nearest eligible $VOQ_{i,j'}$ *in a round-robin fashion.*

-Send the HOL cell from $VOQ_{i,j'}$ *to* $CPB_{i,j'}$ *time slot* $t+1$, *and*

-If $|CC_{i,j'}(t)| > 1$ *then*

$|CC_{i,j'}(t+1)| = |CC_{i,j'}(t)| - 1$,

the pointer points to j'.

-else (i.e., $|CC_{i,j'}(t)| = 1$*)*

Set the new serving cycle as $T+1$,

$\Delta_{i,j'} = Q_{i,j'}(T+1) - Q_{i,j'}(T)$,

$|FC_{i,j'}(T+1)| = |FC_{i,j'}(T)| + \Delta_{i,j'}$,

$|CC_{i,j'}(t+1)| = |FC_{i,j'}(T+1)|$,

the pointer points to (j'+1) modulo N.

- Go to the next time slot.

Figure 5.6 shows an example of FRE. Assume that this figure is related to $VOQ_{i,j}$, where Q shows the occupancy of this VOQ and FC shows how the frame size is set each time $VOQ_{i,j}$ gets a frame service completed. As the figure shows, each time FC

is updated, the addition to FC is the difference between the last frame size and the positive or negative change on the VOQ occupancy.

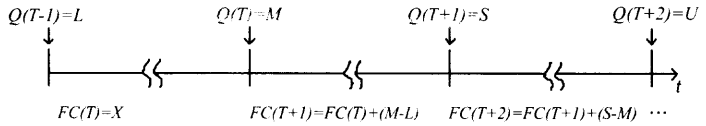

Figure 5.6 Example of FRE arbitration.

Here, three different schemes are being considered for output arbitration: **a)** round-robin (RR) selection, where the output pointer moves to one position beyond the selected one, or $(i + 1)$ modulo N, when $CPB_{i,j}$ is selected, **b)** persistent round-robin (PRR) selection, where the pointer moves to the selected input, or i when $CPB_{i,j}$ is selected, and **c)** FRE, as the input arbitration, where the values of $Q(T)$ are the occupancy of the VOQs at time T (note that the CPBs considered in this chapter have a size of one cell, therefore, considering the occupancy of them would not be practical). The combination of FRE as input arbitration with RR, PRR, and FRE are denoted as FRE-RR, FRE-PRR, and FRE-FRE, respectively.

5.4 Performance Evaluation

The performance evaluations are produced by computer simulation. The traffic models considered have destinations with uniform and nonuniform distributions, with Bernoulli arrivals. The simulation does not consider the segmentation and re-assembly delays. The simulation results are obtained with a 95% confidence interval, not greater than 5% for the average cell delay.

5.4.1 Uniform Traffic

It is assumed that cells arrive at each input in a slot by slot manner. Under a Bernoulli arrival process, the probability that there is a cell arriving in each time slot is identical

in and independent of any other slot. This probability is referred as the offered load to the input. If each cell is equally likely to be destined for any output, the traffic becomes uniformly distributed over the switch.

In the bursty traffic model, each input alternates between active and idle periods of geometrically distributed duration. During an active period, cells destined for the same output arrive in consecutive time slots. It is assumed that there is at least one cell in an active period. An active period is called a burst.

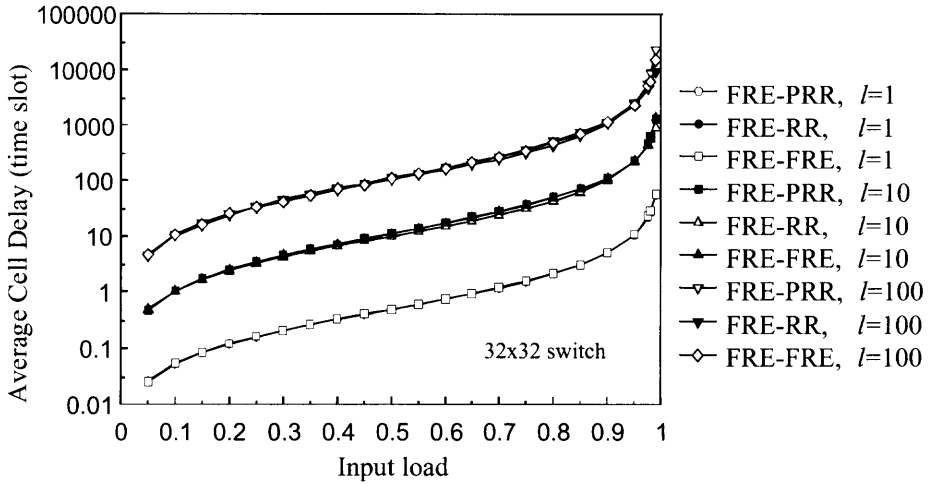

Figure 5.7 Performance with Bernoulli and bursty arrivals.

Figure 5.7 shows simulation results of a 32×32 CICB switch with FRE-PRR, FRE-RR, and FRE-FRE under uniform traffic with Bernoulli arrivals ($l = 1$) and bursts arrivals with average lengths of 10 and 100 ($l = 10$ and $l = 100$) cells. The simulation shows that the average delay is proportional to the burst length and that the throughput is unaffected at any load. The simulation shows that all three arbitration schemes provide nearly 100% throughput under uniform traffic.

5.4.2 Nonuniform Traffic: Unbalanced

The unbalanced traffic model uses a probability, w, as the fraction of input load directed to a single predetermined output, while the rest of the input load is directed to all outputs with uniform distribution. Let us consider input port s, output port d, and the offered input load for each input port ρ. The traffic load from input port s to output port d, $\rho_{s,d}$ is given by,

$$\rho_{s,d} = \begin{cases} \rho\left(w + \frac{1-w}{N}\right) & \text{if } s = d \\ \rho\frac{1-w}{N} & \text{otherwise.} \end{cases} \tag{5.11}$$

When $w = 0.0$, the offered traffic is uniform. On the other hand, when $w = 1.0$, it is completely directional, from input s to output d, where $s = d$. Figure 5.8 shows the

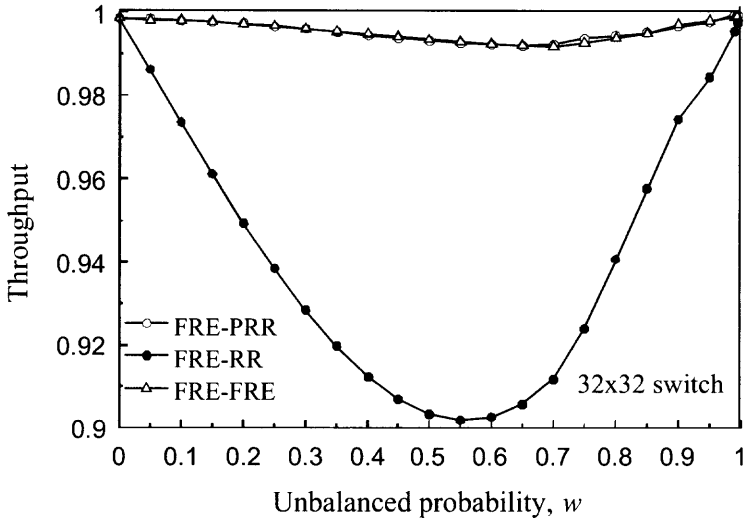

Figure 5.8 Performance under unbalanced traffic.

simulation results of a 32×32 CICB switch with FRE-PRR, FRE-RR, and FRE-FRE under unbalanced traffic. The results show that FRE-PRR and FRE-FRE provide very high throughput (more than 99%) under unbalanced traffic. However, the throughput of FRE-RR when $w = 0.55$ is 90.19%.

5.4.3 Nonuniform Traffic: Diagonal

The diagonal traffic can be represented as $d\rho(i,j) = d\rho$ for $i = j$ and $(1-d)\rho$ for $j = (i+1) \bmod N$, where d is the diagonal degree probability. This traffic model presents load distributions among two outputs per each input. Figure 5.9 shows the performance of the different arbitration schemes under diagonal traffic with $d = 0.75$. The throughput of FRE-RR is the highest, nearly 100%, and the average cell delay is the lowest among the three schemes.

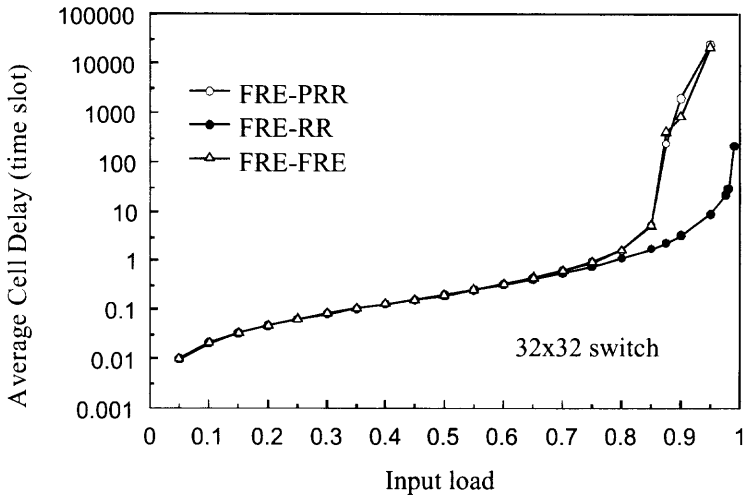

Figure 5.9 Performance under diagonal traffic.

5.4.4 Nonuniform Traffic: Chang's and Asymmetric

The schemes are also tested under other nonuniform traffic models: Chang's [57] and asymmetric [56].

Chang's traffic model can be defined as $\rho = 0$ for $i = j$ and $\rho = \frac{1}{N-1}$, otherwise. Figure 5.10 shows the average cell delay of FRE in a 32×32 switch under Chang's traffic model. As the figure shows, there is no significant difference on the average cell delay of FRE-PRR, FRE-RR, and FRE-FRE under Chang's traffic model. Figure 5.11 shows the

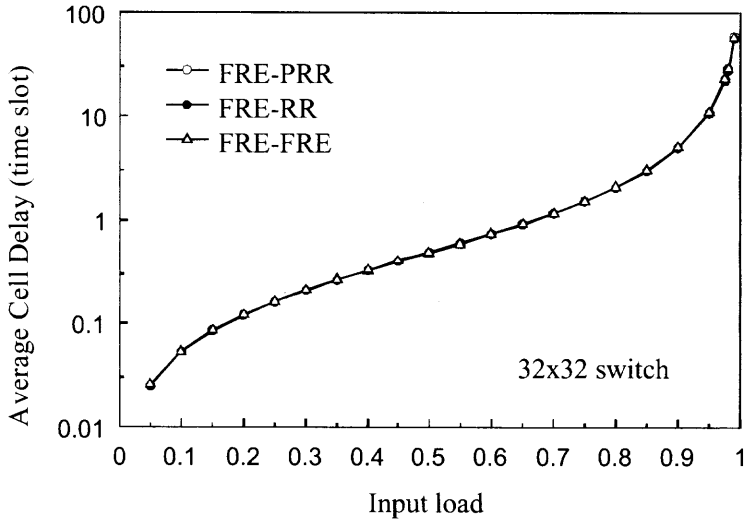

Figure 5.10 Performance under Chang's traffic.

average cell delay experienced by a 32×32 switch under asymmetric traffic model. The average cell delay of FRE-FRE is the largest.

In summary, based on the evaluations of the performance of FRE-PRR, FRE-RR, and FRE-FRE under all the traffic models above and by considering the implementation cost and complexity, FRE-PRR shows an advantage over other schemes as FRE-PRR provides the highest throughput under unbalanced traffic.

5.5 Conclusions

This chapter introduced a frame-based round-robin arbitration scheme with explicit feedback control for CICB packet switches. The frame-based arbitration scheme dynamically sets the frame size according to the input load and to the accumulation of cells in a VOQ. It is shown that the concept of explicitly feedback in a continuous system is stable. This chapter also provides an interpretation of the explicit feedback approach into a discrete system for a cell-based switch. FRE is used as the input arbitration scheme in combination with RR, PRR, and FRE as output arbitration schemes. These combined schemes deliver high performance under the presented uniform and nonuniform traffic models using a buffered

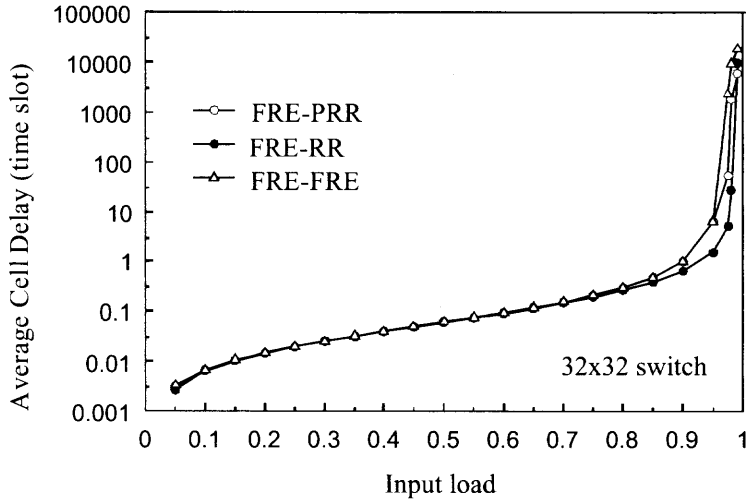

Figure 5.11 Performance under asymmetric traffic.

crossbar with one-cell crosspoint buffers. However, FRE-FRE is probably the scheme with the highest implementation cost as the output arbitration needs the VOQ occupancy (and not the CPB occupancy), which means that the VOQ occupancy needs to be sent from the inputs to the buffered crossbar. This may increase the transmission overhead.

Because the proposed scheme is based on round-robin selection, the arbitration needs not to compare the status among different VOQs, such as weight-based schemes do. Its novelty lies in that each VOQ sets its frame size by an adjustable value that changes according to the input loading and the accumulation of cells experienced in previous service cycles. In this way, the scheme attempts to explicitly control the service rate provided to a VOQ.

CHAPTER 6

CONTROL THEORETIC ANALYSIS OF ARBITRATION AND MATCHING SCHEMES FOR PACKET SWITCHES

6.1 Introduction

With the rapid development of optical networking technology, high-performance switches and routers are required to meet the demand of providing higher throughput than existing schemes. Various arbitration and matching schemes have been proposed. One common requirement, is the delivery of high throughput under admissible traffic conditions. If a switch can provide 100% throughput, the switch is stable as the queue occupancy at any of queuing points does not increase indefinitely.

As discussed in Chapter 4, previous stability analysis approaches concentrate on the evolution of the queue length. One important factor in a switch is the arbitration or matching scheme as this decides which queue can obtain the opportunity to transfer packets to the crosspoint buffers in a CICB switch or to outputs in an IB switch, respectively. Therefore, the scheme also determines how often the queues are served [69].

Two questions arise: what is the degree in which an arbitration or matching scheme affects the stability of the whole system, and how to analyze these issues using a control-theoretical technique?

In this chapter, a queuing system is modeled as a continuous system where discrete units, such as packets, are represented as fluid that can be controlled. It is shown that the dwell time, defined as the time a queue receives service in a service opportunity, is a factor that affects the stability of a queuing system. Furthermore, a case study of an arbitration scheme of a 2×2 switch is used to show that the feedback control model is an alternative approach to evaluate the stability of an arbitration scheme.

The chapter is organized as follows. Section 6.2 introduces the feedback control model for a queuing system and presents the analysis of the stability. Section 6.3 discusses the multi-queue control model and illustrates the dwell time selection as a factor that affects the stability of a queuing system. Section 6.4 presents a case study of an arbitration scheme in a 2×2 IB switch. Section 6.5 presents the conclusions.

6.2 Switch Modeling in a Continuous System

A continuous system is used to emulate a queuing system. The control system of an actuator is modeled as the arbitration/matching scheme.

In Chapter 5, the inflow rate V_i is used to represent the input traffic rate in a switch and V_o is used to represent the outflow rate, which is the serving rate provided by the arbitration scheme. Consider that in a queue with an inflow rate V_i, the queue occupancy y is required to be kept constant by changing the service rate of the queue [84].

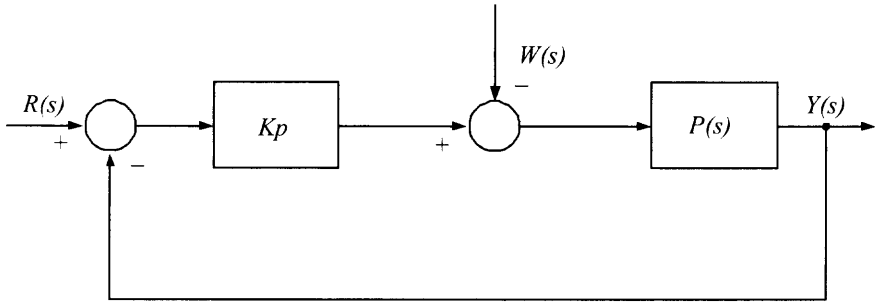

Figure 6.1 Block diagram of a queue occupancy control system with disturbance.

Figure 6.1 shows the block diagram of a fluid system with a feedback control to keep the queue occupancy as the set-point $R(s)$. Here, proportional control is used, where Kp is the gain. In this control model, the outflow rate is determined by the difference between the actual queue occupancy $y(t)$ at time t and the desired queue occupancy $R(t)$ at the initial condition. The objective is to show that the queue occupancy changes in a finite manner. It is assumed that the queue length can be large enough to avoid overflow and to allow measuring those differences. If $y(t) - R(t) > 0$, which means the inflow rate is

greater than the outflow rate, the feedback controller is expected to increase the outflow rate as much as needed to keep the level from raising indefinitely. Otherwise, the outflow rate needs to be decreased to let the level approach $R(t)$. The integral plant $P(s)$ reflects the fact that the level depends on the difference between the inflow and outflow rate. In the point view of control theory, V_i can be modeled as disturbance $W(s)$ [85], which assumes that the input traffic is a stochastic process.

According to Figure 6.1, assuming $R(t)$=0:

$$\frac{Y(s)}{W(s)} = \frac{-1}{s + Kp} \tag{6.1}$$

then, assuming that $W(t)$=0:

$$\frac{Y(s)}{R(s)} = \frac{Kp}{s + Kp} \tag{6.2}$$

It is easy to see that after combining these two cases, as long as Kp is positive, the pole of the system is always in the left half plane (LHP), therefore the queue occupancy control system is stable.

6.3 Selection of a Queue in a Minimum System

The continuous model system is used as a platform to build a multi-queue system to emulate an input queue including multiple VOQs. We use this multi-queue fluid model to theoretically demonstrate that a multiple queuing system of a switch can achieve stability by a suitable control scheme.

Here, a minimum system has only two queues. A controller is used to control this two-queue system. Figure 6.2 shows the equivalent control model of this two-queue system. Each time when the feedback information of queue occupancy $y(t)$ is ready, the difference between the set-point $R(t)$ and $y(t)$, named e, is calculated. If the maximum weight is assigned to the queue that has the larger queue occupancy change e, meanwhile any other queue receives no weight, which means that it will not be served, then the scheme

performs a maximum weight selection. When e_1 of Queue 1 is greater than e_2 of Queue 2, Queue 1 is open while Queue 2 is completely closed. k_1 is the gain for Queue 1's proportional control loop. k_2 is the gain for Queue 2's proportional control loop.

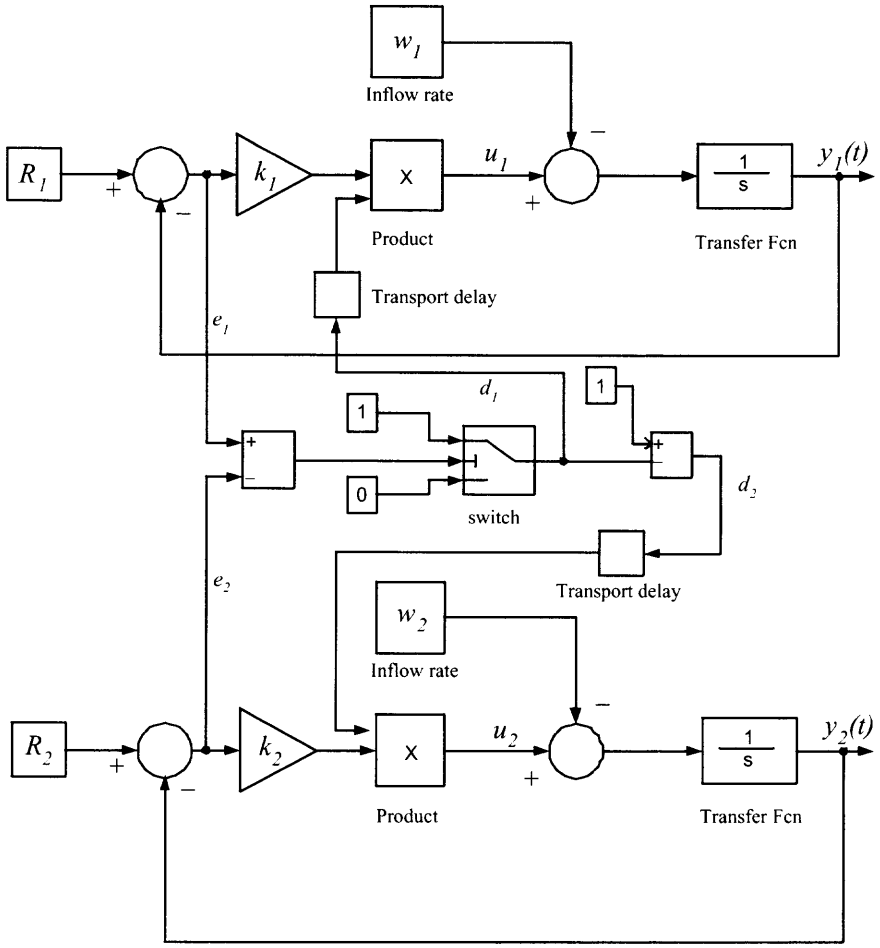

Figure 6.2 Block diagram of a two-queue control system with on/off control.

An on/off control is used between the two queues, e_1 and e_2 are the inputs to the controller, d_1 and d_2 are the outputs. The controller compares e_1 and e_2 and assign a binary value 1 to the higher one, and binary 0 to the lower one. Finally the controller feeds the two binary values d_1 and d_2 into the two queues' feedback loops, respectively. When the on/off controller makes one queue receive the service, this queue stays in the service state

for a time interval T *, which means that there can be no more changes during this time interval. The time interval is defined as the dwell time. On the other hand, the dwell time is regarded as a time delay to the queue does not receive the service.

As it is shown in Figure 6.2, Queue 1 gets $e^{-sT(1-d_1)}$ and Queue 2 gets $e^{-sT(1-d_2)}$. For example, if d_1 is 1, Queue 1 is open, after assigning $d_1 = 1$, the delay term $e^{-sT(1-d_1)}$ becomes equal to 1, which means that the delay term disappears and Queue 1 gets the service immediately; If d_2 is 0, Queue 2 is closed. After substituting $d_2 = 0$, the delay term $e^{-sT(1-d_2)}$ becomes e^{-sT}, which means that Queue 2 has a time delay which is equal to the dwell time T. This is why this dwell time is regarded as a time delay for the queue that does not receive the service.

The two queues' set points R_1 and R_2 are not necessarily of the same value. If they are chosen with the same value, the larger e indicates the larger queue occupancy, then this arbitration scheme is changed to be LQF.

The state space method [86] is used to model this two-queue fluid system, where each VOQ is a subsystem.

For subsystem 1,

$$\dot{x}_1 = A_1 x_1 + B_1(u_1 - w_1) \tag{6.3}$$

$$y_1 = C_1 x_1 \tag{6.4}$$

where x_1 represents the state, which is the queue occupancy in Queue 1, u_1 is the control signal, which is the input to the system, y is the output, and w_1 is the disturbance. The matrices A_1, B_1, C_1 determine the relationships between the state and input/output variables. In this case, 6.3 is called the state equation and 6.4 is called the output equation. This representation of a system provides a complete knowledge of all variables of the system.

From Figure 6.2, it is easy to obtain that:

$$u_1 = (R_1 - y_1)e^{-sT(1-d_1)}k_1$$

*This is different from the T in Chapter 5.

$$= \quad -C_1 k_1 e^{-sT(1-d_1)} x_1 + R_1 k_1 e^{-sT(1-d_1)}$$

Since the fluid system is a first order system as discussed above, $A_1 = 0$, $B_1 = 1, C_1 = 1$, then 6.3 and 6.4 can be represented as:

$$\dot{x}_1 = (-k_1 e^{-sT(1-d_1)}) x_1 + \begin{pmatrix} k_1 e^{-sT(1-d_1)} & -1 \end{pmatrix} \begin{pmatrix} R_1 \\ w_1 \end{pmatrix} \tag{6.5}$$

$$y_1 = x_1 \tag{6.6}$$

Similarly, for subsystem 2, $A_2 = 0$, $B_2 = 1, C_2 = 1$,

$$\dot{x}_2 = (-k_2 e^{-sT(1-d_2)}) x_2 + \begin{pmatrix} k_2 e^{-sT(1-d_2)} & -1 \end{pmatrix} \begin{pmatrix} R_2 \\ w_2 \end{pmatrix} \tag{6.7}$$

$$y_2 = x_2 \tag{6.8}$$

When combining these two subsystems, the LQF defines the states of Queue 1 and Queue 2 as the following.

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} -k_1 e^{-sT(1-d_1)} & 0 \\ 0 & -k_2 e^{-sT(1-d_2)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

$$+ \begin{pmatrix} k_1 e^{-sT(1-d_1)} & -1 & 0 & 0 \\ 0 & 0 & k_2 e^{-sT(1-d_2)} & -1 \end{pmatrix} \begin{pmatrix} R_1 & 0 \\ w_1 & 0 \\ 0 & R_2 \\ 0 & w_2 \end{pmatrix}$$

Furthermore, with the characteristic equation of square matrix A, which is $det[sI - A] = 0$, then

$$\begin{vmatrix} s + k_1 e^{-sT(1-d_1)} & 0 \\ 0 & s + k_2 e^{-sT(1-d_2)} \end{vmatrix} = 0 \tag{6.9}$$

and,

$$s^2 + [k_1 e^{-sT(1-d_1)} + k_2 e^{-sT(1-d_2)}]s + k_1 k_2 e^{-sT(2-d_1-d_2)} = 0 \tag{6.10}$$

where the delay term e^{-sT} can be approximated by Taylor Series Expansion of exponential function or first order $Padé$ approximation. Here, Taylor Expansion is used to approximate the delay term, which becomes:

$$s^2 + [k_1(1 - sT(1 - d_1)) + k_2(1 - sT(1 - d_2))]s + k_1 k_2(1 - sT(2 - d_1 - d_2)) = 0$$

Since $d_1 d_2 = 0$, it is assumed that $d_1 = 1$, $d_2 = 0$ then

$$(1 - k_2 T)s^2 + (k_1 + k_2 - k_1 k_2 T)s + k_1 k_2 = 0 \tag{6.11}$$

Therefore, the solutions are:

$$s_{1,2} = \frac{k_1 k_2 T - k_1 - k_2}{2(1 - k_2 T)} \pm \frac{\sqrt{(k_1 + k_2 - k_1 k_2 T)^2 - 4k_1 k_2(1 - k_2 T)}}{2(1 - k_2 T)} \tag{6.12}$$

The term $\frac{k_1 k_2 T - k_1 - k_2}{2(1 - k_2 T)}$ is now considered. If this term is negative, then the poles are on LHP, which means that the system is stable.

The solution of $\frac{k_1 k_2 T - k_1 - k_2}{2(1 - k_2 T)} < 0$ is calculated as

$$T < \frac{1}{k_2} \tag{6.13}$$

and also

$$T < \frac{1}{k_2}\left[1 + \frac{k_2}{k_1}\right] \tag{6.14}$$

After combining 6.13 and 6.14 together, T becomes

$$T < \frac{1}{k_2} \tag{6.15}$$

Inequality 6.15 gives the condition for the dwell time to affect the stability of the control system, which means the time on arbitration affects the stability of the arbitration scheme.

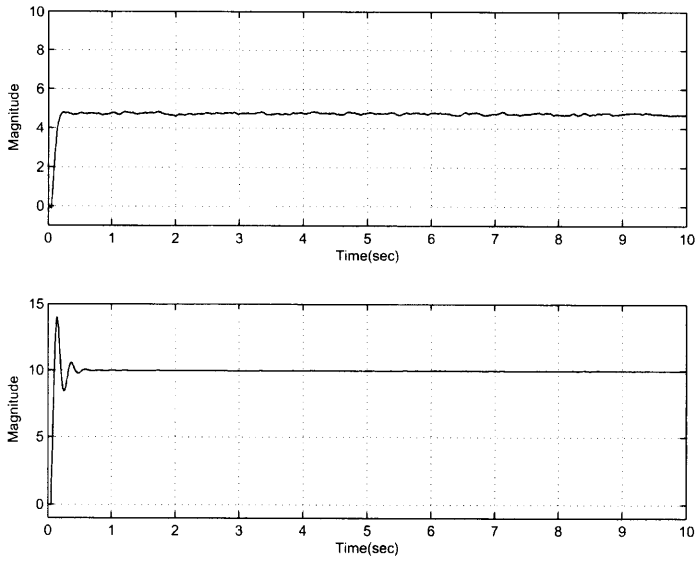

Figure 6.3 Simulation result for a two-queue queue occupancy control system with on/off control and dwell time of 0.04 sec.

In the following experiments, a two-queue control system is simulated in Simulink. The simulation of this system shows that the dwell time affects the stability of the control system. The set points for the two queues are 5 and 10 respectively. The gain k_1 and k_2 are 10 and 20, respectively. From 6.15, the dwell time $T < 0.05$ sec. Figure 6.3 shows the simulation result when the dwell time is 0.04 sec. Figure 6.4 shows the simulation result when the dwell time is 0.06 sec. When the dwell time is 0.06 sec, the system starts to lose its stability. Figure 6.5 shows the simulation result when the dwell time is 0.07 sec. When the dwell time is 0.07 sec, the system starts to become unstable.

6.4 Example: Analysis of a Matching Scheme of an IB Switch

Figure 6.6 shows a 2×2 IB switch. Figure 6.7 shows the equivalent control model of this 2×2 IB switch. There are four queues. The queues $T1$ and $T2$ are used to emulate the two VOQs of input port 1 in a 2×2 switch. The on/off Controller 1 is used to control the service rate of this input port to the switch core. Similarly queues $T3$ and $T4$ are used to emulate the two VOQs of input port 2, Controller 2 controls the rate of this input port into

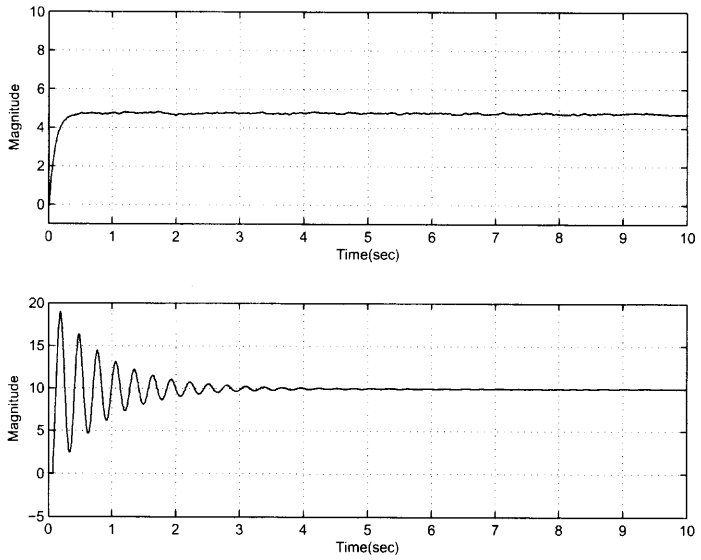

Figure 6.4 Simulation result for a two-queue queue occupancy control system with on/off control and dwell time of 0.06 sec.

the switch core. In addition, Controller 3 is used to control the outflow rate of $VOQ(1,1)$ and $VOQ(2,1)$ into the output 1. e_1 and e_3 are fed into Controller 3 and the controller generates m_1 and m_3 to affect u_1 and $u_3$for the next control cycle. Similarly, Controller 4 is used to control the outflow rate of $VOQ(1,2)$ and $VOQ(2,2)$ into Output 2. At the same time, e_2 and e_4 are fed into Controller 4 and the controller generates m_2 and m_4 to affect u_2 and u_4 for the next control cycle.

The following is the theoretic analysis of this four-queue queuing system which is used to represent a 2×2 IB switch. Each VOQ is a subsystem, there are four subsystems to be considered in this analysis.

For subsystem 1, since Controller 3 generates m_1, which feeds back to u_1, then

$$\begin{aligned} u_1 &= (R_1 - y_1)e^{-sT(1-d_1)}e^{-sT(1-m_1)}k_1 \\ &= -c_1k_1e^{-sT(2-d_1-m_1)}x_1 + R_1k_1e^{-sT(2-d_1-m_1)}. \end{aligned}$$

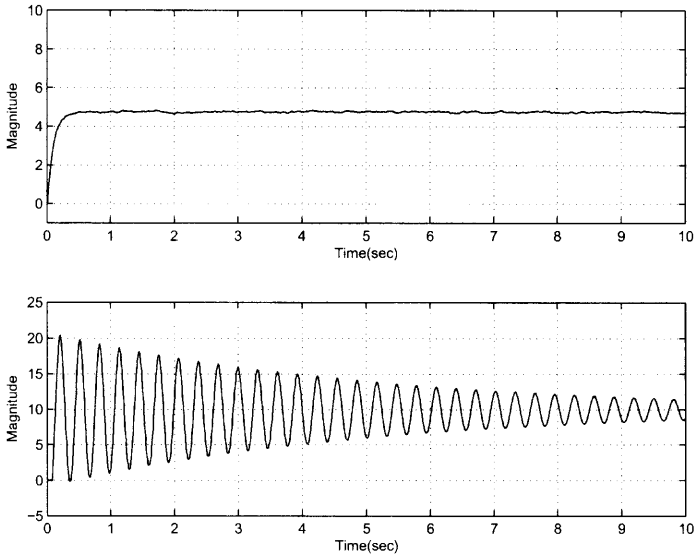

Figure 6.5 Simulation result for a two-queue queue occupancy control system with on/off control and dwell time of 0.07 sec.

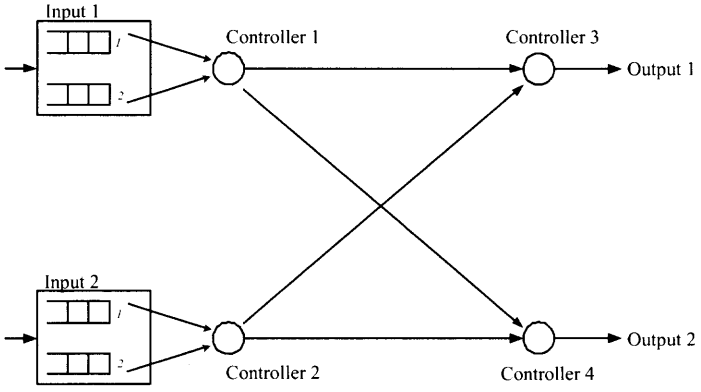

Figure 6.6 Block diagram of a 2×2 IB switch.

Similarly,

$$\begin{aligned} u_2 &= (R_2 - y_2)e^{-sT(1-d_2)}e^{-sT(1-m_2)}k_2 \\ &= -c_2k_2e^{-sT(2-d_2-m_2)}x_2 + R_2k_2e^{-sT(2-d_2-m_2)}. \end{aligned}$$

$$\begin{aligned} u_3 &= (R_3 - y_3)e^{-sT(1-d_3)}e^{-sT(1-m_3)}k_3 \\ &= -c_3k_3e^{-sT(2-d_3-m_3)}x_3 + R_3k_3e^{-sT(2-d_3-m_3)}. \end{aligned}$$

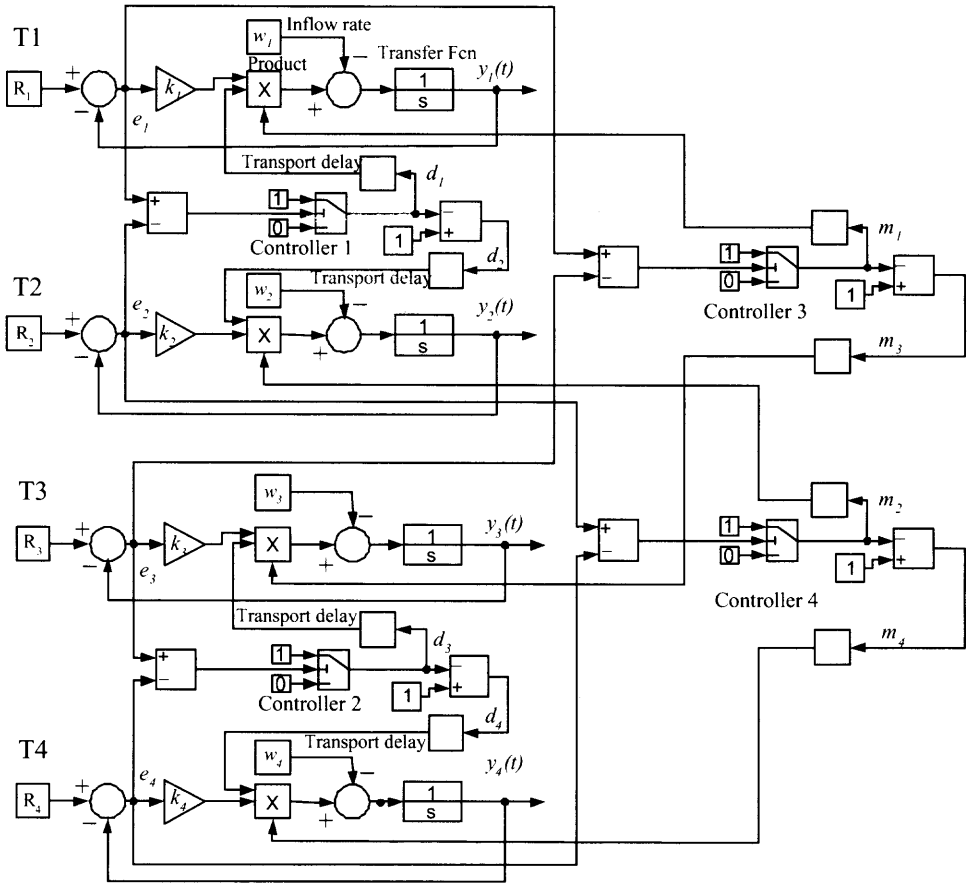

Figure 6.7 Block diagram of a four-queue control model in a 2×2 IB switch.

and

$$\begin{aligned} u_4 &= (R_4 - y_4)e^{-sT(1-d_4)}e^{-sT(1-m_4)}k_4 \\ &= -c_4k_4e^{-sT(2-d_4-m_4)}x_4 + R_4k_4e^{-sT(2-d_4-m_4)}. \end{aligned}$$

Therefore, the following subsystems are obtained:

Subsystem 1:

$$\dot{x}_1 = (-k_1e^{-sT(2-d_1-m_1)})x_1 + \begin{pmatrix} k_1e^{-sT(2-d_1-m_1)} & -1 \end{pmatrix} \begin{pmatrix} R_1 \\ w_1 \end{pmatrix} \tag{6.16}$$

$$y_1 = x_1 \tag{6.17}$$

Subsystem 2:

$$\dot{x}_2 = (-k_2 e^{-sT(2-d_2-m_2)})x_2 + \begin{pmatrix} k_2 e^{-sT(2-d_2-m_2)} & -1 \end{pmatrix} \begin{pmatrix} R_2 \\ w_2 \end{pmatrix} \tag{6.18}$$

$$y_2 = x_2 \tag{6.19}$$

Subsystem 3:

$$\dot{x}_3 = (-k_3 e^{-sT(2-d_3-m_3)})x_3 + \begin{pmatrix} k_3 e^{-sT(2-d_3-m_3)} & -1 \end{pmatrix} \begin{pmatrix} R_3 \\ w_3 \end{pmatrix} \tag{6.20}$$

$$y_3 = x_3 \tag{6.21}$$

Subsystem 4:

$$\dot{x}_4 = (-k_4 e^{-sT(2-d_4-m_4)})x_4 + \begin{pmatrix} k_4 e^{-sT(2-d_4-m_4)} & -1 \end{pmatrix} \begin{pmatrix} R_4 \\ w_4 \end{pmatrix} \tag{6.22}$$

$$y_4 = x_4 \tag{6.23}$$

Then the characteristic equation $det[sI - A] = 0$ is

$$\begin{vmatrix} s + k_1 e^{-sT(2-d_1-m_1)} & 0 & 0 & 0 \\ 0 & s + k_2 e^{-sT(2-d_2-m_2)} & 0 & 0 \\ 0 & 0 & s + k_3 e^{-sT(2-d_3-m_3)} & 0 \\ 0 & 0 & 0 & s + k_4 e^{-sT(2-d_4-m_4)} \end{vmatrix} = 0 \tag{6.24}$$

and the characteristic polynomial which is on left-hand side of the characteristic equation is $\prod (s + k_i e^{-sT(2-d_i-m_i)})$, where i is the index number of the subsystem.

Furthermore, by using Taylor expansion to approximate the delay term, the characteristic equation becomes $\prod (s + k_i(1 - sT(2 - d_i - m_i))) = 0$.

The solutions to this equation are:

if $d_1 = 1, m_1 = 1, s_1 = -k_i$.

if $d_2 = 0, m_2 = 1, s_2 = -\frac{k_2}{1-k_2T}$.

if $d_3 = 1, m_3 = 0, s_3 = -\frac{k_3}{1-k_3T}$.

if $d_4 = 0, m_4 = 0, s_4 = -\frac{k_4}{1-2k_4T}$.

Combining the above cases, under the condition of

$$T < \min\{\frac{1}{k_2}, \frac{1}{k_3}, \frac{1}{2k_4}\} \tag{6.25}$$

the fluid control system is stable.

The above analysis can be extended to $N \times N$ IB switches. With the control theoretic analysis, arbitration designers can obtain a full scenario of the queuing system in the point view of control theory.

6.5 Conclusions

In this chapter, an analytical methodology is proposed to study the stability of arbitration or matching schemes for packet switches. A continuous system is used and a control model is proposed to emulate a queuing system. The technique is applied to an arbitration scheme. It is shown that the dwell time is a factor that affects the stability of a queuing system. Furthermore, a case study of an arbitration scheme on a 2×2 switch is used to show that the feedback control model is an alternative approach to evaluate the stability of an arbitration or matching scheme. This analysis can be used as a complementary approach to evaluate the system stability.

CHAPTER 7

CONCLUSIONS

The increasing demand for higher data rates on the Internet requires routers that deliver high performance for high-speed connections. While output-buffered switches are known for their limited scalability, IB switches have been of interest for research and commercialization. However, IB switches may not be able to keep up with the increasing data rates as optical technology advances rapidly. To keep up with the increasing data rates, switches based on internally-buffered crossbar are considered a feasible solution for the next generation packet switches.

Stability is the utmost important in switch design. A stable arbitration scheme can achieve 100% throughput and the high throughput is one of the aims of switch design. In this dissertation, stability analysis and theoretical proof are given to the arbitration schemes such as RR-AF, LQF-LCO, and FRE.

The RR-AF arbitration scheme uses the concept of adaptable-size frame, where the frame size depends on the service received by a queue. The presented simulation results show that this throughput is achieved with low average cell delay and that the analytical result can be extended to nonuniform traffic patterns, including the unbalanced traffic model. The results also show that a buffered crossbar with one-cell crosspoint buffers is sufficient to provide such throughput with the proposed round-robin based arbitration. One of the contributions of this dissertation is proving that the round-robin scheme with adaptable-size frame arbitration delivers 100% throughput under uniform traffic.

Switches with crosspoint buffers need to consider the transmission delays, or round-trip times. An AQM-based flow-control mechanism for CICB switches is analyzed. The stability margin is investigated by analyzing the relationship between the crosspoint buffer size and the round-trip times. The conditions to make the flow control system stable are

shown. This may provide a guideline on how to choose the crosspoint buffer size under non-negligible round-trip times. Also, it is shown that the system's transient response time is improved by almost 40% when adding an input shaper to the closed-loop system.

A fluid model is used to prove that a combined input-crosspoint buffered, CICB, packet switch can provide 100% throughput with no speedup and under admissible traffic. The fact that input and output arbitrations in a CICB switch are performed separately allows us to analyze the queues at the inputs of the CICB switch while assuming an output arbitration at the buffered crossbar that keeps inputs uninhibited, and by analyzing the buffered crossbar while assuming that an input arbitration can select any VOQ that has a crosspoint buffer available. Furthermore, an output arbitration scheme is proposed to avoid input inhibition and also to keep outputs active. The performance and feasibility of such an output arbitration scheme is shown by the existence of a matrix decomposition scheme. The contribution lies in a theoretical proof on CICB's throughput on the condition of no speedup and proposed a combination of arbitration schemes: LQF with a distributed implementation among all inputs and LCO with a centralized implementation at the buffered crossbar.

Based on the intuition from RR-AF arbitration scheme, a new frame-based round-robin arbitration scheme with explicit feedback control, FRE, is proposed for CICB packet switches. The frame-based arbitration scheme dynamically sets the frame size according to the input load and to the accumulation of cells in a VOQ. It is shown that the concept of explicitly feedback in a continuous system is stable, and an interpretation of the explicit feedback approach into a discrete system for a cell-based switch is provided. FRE is tested as the input arbitration in combination with RR, PRR, and FRE as output arbitration schemes in a CICB switch. These schemes deliver high performance under the presented uniform and nonuniform traffic models using a buffered crossbar with one-cell crosspoint buffers. The novelty of FRE lies in that each VOQ sets its frame size by adjusting parameter, $\Delta_{i,j}$, which indicates the degree of service needed by $VOQ(i,j)$. This value is adjusted according to the input loading and the accumulation of cells experienced in previous service

cycles. In this way, the scheme attempts to explicitly control the service rate provided to a VOQ.

Furthermore, an analytical methodology is proposed to study the stability of arbitration and matching schemes for packet switches. A continuous system is used and a control model is proposed to emulate a queuing system. The study shows that the dwell time, which is defined as the time a queue receives service in a service opportunity, is a factor that affects the stability of a queuing system. This feedback control model is an alternative approach to evaluate the stability of an arbitration scheme.

REFERENCES

[1] S. Nojima, E. Tsutsui, H. Fukuda, and M. Hashimmoto, "Integrated Packet Network Using Bus Matrix," *IEEE J. Select. Areas Commun.*, Vol. SAC-5, No. 8, pp. 1284-1291, Oct. 1987.

[2] A. K. Gupta, L. O. Barbosa, and N. D. Georganas, "16 x 16 Limited Intermediate Buffer Switch Module for ATM Networks," *Proc. IEEE Global Telecommunications Conference (GLOBECOM'91)*, pp. 939-943, Dec. 1991.

[3] A. K. Gupta, L. O. Barbosa, and N. D. Georganas, "Limited Intermediate Buffer Switch Modules and their Interconnection Networks for B-ISDN," *Proc. IEEE International Conference on Communications (ICC'92)*, pp. 1646-1650, June 1992.

[4] E. Oki, N. Yamanaka, Y. Ohtomo, K. Okazaki, and R. Kawano, "A 10-Gb/s (1.25 Gb/s x8) 4 x 0.25-μm CMOS/SIMOX ATM Switch Based on Scalable Distributed Arbitration," *IEEE J. Solid-State Circuits*, Vol. 34, No. 12, pp. 1921-1934, Dec. 1999.

[5] M. Karol and M. Hluchyj, "Queuing in High-performance Packet-switching," *IEEE J. Select. Areas Commun.*, Vol. 6, pp. 1587-1597, Dec. 1988.

[6] Y. Doi and N. Yamanaka, "A High-Speed ATM Switch with Input and Cross-Point Buffers," *IEICE Trans. Commun.*, Vol. E76, No. 3, pp. 310-314, March 1993.

[7] D. E. Re and R. Fantacci, "Performance Evaluation of Input and Output Queuing Techniques in ATM Switching System, " *IEEE Trans. Commun.*, Vol. 40, No. 10, pp. 1565-1575, Oct. 1993.

[8] E. Oki and N. Yamanaka, "Scalable crosspoint buffering ATM switch architecture using distributed arbitration scheme," *Proc. IEEE ATM'97 workshop*, pp. 28-35, 1997.

[9] E. Oki and N. Yamanaka, "Tandem-Crosspoint ATM Switch with Input and Output Buffers," *IEEE Communications Letters*, Vol. 2, No. 7, pp. 465-467, July 1998.

[10] S. Li and N. Ansari, "Chapter 1.3: Switch Architecture and Scheduling Algorithms," in *ATM Handbook* (F. Golshani and F. Groom, Eds.), International Engineering Consortium, pp. 37-54, 2000.

[11] S. Li and N. Ansari, "Input-Queued Switching with QoS Guarantees," *Proc. IEEE Conference on Computer Communications (INFOCOM'99)*, pp. 1152-1159, March 1999.

[12] M. Nabeshima, "Performance Evaluation of a Combined Input- and Crosspoint-Queued Switch," *IEICE Trans. Commun.*, Vol. E83-B, No. 3, March 2000.

[13] F. M. Chiussi and A. Francini, "A Distributed Scheduling Architecture for Scalable Packet Switches," *IEEE J. Select. Areas Commun.*, pp. 2665-2683, Dec. 2000.

[14] N. McKeown, "The iSLIP Scheduling Algorithm for Input-queued Switches," *IEEE/ACM Trans. Networking.*, Vol. 7, No. 2, pp. 188-201, April 1999.

[15] R. Rojas-Cessa, E. Oki, and H. J. Chao, "CIXOB-1: Combined Input-crosspoint-output Buffered Packet Switch," *Proc. IEEE Global Telecommunications Conference (GLOBECOM'01)*, Vol. 4, pp. 2654-2660, Nov. 2001.

[16] N. McKeown, "Scheduling algorithms for input-queued cell switches," Ph.D. dissertation, Dept. Elect. Eng. Comput. Sci., Univ. California at Berkeley, Berkeley, CA, 1995.

[17] D. Shah, "Maximal Matching Scheduling is Good Enough," *Proc. IEEE Global Telecommunications Conference (GLOBECOM'03)*, Vol. 6, pp. 3009-3013, Dec. 2003.

[18] N. McKeown, A. Mekkittikul, V. Anantharam, and J. Walrand, "Achieving 100% Throughput in an Input-queued Switch," *IEEE Trans. Commun.*, Vol. 47, No. 8, pp. 1260-1267, Aug. 1999.

[19] R. Magill, K. Benson, T. Hrabik, and J. Kenney, "A Simple Shaping Scheme for Frame-Based Bandwidth Allocation," *Proc. IEEE Global Telecommunications Conference (GLOBECOM'00)*, Vol. 1, pp. 651-655, Dec. 2000.

[20] O. Bonaventure and J. Nelissen, "Guaranteed Frame Rate: A Better Service for TCP/IP in ATM Networks," *IEEE Network*, pp. 46-54, Jan./Feb. 2001.

[21] H. T. Kung and K. Chang, "Receiver-Oriented Adaptive Buffer Allocation in Credit-Based Flow Control for ATM Networks," *Proc. IEEE Conference on Computer Communications (INFOCOM'95)*, Vol. 1, pp. 239-252, 1995.

[22] S. Floyd and V. Jacobson, "Random Early Detection Gateways for Congestion Avoidance," *IEEE/ACM Trans. Networking*, Vol. 1, pp. 397-413, Aug. 1993.

[23] S. Athuraliya, S. H. Low, V. H. Li, and Q. Yin, "REM: Active Queue Management," *IEEE Network Magazine*, Vol. 15, No. 3, pp. 48-53, May/June 2001.

[24] S. Mascolo, "Smith's Principle for Congestion Control in High-Speed Data Networks," *IEEE Transactions on Automatic Control*, Vol. 45, No.2, pp. 358-364, Feb. 2000.

[25] C. V. Hollot, V. Misra, D. Towsley, and W. Gong, "A Control Theoretic Analysis of RED," *Proc. IEEE Conference on Computer Communications (INFOCOM'01)*, pp. 1510-1519, 2001.

[26] C. V. Hollot, V. Misra, D. Towsley, and W. Gong, "On Designing Improved Controllers for AQM Routers Supporting TCP Flows," *Proc. IEEE Conference on Computer Communications (INFOCOM'01)*, pp. 1726-1734, 2001.

[27] C. V. Hollot, V. Misra, D. Towsley, and W. Gong, "Analysis and Design of Controllers for AQM Routers Supporting TCP Flows," *IEEE Transactions on Automatic Control*, Vol. 47, No. 6, pp. 945-959, June 2002.

[28] S. Kunniyur and R. Srikant, "Analysis and Design of an Active Virtual Queue (AVQ) Algorithm for Active Queue Management," *ACM SIGCOMM'01*, pp. 123-134, 2001.

[29] S. Kunniyur and R. Srikant, "End-to-End Congestion Control Schemes: Utility Functions, Random Losses and ECN Marks," *IEEE Transactions on Networking*, Vol. 11, No. 5, pp. 689-702. Oct. 2003.

[30] S. Kunniyur and R. Srikant, "Stable, Scalable, Fair Congestion Control and AQM Schemes that Achieve High Utilization in the Internet" *IEEE Transactions on Automatic Control* , Vol. 48, No. 11, pp. 2024-2029, Nov. 2003.

[31] S. Kunniyur and R. Srikant, "An Adaptive Virtual Queue (AVQ) Algorithm for Active Queue Management," *IEEE Transactions on Networking*, Vol. 12, No. 2, pp. 286-299, April 2004.

[32] Y. Gao and J. C. Hou, "A State Feedback Control Approach to Stabilizing Queues for ECN-Enabled TCP Connection," *Proc. IEEE Conference on Computer Communications (INFOCOM'03)*, Vol. 3, pp. 2301-2311, April 2003.

[33] L. Zhu, G. Cheng ,and N. Ansari, "Delay Bound of Youngest Serve First Aggregated Packet Scheduling," *IEE Proc. -Commun.*, Vol. 150, No. 1, pp. 6-10, Feb. 2003.

[34] L. Zhu and N. Ansari, "Local Stability of a New Adaptive Queue Management (AQM) Scheme," *IEEE Communications Letters*, Vol. 8, No. 6, pp. 406-408, June 2004.

[35] R. Luijten, C. Minkenberg, and M. Gusat, "Reducing Memory Size in Buffered Crossbar with Large Internal Flow Control Latency," *Proc. IEEE Global Telecommunications Conference (GLOBECOM'03)*, Vol. 3, pp. 3683-3687, 2003.

[36] F. Gramsamer, M. Gusat, and R. Luijten, "Optimizing Flow Control for Buffered Switches," *Proc. IEEE International Conference on Computer Communications and Networks (ICCCN'02)*, pp.438-443, 2002

[37] J. DiStefano, A. Stubberud, and I. Williams, "Feedback and Control Systems," *Schaum's Outline Series, McGraw-Hill, Inc* , 1990.

[38] F. Ren and C. Lin, "Speed up the Responsiveness of Active Queue Management System," *IEICE Trans. Commun.,* Vol. E86-B, No. 2, pp. 630-636, 2003.

[39] N. C. Singer and W. P. Seering, "Preshaping Command Inputs to Reduce System Vibrations," *ASME Journal of Dynamic Systems, Measurement, and Control,* Vol. 112, No. 1, pp. 76-82, 1990.

[40] K. Yoshigoe and K. J. Christensen, "A parallel-polled Virtual Output Queue with a Buffered Crossbar," *Proc. IEEE Workshop on High Performance Switching and Routing (HPSR'01)*, pp. 271-275, May 2001.

[41] K. Yoshigoe and K. J. Christensen, "An Evolution to Crossbar Switches with Virtual Output Queuing and Buffered Cross Points, " *IEEE Network*, pp. 48-56, Sept./Oct. 2003.

[42] R. Rojas-Cessa, E. Oki, Z. Jing, and H. J. Chao, "CIXB-1: Combined Input-One-cell-crosspoint Buffered Switch," *Proc. IEEE Workshop on High Performance Switching and Routing (HPSR'01)*, pp. 324-329, May 2001.

[43] J. G. Dai and S. P. Meyn, "Stability and Convergence of Moments for Multiclass Queueing Networks via Fluid Limit Models," *IEEE Transactions on Automatic Control* , Vol. 40, pp. 1889-1904, 1995.

[44] J. G. Dai and B. Prabhakar, "The throughput of data switches with and without speedup," *Proc. IEEE Conference on Computer Communications (INFOCOM'00)*, pp. 556-564, March 2000.

[45] F. Abel, C. Minkenberg, R.P. Luijten, M. Gusat, and I. Iliadis, "A Four-Terabit Packet Switch Supporting Long Round-Trip Times," *IEEE Micro*, Vol. 23, No. 1 , pp. 10-24, Jan.-Feb. 2003.

[46] T. Javadi, R. Magill, and T. Hrabik, "A High-Throughput Algorithm for Buffered Crossbar Switch Fabric," *Proc. IEEE International Conference on Communications (ICC'01)*, pp.1581-1591, June 2001.

[47] R. Magill, C.E. Rohrs, and R. L. Stevenson, "Output-Queued Switch Emulation by Fabrics with Limited Memory," *IEEE J. Select. Areas Commun.*, Vol. 21, No. 4, pp. 606-615, May 2003.

[48] L. Mhamdi and M. Hamdi, "Practical Scheduling Algorithms For High-Performance Packet Switches," *Proc. IEEE International Conference on Communications (ICC'03)*, Vol. 3, pp. 1659-1663, May 2003.

[49] S-T. Chuang, S. Iyer, and N. McKeown, "Practical Algorithm for Performace Guarantees in Buffered Crossbars," *Proc. IEEE Conference on Computer Communications (INFOCOM'05)*, 2005.

[50] C-S. Chang, W-J. Chen, and H-Y. Huang, "On Service Guarantees for Input Buffered Crossbar Switches: A Capacity Decomposition Approach by Birkhoff and von Neumann," *Proc. IEEE International Workshop on Quality of Service (IWQoS'99)*, pp. 79-86, 1999.

[51] C-S. Chang, D-S. Lee, and Y-S. Jou, "Load Balanced Birkhoff-von Neumann Switches," *Proc. IEEE Workshop on High Performance Switching and Routing (HPSR'01)*, pp. 276-280, May 2001.

[52] C-S. Chang, D-S. Lee, and Y-S. Jou, "Load Balanced Birkhoff-von Neumann Switches, Part I: One-Stage Buffering," *Computer Communications* Vol., 25, pp. 611-622, 2002.

[53] C-S. Chang, D-S. Lee, and C-M. lien, "Load Balanced Birkhoff-von Neumann Switches, Part II: Multi-Stage Buffering," *Computer Communications* Vol. 25, pp. 623-634, 2002.

[54] C-S. Chang, D-S. Lee, and C-Y. Yue, "Providing Guaranteed Rate Services in the Load Balanced Birkhoff-von Neumann Switches," *Proc. IEEE Conference on Computer Communications (INFOCOM'03)*, Vol. 3, pp. 1622-1632, 2003.

[55] I. Keslassy and N. McKeown, "Maintaining Packet Order in Two Stage Switches," *Proc. IEEE Conference on Computer Communications (INFOCOM'02)* , Vol. 2, pp. 1032-1041, June 2002.

[56] R. Schoenen, G. Post, and G. Sander, "Weighted Arbitration Algorithms with Priorities for Input-Queued Switches with 100% Throughput," *Broadband Switching Symposium'99* , 1999.

[57] C-S. Chang, W-J. Chen, and H-Y. Huang "Birkhoff-von Neumann Input Buffered Crossbar Switches," *Proc. IEEE Conference on Computer Communications (INFOCOM'00)*, pp. 1614-1623, March 2000.

[58] L. Mhamdi and M. Hamdi, "MCBF: a high-performance scheduling algorithm for buffered crossbar switches," *IEEE Communications Letters*, Vol. 7, No. 9, pp. 451-453, Sept. 2003.

[59] A. Bianco, M. Franceschinis, S. Ghisolfi, A. M. Hill, E. Leonardi, F. Neri, and R. Webb,"Frame-Based Matching Algorithms for Input-Queued Switches," *Proc. IEEE Workshop on High Performance Switches and Routers (HPSR'02)*, pp. 69-76, May 2002.

[60] R. Rojas-Cessa and E. Oki, "Round-robin Selection with Adaptable-Size Frame in a Combined Input-Crosspoint Buffered Switch," *IEEE Communications Letters*, Vol. 7, No. 11, pp. 555-557, Nov. 2003.

[61] R. Rojas-Cessa, "High-Performance Round-Robin Arbitration Schemes Input-Crosspoint Buffered Switches," *IEEE Workshop on High Performance Switching and Routing (HPSR'04)*, pp. 167-171, 2004.

[62] E. Leonardi, M. Mellia, M. Ajmone Marsan, and F. Neri, "Stability of Maximal Size Matching in Input-Queued Cell Switches," *Proc. IEEE International Conference on Communications (ICC'00)*, Vol. 3, pp. 1758-1763, June 2000.

[63] Z. Guo, R. Rojas-Cessa, and N. Ansari "Packet Switches with Internally-Buffered Crossbars," to appear as a book chapter.

[64] R. Rojas-Cessa and Z. Guo, "Round-Robin Selection with Adaptable Frame-Size for Combined Input-Crosspoint Buffered Packet Switches," accepted and to appear in *IEICE Transaction on Communications*, 2006.

[65] Z. Guo and R. Rojas-Cessa, "Stability Analysis of a Flow Control System for a Combined Input-Crosspoint Buffered Packet Switch," *Conference on Information Sciences and Systems*, March, 2005.

[66] Z. Guo and R. Rojas-Cessa, "Analysis of a Flow Control System for a Combined Input-Crosspoint Buffered Packet Switch," *Proc. IEEE Workshop on High Performance Switching and Routing (HPSR'05)*, pp. 336-340, May 2005.

[67] R. Rojas-Cessa, Z. Dong, and Z. Guo, "Load-Balanced Combined Input-Crosspoint Buffered Packet Switch and Long Round-Trip Times," *IEEE Communications Letter*, Vol. 4, No. 7, pp. 661-663, July 2005.

[68] R. Rojas-Cessa, Z. Guo, and N. Ansari "Combining Distributed and Centralized Arbitration Schemes for Combined Input-Crosspoint Queued Packet Switches," to appear in *Proc. IEEE International Conference on Networks (ICON'05)*, Nov. 2005.

[69] Z. Guo and R. Rojas-Cessa, "A Control Theoretic Analysis of Scheduling and Arbitration Schemes for Packet Switches," accepted in *IEEE Sarnoff Symposium'06*.

[70] A. Kam and K-Y. Siu, "Linear-Complexity Algorithms for QoS Support in Input-Queued Switches with No Speedup," *IEEE J. Select. Areas Commun.*, Vol., 17, No. 6, pp. 1040-1056, June 1999.

[71] Q. Chen and Q. W. W. Yang, "AQM Controller Design for IP routers Supporting TCP Flows Based on Pole Placement," *IEE Proc.-Commun.*, Vol. 151, No. 4,pp. 347-354, 2004.

[72] X. Lin and M. Hamdi, "On Scheduling Optical Packet Switches With Reconfiguration Delay," *IEEE J. Select. Areas Commun.*, Vol. 21, No. 7, Sept. 2003.

[73] L. Mhamdi and M. Hamdi, "CBF: A High-performance Scheduling Algorithm for Buffered Crossbar Switches," *Proc. IEEE Workshop on High Performance Switching and Routing (HPSR'03)*, pp. 67-72, 2003.

[74] L. Mhamdi and M. Hamdi, "Practical Scheduling Algorithm for High-performance Packet Switches," *Proc. IEEE International Conference on Communications (ICC'03)*, pp. 1659-1663, 2003.

[75] J. Li and N. Ansari, "Enhanced Birkhoff-Von Neumann Decomposition Alorithm for Input Queued Switche," *IEE Proc.-Commun.* , Vol.148, No. 6, pp. 339-342, Dec. 2001.

[76] J. Li and N. Ansari, "Credit-Based Scheduling Algorithms for Input Queued Switch," *IEICE Trans. Commun.*, Vol. E85-B, No. 9, pp. 1698-1705, Sept. 2002.

[77] R. Rojas-Cessa and C-B. Lin, "Capture-Frame Eligibility and Round-Robin Matching for Input-Queued Packet Switches," *IEEE Communications Letters*, Vol. 8, No. 9, pp. 585-587, Sept. 2004.

[78] S. Motoyama, D. W. Petr, amd V. S. Frost, "Input-Queued Switch Based on a Scheduling Algorithm," *Electronics Letters*, Vol. 31, No. 14, pp. 1127-1128, July 1995.

[79] A. K. Parekh and R. G. Gallager, "A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Single-Node Case," *IEEE/ACM Trans. On Networking*, Vol. 1, No. 3, pp. 344-357, June 1993.

[80] L. Mhamdi and M. Hamdi, "Output Queued Switch Emulation by a One-Cell-Internally Buffered Crossbar Switch," *Proc. IEEE Global Telecommunications Conference (GLOBECOM'03)*, pp. 3688-3693, 2003.

[81] S-T. Chuang, A. Goel, N. McKeown, and B. Prabhakar, "Matching Output Queueing with a Combined Input Output Queueing Switch," *IEEE J. Select. Areas Commun.*, Vol. 17, pp. 1030-1039, June 1999.

[82] P. Krishna, N. S. Patel, A. Charny, and R. J. Simcoe "On the Speedup Required for Work-Conserving Crossbar Switches," *IEEE J. Select. Areas Commun.*, Vol. 17, No. 6, pp. 1057-1066, June 1999.

[83] E. Altman, Z. Liu, and R. Righter, "Scheduling of an Input-Queued Switch to Achieve Maximal Throughput," *Probability in Engineering and Information Sciences.*, Vol. 14, pp. 327-334, 2000.

[84] J. Schwartzenbach and K. F. Gill, "System Modelling and Control," *Edward Arnold,* 2nd Edition, 1984.

[85] K. S. Narendra and A. M. Annaswamy, "Stable Adaptive Systems," *Prentice Hall*, 1989.

[86] G. F. Franklin and J. D. Powell, "Feedback Control of Dynamic System," *Prentice Hall*, 2002.