Work-conserving WRR-CSVP Resource Allocation in ATM Networks

by

Atsushi Yamada

A thesis

presented to the University of Waterloo

in fulfilment of the

thesis requirement for the degree of

Doctor of Philosophy

in

Electrical Engineering

Waterloo, Ontario, Canada, 1997

The University of Waterloo requires the signatures of all persons using or photocopying this thesis. Please sign below, and give address and date.

Abstract

This thesis investigates resource allocation at an output port of an Asynchronous Transfer Mode (ATM) switch. The resources considered are buffer space and link capacity. The objectives are to provide efficient use of resources to obtain better cell loss performance, to guarantee a minimum amount of resources to protect well-behaved traffic, and to be feasible. To this end, the Complete Sharing with Virtual Partition (CSVP) resource allocation strategy introduced by Wu and Mark [WM95] is applied to both buffer space and link capacity allocations to constitute the Work-conserving Weighted Round Robin-Complete Sharing with Virtual Partition (WRR-CSVP) resource allocation mechanism. This allocation mechanism provides full sharing of buffer space and link capacity among all traffic flows, i.e., adopts a work-conserving queueing discipline, and thus achieves maximum resource utilization. It also possesses the mechanics to guarantee minimum amounts of buffer space and link capacity.

The Work-conserving WRR-CSVP resource allocation mechanism was implemented as an application program with an architecture that permits direct mapping to hardware design, in order to obtain insights for implementation. Then, an emulator was developed by adding peripheral software to it to evaluate system performance such as cell loss, cell delay, and cell delay variation. The efficiency of our resource allocation mechanism and its capability to guarantee a minimum amount of resources are demonstrated by emulation for both single node and end-to-end performance. It is also shown that the virtual partitioning of buffer space can be a useful tool to adjust cell loss performance without affecting cell delay and cell delay

variation. Three implementation issues are raised regarding the detailed algorithms and some alternative mechanisms are proposed. These alternatives are compared for their cell loss performance and design guidelines are provided.

Finally, a hardware design derived from the software implementation is proposed and its feasibility is discussed. Our resource allocation mechanism should be implementable for small switches, such as 4×4.

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Professor Jon .W. Mark, for his advice, guidance, patience, encouragement, and support over the years of my doctoral research. He gave me the warmest encouragement and support during the hardest time of my life. I am honored to have been his student.

I would also like to thank my thesis committee, Dr. G. Agnew, Dr. J. Black, Dr. P. Min, and Dr. B. Preiss, for their support of my research and the numerous suggestions which helped to improve its character.

In addition, I would like to thank the Chair of the Department of Electrical and Computer Engineering, Dr. S. Chaudhuri, and the Associate Chair for Graduate Studies, Dr. A. Vannelli, for their continuous support and encouragement over the years. They stood behind me strongly when I was suffering from Hashimoto's disease.

I owe this unforgettable opportunity to study at the University of Waterloo to Professor T. Hasegawa of Kyoto University and Professor S. Nishio of Osaka University, who introduced and recommended this wonderful university to me originally. Also my appreciation to my former employer, Sumitomo Electric Industries, Ltd., for letting me study at the University of Waterloo and providing me with financial support for the first three years.

My gratitude to my doctor, Dr. T. Stecho, and my counselor, Ms. S. Sundberg, for their professional support both physically and mentally, and their friedship.

Being alone in a foreign country, friendship has been and is the only support I have and the most precious thing to me. In addition to the people listed above, I

have been fortunate enough to make a number of friends, without whose friendship this would not have been possible. Unfortunately, I cannot list the names of all friends because it would make this thesis two volumes. However, I would like to name a few. First of all, I would like to thank the members of the "gang" of our hallway: Bill Anderson, Claude Bergeron, Sai Tak Chu, Bob Lehr, and Biswajit Nandy. Obliged by their unending friendship, Bill and Bob stuck with me to see me through to the end. I would also like to thank the members of our research group: Majid Barazande-Pour, Michael Cheung, Nasir Ghani, B. J. Lee, Hien Nguyen, Jing-Fei Ren, Meenaradchagan Vishnu, Tung Chong Wong, and Vincent Wong. Many thanks to my dearest friends: Dorothy Bonas, Deborah Dennis, Bart Domzy, Verna Friesen, Kei Fukuyama, Claudia Iturriaga, Naree Keh, Christine Lake, Alex Lopez-Ortiz, Brian Mark, Kim Martin, Junko Nakai, James Parker, Ken and Sayuri Ritchie, Wendy Rush, Vanessa Yingling, Steve Woods, and the last but certainly not the least, Mario Ivanic and Christine Susak. In addition, I should not forget to mention my fellow graduate students and the staff of the Graduate Student Association, its Board of Directors, and its International Graduate Student Committee, in which I was honored to be elected as the first Chair. Working with these people gave me enjoyments of the various aspects of student life other than academic activities. I have learned a lot from them.

I would also like to thank the department staff, especially Wendy Boles, who helped me through the bureaucracies of university regulations on numerous occasions, sometimes beyond her duties, with her warm friendship and humor.

Dedication

To my late father, Professor Emeritus Junzo Yamada,
who was looking forward to seeing this thesis,
and would have been the one who recognized the value of the degree and
appreciated the effort most,
with his favorite Scotch whisky, Ballantine's 30 years old, and
cigarette, Hi-Lite.
To my mother and my brother,
without whose unwavering support this work would not have been possible.
And to the memory of my grandmother, Natsu Kiguchi,
who also left us during my time in Waterloo,
who loved me unconditionally.

This thesis is also dedicated to the memory of my best friends, Jim and Diane Ohi, and Tomik Yaghoobian.

Contents

List of Tables

List of Figures

List of Abbreviations

AAL	Asynchronous Transfer Mode Adaptation Layer
ASM	algorithmic state machine
ATM	Asynchronous Transfer Mode
BA-SM	Buffer Allocation State Machine
B-ISDN	Broadband Integrated Services Digital Network
BOSCT	Buffer Over-Subscribing Connection Table
BS	Buffer Space
BSCT	Buffer Subscribing Connection Table
CB-SM	Cell Buffering State Machine
CCITT	Comité Consultatif International de Télégraphique et Téléphonique
CLP	Cell Loss Priority
CMOS	complementary metal-oxide semiconductor
CP	complete partitioning
CS	complete sharing
CSVP	Complete Sharing and Virtual Partition
CX-SM	Cell Transmission State Machine

E/O	electrical-to-optical
Delay-EDD	Delay Earliest-Due-Date
FDDI	Fiber Distributed Data Interface
FFQ	fluid fair queueing
FIFO	first-in-first-out
GFC	General Flow Control
GPS	Generalized Processor Sharing
HDTV	High Definition Television
HEC	Header Error Check
HOL-EDD	Head-Of-the-Line Earliest-Due-Date
HRR	Hierarchical Round Robin
IBP	Interrupted Bernoulli Process
ITU-T	International Telecommunication Union-Telecommunication Standardization Sector
Jitter-EDD	Jitter Earliest-Due-Date
LAN	local area network
LIFO	last-in-first-out
MPEG	Motion Pictures Expert Group
MPEG-1	Motion Pictures Expert Group first-phase
MPEG-2	Motion Pictures Expert Group second-phase
PB	Pre-Buffer
PB-SM	Pre-Buffering State Machine
PGPS	Packetized Generalized Processor Sharing

POT	Pushout with Threshold
PS	partial sharing
PT	Payload Type
QD	queueing discipline
QoS	quality of service
RAM	random access memory
RCS	Rate-Controlled Service
RCSP	Rate-Controlled Static Priority
RISC	reduced instruction set computer
SCFQ	Self-Clock Fair Queueing
SM	state machine
SS-SM	Service Scheduling State Machine
VC	virtual circuit
VCI	Virtual Channel Identifier
VCT	Virtual Circuit Table
VLSI	very large scale integration
VPI	Virtual Path Identifier
WAN	wide area network
WCD	work-conserving discipline
WF^2Q	Worst-case Fair Weighted Fair Queueing
WRR	Weighted Round Robin
XB	Transmission Buffer

Chapter 1

Introduction

1.1 Background

The necessity of high speed public network service to support forthcoming multimedia applications was predicted in the 1980's and the concept of the Broadband Integrated Services Digital Network (B-ISDN) was established (See, e.g., [Han89, Kle91]). In fact, the recent emergence of the popular internet services proved the insufficient bandwidth of current public telecommunication services. The demand for larger bandwidth is higher than ever.

Asynchronous Transfer Mode (ATM) has been adopted as the protocol standard for the B-ISDN by the Comité Consultatif International de Télégraphique et Téléphonique (CCITT), which is now the International Telecommunication Union-Telecommunication Standardization Sector (ITU-T) [CCI89]. The protocol reference model [CCI91a], the specification of ATM Layer [CCI91b], the functional description of ATM adaptation layer (AAL) [CCI91c], and the specification of AAL

[CCI91d] were published by the CCITT in 1991. The ATM Forum was formed to further pursue the details of the protocol standard contained in "ATM User-Network Interface Specification" [For95]. In the meantime, some commercial products of the local area network (LAN) version of ATM have already launched into the market. Reports of actual use of ATM networks can be found in [NGT+95, KKM+97].

The heterogeneous services supported by ATM networks include applications such as packetized voice, image, video, electronic message, transaction data, and file transfer, which create various types of traffic flows. Often, these different applications have varied requirements of quality of service (QoS), such as cell loss ratio, cell delay, and cell delay variation. For example, packetized voice and video transmission services may require a certain delay bound while being able to tolerate a certain amount of loss, whereas transaction data and file transfer services may require very small loss ratio but can tolerate some delay. It is one of the goals of ATM networks to provide guaranteed QoS to users while achieving the best network efficiency possible. In order to accomplish this goal, network resources at multiplexers and switches have to be carefully managed. This work is concerned with resource allocation at multiplexers and switches, paying attention mainly to cell loss performance.

1.1.1 ATM Networks

An ATM network is a mesh network, where ATM multiplexers and demultiplexers are located at the edge of the network, and the ATM switches in the interior of

the network. User traffic enters an ATM network from an ATM multiplexer, may traverse ATM switches, and exits at an ATM demultiplexer.

The architecture of ATM is built to support various user applications in the same structure. Various user applications generate traffic flows with variant characteristics. A small fixed size block, referred to as a *cell*, is adopted as the transmission unit of ATM networks in order to absorb the variations.

Some applications create intense traffic for a short period of time but do not utilize the bandwidth of the network throughout the call duration. Taking this into consideration, the architecture of ATM is aimed at achieving bandwidth efficiency by statistically multiplexing such bursty traffic flows in order to support more users, which is referred to as *multiplexing gain*. Buffering of cells at the ATM multiplexers and the ATM switches is necessary to realize statistical multiplexing.

An ATM cell is 53 bytes, which consists of a 5-byte header field and a 48-byte information field. The structure of an ATM cell is illustrated in Figure 1.1 [CCI91b, For95]. The mode of service is connection-oriented and switching and multiplexing are cell-oriented. An ATM connection, virtual circuit (VC), is identified by a pair of Virtual Path Identifier (VPI) and Virtual Channel Identifier (VCI).

1.1.2 ATM Switch

At an ATM switch, the paths of cells from input ports to output ports may conflict internally. Also, more than one cell arriving at the input ports may be destined for the same output port and create output conflicts. These switching conflicts may cause blocking of cells. It is one of the objectives of ATM switches to route cells

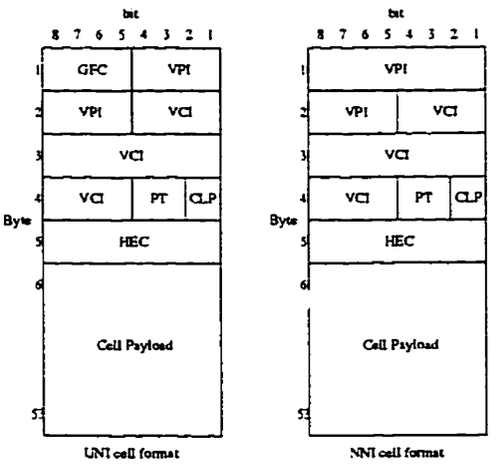

Figure 1.1: ATM cell format

arriving at the input ports successfully to the destination output ports. In order to solve output conflict, buffering of cells is necessary since the capacities of an input link and an output link are the same. Buffering of cells is also needed to obtain multiplexing gain, as mentioned in the previous subsection. The cells in the buffer have to be transmitted to the output link in some order. Thus, the key functions of an ATM switch are routing, buffering, and scheduling.

The design of an ATM switch which minimizes the blocking of cells due to switching conflict has been a major challenge in the realization of ATM networks. In fact, the ATM switches currently in the market are non-blocking. A number of ATM switch designs have been proposed, focusing mainly on routing. The various design approaches in the early years are summarized in [Tob90]. The shared medium type switches, such as the Synchronous Composite Packet Switching [TYN+87], are built on a bus or ring structure and use a medium sharing method such as time division

multiplexing. This architecture is simple and provides non-blocking. However, the shared medium architecture faces significant difficulties in increasing the internal speed to handle a large number of input ports. The shared memory type switches, such as the Prelude switch [DCS88], the general-purpose ATM switch [KSC91], and the Shared Buffer Memory Switch [EKO+93], also realize non-blocking. However, this architecture is known to be unsuitable for a large switch [AD88].

Larger switches are needed for the wide area network (WAN) version of ATM. The space-division type switch, in which the routes for cells from the input ports to the output ports are established simultaneously, may be the solution to realize a large non-blocking switch. A banyan network is a multistage interconnection network where there exists exactly one path from any input port to any output port. Well-known interconnection networks such as omega, flip, cube, shuffle-exchange, and baseline belong to the class of banyan networks. In banyan networks, not only blocking due to output conflict but also internal blocking is inherent; the contention of a link inside may occur due to the confliction of paths of input-port-output-port pairs. In order to increase throughput of the banyan based switches, buffers may be placed internally such as the Integrated Service Packet Network [Tur86]. Performance of the buffered banyan switches is studied in [KLG90b]. Internal blocking can be avoided by a particular interconnection of banyan network, which is referred to as a sort-banyan network [KLG90a, Hui91]. Internal blocking can also be avoided by adding a Batcher sorter network in front of a banyan network. This Batcher-banyan is adopted in switches such as the Sunshine switch [GSL94], the MULTI-PAC switch [Tur90], and the Shared Concentration and Output Queueing switch

[CM93, CM94]. The Pipeline banyan switch [WY95] consists of a control plane and data planes, which are of the same banyan topology; the control plane is used for path reservation while the actual cell transmission is performed on data planes. The Shuffleout [BDGP94, DGP94] uses an interconnection of different switching elements from those used in banyan networks to prevent internal blocking without internal buffering. A crossbar switch, such as the Bus Matrix Switch [NTFH87], consists of an array of cross points, each of which corresponds to an input port-output port pair. In a Knockout switch, a broadcast bus from each input port is connected to the output ports by dropping lines to achieve a complete interconnection of input port-output port pairs [YHA87, EYH87, Cha91b, MLLK95, KL95]. By allowing multipaths from input ports to the destination output ports, both internal and output blockings can be avoided. Such an architecture is proposed in [LS93b, WLG94, Kim94, JU95, MSH95].

The location of buffers has a significant impact on switch performance. Between input and output queueing, the output queueing approach has been shown to be more efficient [KHM87, HK88]. Therefore, output queueing is generally preferred. There is another reason to favor output queueing. It is known that a sophisticated service scheduling scheme is necessary to guarantee delay QoS, as we further discuss in the following subsection. Output queueing provides the flexibility to implement such a scheduling scheme: an output port system consisting of a buffer allocation controller and a service scheduler can be adopted, if a fast non-blocking switching fabric, where cells are routed from input ports to output ports without blocking, is available. Such an architecture can be seen in [Cha91a, CU95, LLG96]. By

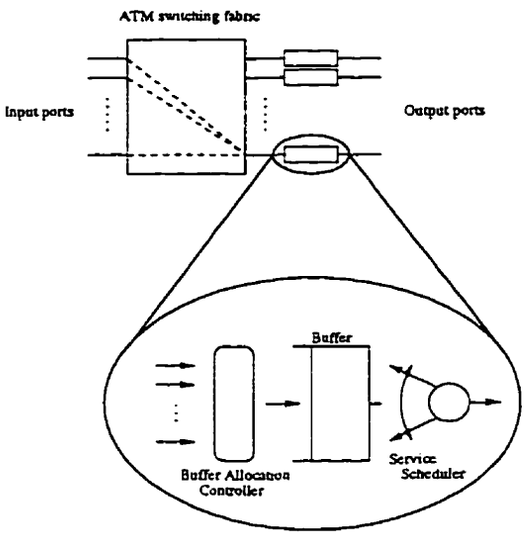

Figure 1.2: An output buffering ATM switch

assuming a non-blocking switching fabric, this work considers such an output port system of an ATM switch illustrated in (Figure 1.2).

1.1.3 Service Scheduling Schemes

The average delay and delay variation have been the issue of study on traditional communication networks, which provide best effort services. However. ATM networks intend to provide QoS guarantee; delay sensitive sources such as real-time applications may require bounded delay and delay variation. This calls for sophisticated scheduling of cells for transmission.

One of the objectives of scheduling schemes proposed for ATM switches is to partition the link capacity and provide the required bandwidth to each VC so that the delay and the delay variation bounds are provided. It is a challenging task

to reach this objective since ATM networks are slotted time systems due to the fixed size cells. In order to support as many VCs as possible or to offer better performance to each VC, it is generally preferred for scheduling schemes to be efficient in the utilization of the link capacity. The scheduling scheme also needs to be computationally inexpensive in order to cope with the speed of ATM networks and to support a number of VCs.

A number of scheduling schemes have been proposed and are summarized in [Zha95], where the scheduling schemes are categorized into the classes of work-conserving schemes and non-work-conserving schemes. When a server never becomes idle as long as packets (or cells) exist in the queue (or buffer) to transmit, the queueing system is said to be *work-conserving* (See, e.g., [Kle76, KK77]). Therefore, the work-conserving scheduling schemes maximally utilize link capacity. With a non-work-conserving discipline, on the other hand, the server may become idle even when there are cells in the buffer waiting to be transmitted. Therefore, the non-work-conserving schemes may not maximally utilize the link capacity. However, generally speaking, the non-work-conserving scheduling schemes adopt computationally less expensive algorithms, such as fixed time slot assignments, than the work-conserving schemes.

The class of work-conserving schemes includes VirtualClock [Zha91], Packetized Generalized Processor Sharing (PGPS) [DKS90, PG93], Worst-case Fair Weighted Fair Queueing (WF^2Q) [BZ96], Self-Clock Fair Queueing (SCFQ) [Gol94], Delay Earliest-Due-Date (Delay-EDD) [FV90, ZS94], and Head-Of-the-Line Earliest-Due-Date (HOL-EDD) [VM96, Vis96]. VirtualClock scheme aims to emulate the time

division multiplexing. PGPS and WF^{2}Q attempt to approximate fluid fair queueing (FFQ) or Generalized Processor Sharing (GPS) (See, e.g., [Kle76]). Unlike PGPS or WF^2W, SCFQ approximates FFQ without maintaining a reference FFQ. In Delay EDD, which is an extension to the conventional Earliest-Due-Date-First scheduling, a deadline is assigned to each cell to provide bounded delay. HOL-EDD attempts to transmit cells belonging to a VC in a constant interdeparture time without time-stamping. For these work-conserving scheduling schemes, efforts have been made to obtain not only a single-node delay bound but also an end-to-end delay bound for some classes of input traffic such as the (σ, ρ) traffic model [Cru91a] or the $(X\text{min}, X\text{ave}, I, S\text{max})$ traffic model [FV90] (See, e.g., [PG94, Gol95, Zha95, Vis96]). These work-conserving scheduling schemes require sorted priority queue algorithms and thus may become computationally expensive when a large number of VCs are supported. Some of these schemes also require floating point calculations.

The non-work-conserving schemes may not utilize the link capacity as efficiently as the work-conserving schemes but have their advantages. Not only do they guarantee delay bound but they also bound the output traffic so that the end-to-end delay is bounded for a more general class of traffic than those used in the analysis of the work-conserving scheduling schemes. In addition, as already mentioned, they usually adopt simpler algorithms. The class of non-work-conserving schemes includes Weighted Round Robin (WRR) [KSC91, RRA95], Hierarchical Round Robin (HRR) [KKK90], Stop-and-go [Gol90], Jitter Earliest-Due-Date (Jitter-EDD) [VZF91], and Rate-Controlled Static Priority (RCSP) [ZF93]. In WRR, the time axis is divided into fixed length frames of S time slots. The

kth VC receives S_k time slots for every frame. Thus the transmission rate given to this VC, C_k is given by $C_k = C \times S_k/S$, where C is the link capacity. Since S and S_k are integers, the capacity allocation granularity to provide C_k is limited. However, since the time slot allocation is fixed and no comparison is required, the WRR is seen to be one of the computationally least expensive schemes. HRR introduces hierarchical structure into WRR to improve the granularity of capacity allocation. Stop-and-go scheme utilizes another framing strategy, where arriving frames are defined for an input link, and departing frames for an output link; the arriving frames are mapped to the departing frames by introducing a constant delay; the transmission of cells that have arrived during a frame is always postponed until the beginning of the next frame to ensure the cells on the same frame at the source stay in the same frame throughout the network. In this scheme, synchronization between input and output links is necessary, and thus it may be difficult to actually implement this scheme since generally, input and output links are not synchronized in ATM networks. Jitter-EDD extends Delay-EDD to provide delay variation bounds by time-stamping the difference between the deadline and the actual finishing time. A RCSP server consists of a rate-controller and a static priority scheduler, and provides flexibility in the allocation of delay and bandwidth.

A class of scheduling schemes referred to as Rate-Controlled Service (RCS) was introduced by Zhang and Ferrari [ZF94]. The non-work-conserving scheduling schemes listed above belong to this class. The end-to-end delay bound of the scheduling schemes belonging to the RCS is studied in [ZF94, Zha95, GGPS96]. The non-work-conserving schemes can be easily modified to be work-conserving. It

is shown in [GGPS96] that the RCS schemes created from the non-work-conserving schemes have the same end-to-end delay bound as the original RCS schemes.

By focusing on two parameters, the latency and the allocated rate, a class of scheduling schemes referred to as Latency-Rate Servers was introduced by Stiliadis and Varma, and the end-to-end delay bound was derived [SV96]. This class includes scheduling schemes such as VirtualClock, PGPS, SCFQ, and WRR.

There are a few approaches to control cell loss performance by scheduling, such as the threshold-based priority scheme in [LS93a] and the work-conserving schemes proposed in [LS96b].

It should be noted that in order to provide VC based delay performance guarantee by the scheduling schemes listed in this subsection, it is necessary to maintain VC based queues of cells in the buffer space. Therefore, the proposals of these scheduling schemes assume complete partitioning buffer allocation and some also assume infinite buffer space for analysis.

1.1.4 Buffer Allocation Schemes

As already mentioned, buffering of cells is necessary at an output port of a non-blocking ATM switch not only to support arrivals destined to the same output port but also to sustain bursts and obtain statistical multiplexing gain. Buffer allocation for ATM switches is often discussed for the shared memory type switch architecture, where the buffer space is shared among the output ports (See, e.g., [CGGK95, LS96a]). This work, however, considers the buffer sharing among the traffic flows, the VCs, at an output port.

The buffer size cannot be infinite and shortage of buffer space leads to a high loss probability. Therefore, efficient use of the finite buffer space has been one of the most significant issues of study in traditional communications networks. The complete sharing (CS), the complete partitioning (CP), and the partial sharing (PS) buffer allocation schemes are well studied for Poisson arrivals and exponential service time in [KK80]. The Sharing with Maximum Queue Length, the Sharing with a Minimum Allocation, and the Sharing with a Maximum Queue and Minimum Allocation strategies are discussed for the PS scheme.

ATM networks impose new challenges. The bursty traffic of ATM networks and the low cell loss requirements of the users demand a large buffer space and thus an efficient use of the buffer space. The efficiency is accomplished by sharing the buffer space. However, when the buffer space is shared by two or more traffic flows, a bursty traffic flow may momentarily over-subscribe the buffer space and cause unnecessary cell loss for a non-bursty traffic flow since such a traffic flow may require a certain amount of buffer space constantly. In order to prevent this from occurring, a mechanism to guarantee a minimum amount of buffer space is necessary. In addition, since ATM is connection-oriented, the order of cells belonging to the same traffic flow has to be preserved.

The protocol of ATM specifies the Cell Loss Priority (CLP) field in the header of an ATM cell (See Figure 1.1). Thus, providing a very low loss ratio to the high priority cells designated by the CLP field has been the main focus of study in buffer allocations for ATM switches.

Generally speaking, there are two kinds of cell prioritization strategies. One is

to divide traffic flows into two priority classes, high and low, depending on the cell loss sensitivity of the source applications. Cell loss prioritization for this strategy can be accomplished by a scheduling scheme using two priority queues assigned to the two priority classes (See, e.g., [LS93a]). The other prioritization strategy is to give higher priority to the significant cells in the same traffic flow. In this case, the two priority queue mechanisms above may not be able to preserve the order of cells. Some other mechanism is necessary.

A shared buffer constructed by a single first-in-first-out (FIFO) queue maintains the order of cells while achieving buffer efficiency. The cells are transmitted by the Head of Line order of the FIFO queue. Several priority mechanisms have been proposed using this single FIFO queue. Nested thresholds are given to the FIFO queue to restrict the admission of cells with different priorities in [PF91]. In [LS91] and [KHBG91], a PS scheme and a mechanism referred to as the pushout scheme are discussed. In this PS scheme, the first T space of the FIFO queue is shared and the rest of space is dedicated for higher priority cells, i.e., when the total buffer occupancy exceeds T, only high priority cells are admitted. The scheme using the pushout mechanism allows traffic flows to fully share the buffer space; when the buffer becomes saturated, higher priority arrivals are admitted by pushing out lower priority cells in the buffer space, i.e., the FIFO queue, while lower priority arrivals are blocked. A PS strategy where a maximum of L space is allowed to the low priority cells is discussed in [CGK94]. A temporary buffer can be placed between the switching fabric and the actual buffer space to examine the admission of cells arriving in the same time slot. A pushout scheme where low priority cells may be

pushed out if the buffer space is over-subscribed by a VC even if the entire buffer space is not full is introduced and compared with the other PS schemes and the pushout scheme described previously in [THP94]. The performance of the pushout buffer is studied in [CT94] and [KW94].

The queue manager in [CU95] deals with the cell loss and the cell delay priorities using several queues of different priority. The server transmits the cells according to the priority given to the queues. The cell admission adopts a pushout mechanism with priorities: a cell of a higher priority may be admitted by pushing out a cell in a lower priority queue. Hardware implementation is also discussed. Note that this queue manager cannot preserve the order of cells when a different priority is given to the cells belonging to the same traffic flow.

The service scheduling schemes discussed in the previous section require VC queues. The single FIFO queue approach described above is not suitable for such scheduling schemes. VC queues can be easily established by the CP scheme. However, since VC traffic may be very bursty, a significantly large buffer space may be necessary, while the space may not be utilized most of the time since the silence period may be very long. The CS scheme may be a choice for efficiency. However, if the bursts of two different traffic flows collide, it may lead to a large loss rate for one of these two traffic flows. Or if there is a fairly constant traffic flow, this flow may suffer from a high cell loss by the bursty traffic even if this flow does not require a large buffer space. The *Complete Sharing and Virtual Partition* (CSVP) introduced by Wu and Mark [WM95] is a better mechanism to handle VC queues. The CSVP buffer allocation scheme fully utilizes the buffer space by complete shar-

ing while guaranteeing a minimum buffer space allocated by the virtual partition using a pushout mechanism. In [WM95], the characteristics of the CSVP buffer allocation scheme are analyzed for the multiplexing of two classes of Markov fluid processes using a fluid-flow model. The Pushout with Threshold (POT) policy in [CGGK95] is the CSVP scheme applied to buffer sharing among the output ports of a 2×2 shared memory type switch. The switch with POT policy is shown to achieve an optimal throughput for Poisson arrivals and exponential service time.

1.1.5 End-to-end Performance Evaluation

The QoS provided to a VC is defined on an end-to-end basis. Therefore, it is important to assess end-to-end performance of ATM networks. Simulation is a straightforward approach to evaluate end-to-end performance of ATM networks although it may be time consuming. Simulations of a tandem single-server queue model to evaluate end-to-end performance of ATM networks are conducted in [Gru91, YKF91]. More practical traffic is considered in [NLG96]. A simulation study of a more complex topology is performed in [Fri91].

ATM networks provide connection-oriented services. Therefore, end-to-end performance analysis of an ATM network would involve investigations of the behavior of a connection. This behavior can be studied by a tandem connection of single-server queues, where a single server models an output port of a switch in the physical chain. Such an approach is used to analyze conventional narrowband packet switching networks with Poisson input [Kle76, Hay84]. In [Fis91], an approximation approach using a $GI_i/GI_i/1$ model is applied. In [OMM91], the per-flow

interdeparture-time distribution is derived for multiplexing of Interrupted Bernoulli Processes and the departure process is applied as input traffic at the downstream node. In [SBPD93], a $Geo/D/1$ model is used. The traffic characteristics of the interior of an ATM network are also studied in [DM95]. End-to-end delay variation is studied by characterizing the departure traffic in [MSB97]. In [RMW94], it is reported that the departure process in response to an ON-OFF input, where the ON and OFF periods are geometrically distributed, can be approximated by another ON-OFF model with different parameter values although the ON and OFF periods are no longer geometrically distributed.

A merging and splitting technique for the analysis of networks with two-state Markov models was originally introduced in [Vit86] and was extended to more complex source models and splitting mechanisms in [Sta91]. This approach can be seen as another way to decompose a network. In this approach, a network is decomposed into the merging points and splitting points. However, the application of this technique to the cell streams of ATM networks has not been solved.

Upper bound analysis is often used to evaluate cell delay and cell delay variation. As already mentioned, for the purpose of QoS guarantee, the bounds of end-to-end delay and delay variation are the significant performance measures. Cruz introduced a new approach for end-to-end performance analysis which considers delay bound instead of traditional mean delay [Cru91a, Cru91b]. This approach is extended to tail distribution analysis by [Kur92]. Yaron and Sidi applied this method to more general source models [YS93]. The performance analysis of the work-conserving scheduling schemes introduced in the previous subsection also uses Cruz's approach.

However, this approach is not applicable to loss evaluation.

1.2 Motivation, Objectives, And Methodology

Non-blocking ATM switching fabrics have started to become a reality and resource allocation at an output port to provide QoS guarantee is now a significant issue of study. Although mechanisms to provide guaranteed delay and delay variation performance have been well studied and a number of scheduling schemes have been proposed, a mechanism to provide low cell loss performance is yet to be further discussed. The bursty traffic and the low cell loss requirements of ATM networks, together with QoS guarantees, call for an efficient use of resources, i.e., buffer space and link capacity, while guaranteeing of the minimum amount of resources is essential to protect well-behaving users. It is important to find a feasible resource allocation scheme at the output ports of ATM switches, which allocates both buffer space and link capacity to satisfy these criteria above.

The results by Wu and Mark [WM95] show that the CSVP buffer allocation scheme obtains efficient use of buffer space and robustness of buffer allocation to guarantee a minimum buffer space. This scheme is capable of providing VC based queues and thus is suitable for the scheduling schemes proposed for ATM switches. If efficient use of link capacity and a guarantee of a minimum amount of transmission capacity can be achieved by applying the CSVP strategy to capacity allocation, a resource allocation mechanism constructed by combining this scheduling scheme with the CSVP buffer allocation scheme may be the solution for low cell loss services. To this end, we propose the Work-conserving WRR-CSVP

resource allocation mechanism, which provides maximum resource utilization since it is work-conserving, while guaranteeing a minimum amount of buffer space and transmission capacity. Furthermore, if this Work-conserving WRR-CSVP resource allocation mechanism can be made feasible at an output port of an ATM switch, it will provide a significant impact on the design of ATM switches. Therefore, it is necessary to provide detailed feasible algorithms for implementation.

In order to investigate the performance and feasibility of the Work-conserving WRR-CSVP resource allocation mechanism, an emulation approach is adopted in this study. The Work-conserving WRR-CSVP resource allocation mechanism is implemented as a program on a workstation in order to obtain insights for implementation. Then, an emulator is constructed by developing peripheral software to evaluate the cell loss, cell delay, and cell delay variation performances of the Work-conserving WRR-CSVP resource allocation mechanism. The cell loss performance of the CSVP scheme was analyzed in [WM95] using a fluid-flow model. For tractability, the analysis is limited to a single-node case with two traffic flows. The emulation approach allows us to investigate the performance for more than two traffic flows and to evaluate other performance measures, such as cell delay and cell delay variation, which were not available in [WM95], although our main focus is on the cell loss performance. Furthermore, not only is the performance of a single node studied, but also the end-to-end performance.

The software design is mapped to a hardware design specification. The feasibility is determined by examining the hardware design.

1.3 Contributions

The major contributions of this dissertation are: (i) the proposal of the Work-conserving WRR-CSVP resource allocation mechanism; (ii) software implementation of the mechanism with an architecture that permits direct mapping to hardware design; and (iii) hardware design and implementation specification. The Work-conserving WRR-CSVP provides a feasible resource allocation for ATM multiplexers and switches. This mechanism maximally utilizes available buffer space and link capacity while guaranteeing a minimum amount of buffer space and transmission capacity to each traffic flow. By developing an emulator, not only is an understanding of various performance aspects obtained but also insights for implementation of the mechanism are acquired. From these insights, a hardware design is proposed and the conditions for implementation are clarified.

1.4 Scope

In Chapter 2, the concept of the CSVP resource allocation strategy is introduced. This concept is applied to both buffer and capacity allocation to construct the Work-conserving WRR-CSVP resource allocation mechanism investigated in detail in this thesis. Then, the performance measures are defined and the queueing behavior is studied. In Chapter 3, the detailed algorithms for the Work-conserving WRR-CSVP resource allocation mechanism are described and the development of an emulator is discussed. Three implementation issues are raised regarding the detailed algorithms and some alternative mechanisms are proposed. The performance

of a single node is studied by emulation in Chapter 4. Also, alternatives proposed in Chapter 3 are examined to provide design guidelines in Chapter 4. The emulator is extended to examine the end-to-end performance in Chapter 5, using Motion Pictures Expert Group (MPEG) video data as test sources. Chapter 6 proposes the implementation strategies for the Work-conserving WRR-CSVP resource allocation mechanism and discuss its feasibility. Concluding remarks are presented in Chapter 7.

Chapter 2

The Work-conserving WRR-CSVP

2.1 Introduction

2.1.1 Generic Node

As already mentioned in Chapter 1, non-blocking ATM switching fabrics have already started to appear in the market. Therefore, assuming such a switching fabric, we consider an output-buffering non-blocking ATM switch. Our focus is on the mechanisms between an output port of a switching fabric and an output link, where cells from the switching fabric are buffered and transmitted to the output link. The essential function of this buffered output port system is queue management and servicing of multiple traffic flows. Note that this system uses the same resource allocation model as an ATM multiplexer. Thus, we refer to this as a

generic node. The resource of a generic node is a combination of buffer space and link capacity.

An ATM link is a slotted channel. Therefore, it is appropriate to assume a slotted time system, where a time slot is a time interval long enough to transmit one cell. In this chapter, a queueing model of the output port system, or a generic node, is formulated as a discrete-time system. The performance measures are also defined in this chapter.

2.1.2 The CSVP Resource Allocation Strategy

The objectives of the resource allocation mechanism investigated in this work are: to utilize resources efficiently and to guarantee a minimum amount of resources. If resources are utilized efficiently, a large number of traffic flows can be supported, or lower cell loss ratio services can be provided for the same traffic conditions, given the same amount of resources. By guaranteeing a minimum amount of resources, a maximum level of cell loss can be assured. For given finite amount of resources, if the resources are shared by all traffic flows, it is maximally utilized and a minimum aggregate cell loss ratio is achieved (See, e.g., [CR87]). The cell loss ratio of bursty traffic is also expected to improve owing to the multiplexing gain attained by complete sharing. Let us refer to this as an *efficient* use of resources. As discussed in Section 1.1.4, however, if the resources are completely shared without any restriction, some traffic flows may suffer from unnecessarily high cell loss ratios. This occurs because a bursty traffic flow may over-subscribe the buffer space and cause a momentary lack of buffer space for another traffic flow. When the finite resources

are completely partitioned, the resources may not be maximally utilized. However, a minimum amount of resources is guaranteed to each traffic flow. Let us refer to this as a *robust* allocation of resources.

In the CSVP resource allocation strategy introduced in [WM95], the resources are shared completely and maximally utilized while a minimum amount of resources specified by the virtual partition is guaranteed to each traffic flow. Let K be the number of traffic flows supported by a generic node. The CSVP resource allocation strategy is defined as follows:

Definition 1 (CSVP Strategy)

In the CSVP resource allocation scheme,

1. *the total resources are partitioned into K segments by* virtual partition *according to the traffic loads (or some measurement or estimation).*

2. *under-utilized segments of resources can be utilized by an over-subscribing flow, and*

3. *this over-subscribed portion of resources will be returned to the under-utilizing flows when they need the over-subscribed portion.*

2.2 The Work-conserving WRR-CSVP Resource Allocation Mechanism

Let us consider a generic node characterized by a buffer of B spaces and a service capacity of C cells/s.

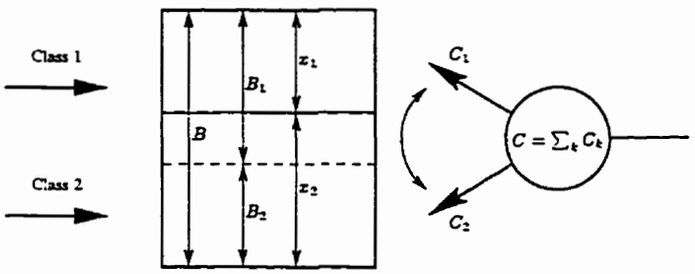

Figure 2.1: The CSVP buffer allocation scheme

2.2.1 The CSVP Buffer Allocation Scheme

In [WM95], the CSVP strategy is applied to buffer allocation using a pushout operation. The CSVP buffer allocation scheme, illustrated in Figure 2.1, is described as follows [WM95]:

CSVP Buffer Allocation Scheme:

1. The total buffer size B is divided into K segments, B_k, $k = 1, 2, \cdots, K$, such that $B = \sum_{k=1}^{K} B_k$.

2. The entire buffer space is shared by K traffic flows. Therefore, when the buffer is not full, cells belonging to any traffic flow are accepted upon arrival.

3. When the buffer is full, one of the traffic flows, say the ith flow, may occupy fewer spaces than B_i. Then there is at least one traffic flow, say the jth flow, that must be occupying more than B_j spaces. The admission policy will admit a newly arriving cell belonging to the ith traffic flow by pushing out a cell belonging to the jth traffic flow from the buffer, but will reject a newly arriving cell of the jth traffic flow.

The pushout operation is illustrated in Figure 2.2.

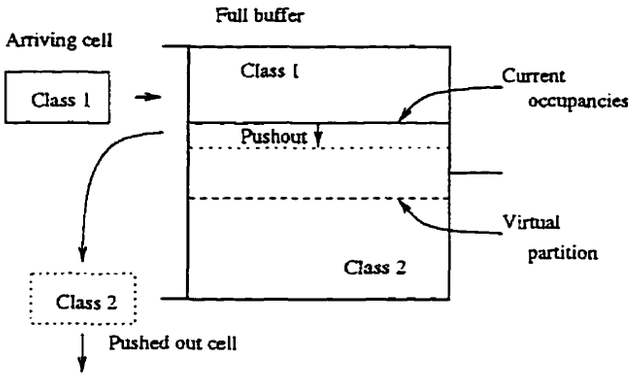

Figure 2.2: The pushout operation of the CSVP buffer allocation scheme

When there are only two traffic flows supported, if a traffic flow is under-subscribing the buffer space in a full buffer, the other traffic flow is the only traffic flow which is over-subscribing the buffer space. Thus, the description above was sufficient for the analysis in [WM95], where only the case of two traffic flows was considered. However, the existence of more than two traffic flows complicates the problem: there may be two or more traffic flows over-subscribing the buffer space and thus some mechanism is necessary to determine a traffic flow to be pushed out. This selection mechanism is further discussed in Section 3.3.2.

2.2.2 The WRR Scheduling Scheme

If the CSVP strategy can be applied to capacity allocation, it may form a suitable scheduling scheme to incorporate with the CSVP buffer allocation scheme. The objectives for the scheduling scheme here are to guarantee a minimum bandwidth to each traffic flow and to maximize the utilization of total link capacity. It is im-

portant for the resource allocation mechanism, a combination of a buffer allocation scheme and a scheduling scheme, to be feasible at an output port of an ATM switch. Therefore, a computationally less expensive scheduling scheme is preferable, considering the computational cost of the CSVP buffer allocation scheme (See Chapter 6 for further discussion of computational cost).

As discussed in Section 1.1.3, the WRR scheduling scheme described in [KSC91, RRA95] is one of the computationally least expensive scheduling schemes proposed for ATM switches, which provides a minimum bandwidth guarantee and a delay bound to each traffic flow. Another advantage of the WRR scheme is that the output traffic is bounded and thus the traffic characteristics in the interior of the network become predictable [RRA95]. In fact, the end-to-end delay is bounded [Zha95, GGPS96]. One of the most important drawbacks of the WRR scheduling scheme is the lack of the granularity of capacity allocation as pointed out in Section 1.1.3.

The WRR scheduling scheme provides cell transmissions on a cyclic basis, where the cycle time is S slots.

WRR Scheduling Scheme:

1. The service cycle, S, is divided into S_k, $k = 1, 2, \cdots, K$, satisfying $S = \sum_{k=1}^{K} S_k$.

2. For every S time slots, S_k slots are allocated to the kth traffic flow.

3. At most one cell belonging to the traffic flow to which the time slot is assigned is transmitted.

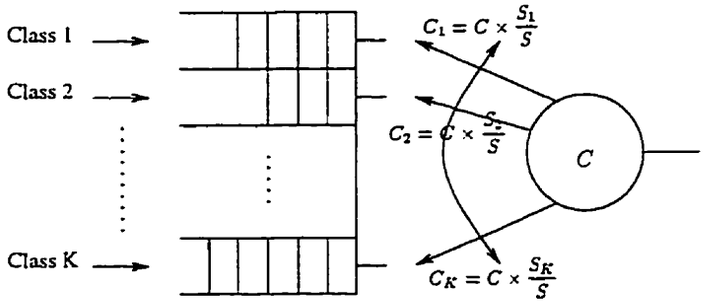

Figure 2.3: The WRR scheduling scheme

4. If there is no cell belonging to the traffic flow to which the time slot is assigned, no cell is transmitted.

By this capacity allocation, the kth traffic flow receives a service capacity of C_k cells/s given by

$$C_k = C \times \frac{S_k}{S}. \quad k = 1, 2, \cdots, K \tag{2.1}$$

The WRR scheduling scheme can be seen as a CP resource allocation strategy applied to capacity allocation. The WRR scheduling scheme is illustrated in Figure 2.3

The WRR scheduling scheme belongs to the class of non-work-conserving schemes: no cell transmission may take place even when there is a cell in the buffer waiting to be transmitted. A time slot is said to be *wasted*, if the server is idle when the buffer is not empty. How many time slots are actually wasted by the WRR scheduling scheme? Our emulation study indicates that a fairly large number of time slots (10 to 20 % of the unused time slots) are wasted by the WRR scheduling scheme (See

Section 4.3.1 and Appendix B). Can we make the WRR scheduling scheme, which is a CP capacity allocation, more efficient by applying the CSVP strategy? The WRR scheduling scheme can be made more efficient by reassigning the wasted slots to the other traffic flows. This mechanism is achieved by modifying Item 4 of the WRR scheduling scheme as follows:

4. If there is no cell belonging to the traffic flow to which the time slot is allocated, a cell belonging to another busy traffic flow is transmitted.

This enhanced scheduling scheme can be seen as an application of the CSVP strategy to capacity allocation: the time slot reassignment operation enables the system to fully utilize the service capacity while the WRR scheduling scheme guarantees a minimum capacity of C_k to the kth traffic flow. Note that this enhanced scheduling scheme belongs to the class of work-conserving schemes. Therefore, this scheme is referred to as the *Work-conserving WRR* scheduling scheme. The addition of the time slot reassignment operation preserves the delay bound by the WRR scheduling scheme [GGPS96]. However, it should be noted that the output traffic is no longer bounded and thus the cell loss performance inside the network may be unpredictable.

Similar to the pushout operation, further consideration is necessary when more than two traffic flows are supported. In such a case, there may be two or more traffic flows occupying the buffer space and thus eligible to receive the reassigned time slot. A mechanism is necessary to determine which traffic flow is to receive the reassigned time slot. This mechanism is further discussed in Section 3.4.3.

2.2.3 The Work-conserving WRR-CSVP Mechanism

A *queueing discipline* (QD) specifies both buffer allocation and service scheduling schemes. The work-conserving property of queueing systems (See, e.g., [Kle76. KK77]) is specified for the QD's with deterministic service time and finite buffer space, and a class of QD's referred to as *work-conserving discipline* (WCD) is defined by Clare and Rubin [CR87] as follows:

Definition 2 (Work-conservation [CR87])
Let $\mathcal{D}$ denote a set of QD's where cell loss ratio exists. A work-conserving discipline (WCD) is a QD in $\mathcal{D}$ for which:

1. *no cell is blocked from entry unless the buffer is full,*
2. *no cell is removed prior to transmission unless the buffer is full and it is replaced by a new arrival,*
3. *transmission occurs whenever the buffer is not empty, and*
4. *cells are immediately removed after they are transmitted.*

A WCD maximizes throughput and thus provides the lowest aggregate cell loss ratio for given total buffer size, total link capacity, and input traffic [CR87].

The CSVP buffer allocation scheme and the Work-conserving WRR scheduling scheme are combined to constitute a QD referred to as the *Work-conserving WRR-CSVP* resource allocation mechanism under investigation. This resource allocation scheme satisfies the conditions above and thus belongs to the class of WCD's. A combination of the Work-conserving WRR scheduling scheme and the CS buffer

allocation scheme, the Work-conserving WRR-CS resource allocation scheme, also satisfies the above criteria and thus is a WCD.

2.3 Queueing Model of a Generic Node

2.3.1 Timing Structure

In our queueing model, it is assumed that there is a transmission buffer. Therefore, when a scheduling decision is made, a cell is transferred to this transmission buffer immediately and a cell space is created in the buffer. It is common to have a transmission buffer since the speed of an output link is much slower (it takes one cell time to transmit a cell) than the speed of the operations inside of an ATM switch. In fact, the hardware design proposed in Chapter 6 uses a transmission buffer.

Let us specify the order of operations of the system in a time slot. In the nth time slot, the operations of the system occur as follows. Let $\triangle$ denote the slot time length. Let $0 < \delta < \epsilon < \triangle/2$. The scheduling decision is made at $n\triangle - \epsilon$. If there is a cell in the buffer which satisfies the scheduling conditions, the cell is transferred from the buffer space to the transmission buffer by $n\triangle - \delta$, and thus, a cell space is created in the buffer. Then, between $n\triangle - \delta$ and $n\triangle$, the cells arrive to the system. The admission of these newly arriving cells is examined in a random order between $n\triangle$ and $n\triangle + \delta$. The cell admissions and losses occur during this time. The system is measured at time $n\triangle + \epsilon$. This timing structure is illustrated in Figure 2.4. Note that the parameters, δ and ϵ, are chosen for convenience of

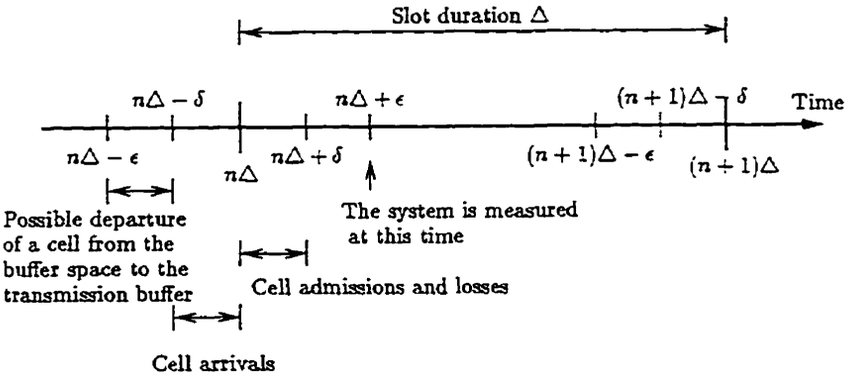

Figure 2.4: Timing structure

specifying the order of operations. Therefore, the symmetric structure around $n\Delta$ does not represent actual operations of a real system. Hereafter, the cell transfer from the buffer space to the transmission buffer is referred to as cell *transmission* since the cell is transmitted to the output link by the end of the time slot.

2.3.2 Performance Measures

Let the initial time be slot time $n = 0$. Let $u_k(n)$ be an indicator function defined by:

$$u_k(n) = \begin{cases} 1, & \text{if a cell of } k\text{th traffic flow is transmitted in the } n\text{th slot,} \\ 0, & \text{otherwise.} \end{cases} \tag{2.2}$$

Let $u(n)$ be an indicator function which indicates whether a cell is transmitted in the nth time slot, i.e.,

$$u(n) = \begin{cases} 1, & \text{if a cell is transmitted in the } n\text{th slot,} \\ 0, & \text{otherwise.} \end{cases} \tag{2.3}$$

Note that $u(n) = \sum_{k=1}^{K} u_k(n)$, i.e., at most one cell can be transmitted in one slot interval. Denote the number of cells belonging to the kth traffic flow that are transmitted during $[0, n]$ by $U_k(n) = \sum_{i=0}^{n} u_k(i)$. Let $U(n)$ be the number of cells transmitted during $[0, n]$, i.e., $U(n) = \sum_{i=0}^{n} u(i)$. Note that $U(n) = \sum_{k=1}^{K} U_k(n)$. Denote the number of cells belonging to the kth traffic flow which are blocked from entry in the nth time slot by $l_k(n)$, and denote the number of cells belonging to the kth traffic flow which are pushed out in the nth time slot by $p_k(n)$. The aggregate number of cells blocked is given by $l(n) = \sum_{k=1}^{K} l_k(n)$ and pushed out by $p(n) = \sum_{k=1}^{K} p_k(n)$. Let $L_k(n)$ denote the number of cells of the kth traffic flow that are blocked during $[0, n]$, i.e., $L_k(n) = \sum_{i=0}^{n} l_k(i)$, and let $L(n)$ denote the number of cells that are blocked from entering the buffer during $[0, n]$, i.e., $L(n) = \sum_{i=0}^{n} l(i)$. Similarly, let $P_k(n)$ be the number of cells of the kth traffic flow that are pushed out during $[0, n]$, i.e., $P_k(n) = \sum_{i=0}^{n} p_k(i)$, and let $P(n)$ denote the number of cells that are pushed out from the buffer during $[0, n]$, i.e., $P(n) = \sum_{i=0}^{n} p(i)$. Note that $L(n) = \sum_{k=1}^{K} L_k(n)$ and $P(n) = \sum_{k=1}^{K} P_k(n)$.

Cell Loss Ratio

As previously mentioned, our main interest is in the cell loss performance. The cell loss ratio is defined as follows:

Definition 3 (Cell Loss Ratio)

The aggregate cell loss ratio, r, is defined by:

$$r = \lim_{n\to\infty} \frac{L(n) + P(n)}{L(n) + P(n) + U(n)}. \tag{2.4}$$

The cell loss ratio of the k^{th} traffic flow, r_k, $k = 1, 2, \cdots, K$ is defined by:

$$r_k = \lim_{n\to\infty} \frac{L_k(n) + P_k(n)}{L_k(n) + P_k(n) + U_k(n)}. \tag{2.5}$$

We assume stationarity, i.e., that the cell loss ratios, r and r_k, exist. Therefore, with sufficiently long observation periods, $[0, N]$, the aggregate cell loss ratio and the cell loss ratios of the kth traffic flow are approximated by:

$$r \approx \frac{L(N) + P(N)}{L(N) + P(N) + U(N)} \tag{2.6}$$

and

$$r_k \approx \frac{L_k(N) + P_k(N)}{L_k(N) + P_k(N) + U_k(N)} \tag{2.7}$$

respectively.

Cell Delay and Cell Delay Variation

The cell delay and cell delay variation performances are also examined. First, we define cell delay at a generic node as follows:

Definition 4 (Cell Delay)

Cell delay at a generic node is defined only for the cells which are transmitted. Let n_a be the slot time that a cell is admitted to the buffer and let n_d be the slot time that the cell transmission completes and the cell leaves the system. The cell delay, d, is defined by:

$$d = n_d - n_a. \tag{2.8}$$

Note that it takes one slot time to transmit a cell in the transmission buffer to the output link. Therefore,

$$d \geq 1. \tag{2.9}$$

Let $d_{k,i}$ be the queueing delay of the ith departure among the $U_k(n)$ departures belonging to the kth traffic flow during $[0, n]$. The mean cell delay of the aggregate traffic and the kth traffic flow are defined as follows:

Definition 5 (Mean Cell Delay)

The mean cell delay of the aggregate process, $\bar{d}$, is given by:

$$\bar{d} = \lim_{n\to\infty} \frac{\sum_{k=1}^{K} \sum_{i=1}^{U_k(n)} d_{k,i}}{U(n)}. \tag{2.10}$$

The mean cell delay of the k^{th} traffic flow, $\bar{d}_k$, $k = 1, 2, \cdots, K$, is given by:

$$\bar{d}_k = \lim_{n\to\infty} \frac{\sum_{i=1}^{U_k(n)} d_{k,i}}{U_k(n)}. \tag{2.11}$$

Cell delay variation is a measure of the dispersion about the mean cell delay. The standard deviation is used to measure cell delay variation.

Definition 6 (Cell Delay Variation)

The standard deviation of cell delay of the aggregate process, σ^d, is given by:

$$\sigma^d = \lim_{n\to\infty} \sqrt{\frac{\sum_{k=1}^{K}\sum_{i=1}^{U_k(n)} (d_{k,i} - \bar{d})^2}{U(n)}}. \tag{2.12}$$

The standard deviation of cell delay of the kth traffic flow, σ_k^d, $k = 1, 2, \cdots, K$, is given by:

$$\sigma_k^d = \lim_{n\to\infty} \sqrt{\frac{\sum_{i=1}^{U_k(n)} (d_{k,i} - \bar{d}_k)^2}{U_k(n)}}. \tag{2.13}$$

Under a stationarity assumption, if the observation interval, $[0, N]$, is sufficiently long, the approximations to these performance measures are given by:

$$\bar{d} \approx \frac{\sum_{k=1}^{K}\sum_{i=1}^{U_k(N)} d_{k,i}}{U(N)}, \tag{2.14}$$

$$\bar{d}_k \approx \frac{\sum_{i=1}^{U_k(N)} d_{k,i}}{U_k(N)}, \tag{2.15}$$

$$\sigma^d \approx \sqrt{\frac{\sum_{k=1}^{K}\sum_{i=1}^{U(N)} (d_{k,i} - \bar{d})^2}{U(N)}}, \tag{2.16}$$

and

$$\sigma_k^d \approx \sqrt{\frac{\sum_{i=1}^{U_k(N)} (d_{k,i} - \bar{d}_k)^2}{U_k(N)}}. \tag{2.17}$$

Wasted Slot Rate

The WRR scheduling scheme may waste time slots. A wasted time slot is defined as follows:

Definition 7 (Wasted Time Slot)

The nth time slot is said to be wasted if and only if

$$u(n) = 0 \tag{2.18}$$

and

$$x(n-1) \neq 0, \tag{2.19}$$

where $x(n-1)$ is the total buffer occupancy in the $(n-1)$th time slot, i.e., at the beginning of the nth time slot, $n\triangle - \epsilon$.

In order to assess the amount of waste, a measure of waste is introduced. Let $W(n)$ be the number of time slots that have been wasted during $[0, n]$ and let $W_k(n)$ be the number of wasted time slots belonging to the kth traffic flow during $[0, n]$. Let $\nu_k(n)$ be the number of time slots that the kth traffic flow received for cell transmission during $[0, n]$. Wasted slot rate is defined as follows:

Definition 8 (Wasted Slot Rate)

The wasted slot rate, w, of the aggregate traffic is defined by:

$$w = \lim_{n\to\infty} \frac{W(n)}{n}. \tag{2.20}$$

The wasted slot rate, $w_k(n)$, of the kth traffic flow, $k = 1, 2, \cdots, K$, is defined by:

$$w_k = \lim_{n\to\infty} \frac{W_k(n)}{\nu_k(n)}. \tag{2.21}$$

Again, we assume the stationarity, i.e., the existence of the wasted slot rates. Thus, if the observation interval, $[0, N]$, is sufficiently long, the wasted slot rates are approximated by:

$$w \approx \frac{W(N)}{N} \tag{2.22}$$

and

$$w_k \approx \frac{W_k(N)}{\nu_k(N)}. \tag{2.23}$$

Note that wasted slot rate is defined only for the systems with WRR scheduling scheme since the Work-conserving WRR scheduling scheme does not waste time slots.

2.3.3 Queueing Models

The WRR-CSVP System

Let us formulate a queueing model of the WRR-CSVP resource allocation mechanism. Recall that B denotes the buffer size and K denotes the number of traffic flows supported. Let $x_k(n)$, $k = 1, 2, \cdots\cdots, K$, be the number of cells belonging to the kth traffic flow in the buffer in the nth time slot. The total buffer occupancy in this time slot, $x(n)$, is given by $x(n) = \sum_{k=1}^{K} x_k(n)$. Let x_k^- be the number of cells belonging to the kth traffic flow in the buffer initially, i.e., at $0 - \epsilon$. The total buffer occupancy in the buffer initially, x^-, is given by $x^- = \sum_{k=1}^{K} x_k^-$. Let $a_k(n)$, $k = 1, 2, \cdots\cdots, K$, be the number of arrivals belonging to the kth traffic flow in the nth time slot. The aggregate number of arrivals in this time slot, $a(n)$, is given by $a(n) = \sum_{k=1}^{K} a_k(n)$. The total buffer occupancy is described by

$$x(0) = \min\{B, a(0) + x^-\}, \tag{2.24}$$

$$x(n) = \min\{B, a(n) + x(n-1) - u(n)\}, \quad n = 1, 2, \cdots. \tag{2.25}$$

The total number of cells discarded, i.e., blocked or pushed out, in a time slot is given by:

$$l(0) + p(0) = [a(0) + x^- - B]^+, \tag{2.26}$$

$$l(n) + p(n) = [a(n) + x(n-1) - u(n) - B]^+, \quad n = 1, 2, \cdots. \tag{2.27}$$

where $[y]^+ = \max[0, y]$. Now, let us describe the behavior of the kth traffic flow. The buffer occupancy of the kth traffic flow is described by:

$$x_k(0) = a_k(0) + x^- - l_k(0) - p_k(0), \tag{2.28}$$

$$x_k(n) = a_k(n) + x_k(n-1) - u_k(n) - l_k(n) - p_k(n), \quad n = 1, 2, \cdots. \tag{2.29}$$

To further assess the behavior of the queueing system based on the WRR-CSVP resource allocation mechanism, it is necessary to determine the values of $l_k(n)$, $p_k(n)$, and $u_k(n)$. The values of $l_k(n)$ and $p_k(n)$ are determined by the cell admission according to the CSVP buffer allocation scheme. The cell admission decision has to consider the following factors: the arrivals in the time slot, $a_k(n)$; and the buffer occupancies after cell transmission, $(x_k(n-1) - u_k(n))$. The constants for the decision include the total buffer size B and the virtual buffer allocations, B_k. Let us denote the set of $p_k(n)$ and the set of $l_k(n)$ by the vectors $\mathbf{l}(n) = (l_1(n), l_2(n), \cdots, l_K(n))$ and $\mathbf{p}(n) = (p_1(n), p_2(n), \cdots, p_K(n))$. Similarly, we denote the number of arrivals, the buffer occupancies, and the indicator functions by $\mathbf{a}(n) = (a_1(n), a_2(n), \cdots, a_K(n))$, $\mathbf{x}(n) = (x_1(n), x_2(n), \cdots, x_K(n))$, and $\mathbf{u}(n) = (u_1(n), u_2(n), \cdots, u_K(n))$. The cell admission decision by the CSVP buffer

allocation scheme can be expressed in a functional form as,

$$(\mathbf{l}(n), \mathbf{p}(n)) = \Phi(\mathbf{a}(n), \mathbf{x}(n-1) - \mathbf{u}(n)), \tag{2.30}$$

where Φ is a functional operator or mapping.

The analytical approach in [WM95] was limited to $K = 2$ since the dimension of the system increases for $K > 2$ and it becomes difficult to apply the same approach. In addition, for $K > 2$, the pushout operator Φ is no longer trivial. As already pointed out, the pushout traffic flow selection in [WM95] is valid only for $K = 2$. When more than two traffic flows are supported, i.e., $K > 2$, there may be two or more traffic flows over-subscribing the buffer space and thus it is necessary to select one. Therefore, the argument in [WM95] is no longer directly applicable. There can be a selection policy such as pushing out a traffic flow which is over-subscribing the buffer space most; other policies are further discussed in Section 3.3.2. Furthermore, a set of input values to Φ may not determine a unique set of output values; several sets of potential output may result with certain probabilities. These factors make the system analytically intractable.

In the WRR scheduling scheme, each time slot is assigned to one of the traffic flows. We denote the set of time slots assigned to the kth traffic flow by Θ_k, $k = 1, 2, \cdots, K$. The indicator function, $u_k(n)$, for the WRR scheduling scheme is thus further specified by:

$$u_k(n) = \begin{cases} 1, & \text{if } n \in \Theta_k \text{ and } x_k(n-1) > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{2.31}$$

The cell loss relationship of the CSVP and the CP buffer allocation schemes is studied in the following. It is shown that the cell loss ratio achieved by the CSVP scheme is less than or equal to the cell loss ratio achieved by the CP scheme for any input traffic. A similar relationship was numerically observed in [WM95].

The WRR-CSVP system for the kth traffic flow is a queueing system with a B_k partitioned (guaranteed) buffer space and an extra $B - B_k$ shared buffer space which may potentially be available. In the WRR-CP system, a B_k buffer space is strictly allocated to the kth traffic flow and there is no extra space available. The WRR scheduling scheme provides exactly the same time slot allocation to both systems. Therefore, the cell loss ratio of the kth traffic flow in the WRR-CSVP system should be less than or equal to that in the WRR-CP system. Thus we have the following:

Theorem 1 *Let $\tilde{x}_k(n)$ denote the buffer occupancies of the kth traffic flow in the n^{th} time slot in the WRR-CP system. Assume that the buffer occupancies of both the WRR-CSVP system and the WRR-CP system are equal in the initial time slot, i.e., $x_k(0) = \tilde{x}_k(0)$. Given identical input traffic, if the same amount of buffer space and service capacity are allocated to the k^{th} traffic flow in both systems, the cell loss ratio, r_k's, $k = 1, 2, \cdots, K$, of the k^{th} traffic flow in the WRR-CSVP system and the cell loss ratios, $\tilde{r}_k$'s, $k = 1, 2, \cdots, K$, of the k^{th} traffic flow in the WRR-CP system satisfy*

$$r_k \leq \tilde{r}_k, \tag{2.32}$$

for $k = 1, 2, \cdots, K$.

Proof: Let $\tilde{U}_k(n)$ be the number of cells belonging to the kth traffic flow that are transmitted during $[0, n]$ in the WRR-CP system. And let $\tilde{L}_k(n)$ be the number of blocked cells belonging to the kth traffic flow during $[0, n]$. Note that no cell is pushed out in the WRR-CP system.

First, we show $x_k(n) \geq \tilde{x}_k(n)$, $n = 1, 2, \cdots$. Since the WRR scheduling scheme is a fixed time slot allocation scheme, the patterns of the time slot allocation in the both systems are exactly the same. Recall that the buffer occupancy at the beginning of the time slot, based on which the service scheduling decision is made, is equal to the buffer occupancy of the previous time slot, i.e.,

$$x_k(n\triangle - \epsilon) = x_k(n-1). \tag{2.33}$$

Therefore, the value of the indicator function is given by

$$\begin{cases} u_k(n) = 1 \quad \text{and} \quad \tilde{u}_k(n) = 0, & \text{if } n \in \Theta_k, \text{ and } x_k(n-1) > 0 \text{ and } \tilde{x}_k(n-1) = 0. \\ u_k(n) = 0 \quad \text{and} \quad \tilde{u}_k(n) = 1, & \text{if } n \in \Theta_k, \text{ and } x_k(n-1) = 0 \text{ and } \tilde{x}_k(n-1) > 0. \\ u_k(n) = \tilde{u}_k(n) = 1, & \text{if } n \in \Theta_k, \text{ and } x_k(n-1) > 0 \text{ and } \tilde{x}_k(n-1) > 0. \\ u_k(n) = \tilde{u}_k(n) = 0, & \text{if } n \notin \Theta_k, \text{ or} \\ & \text{if } n \in \Theta_k, \text{ and } x_k(n-1) = \tilde{x}_k(n-1) = 0. \end{cases} \tag{2.34}$$

Therefore, if

$$x_k(n-1) \geq \tilde{x}_k(n-1), \tag{2.35}$$

then

$$x_k(n\triangle - \delta) \geq \tilde{x}_k(n\triangle - \delta). \tag{2.36}$$

In the WRR-CS system, the buffer space, B_k, is guaranteed, whereas in the WRR-CSVP system, not only the buffer space, B_k, is guaranteed but also the kth traffic may utilize some extra buffer spaces. Therefore, it is easily derived that if

$$x_k(n\triangle - \delta) \geq \tilde{x}_k(n\triangle - \delta), \tag{2.37}$$

then

$$x_k(n\triangle + \epsilon) \geq \tilde{x}_k(n\triangle + \epsilon), \tag{2.38}$$

for the same arrival pattern to the two systems.

The system is measured at $n\triangle + \epsilon$, i.e.,

$$x_k(n) := x_k(n\triangle + \epsilon) \tag{2.39}$$

and

$$\tilde{x}_k(n) := \tilde{x}_k(n\triangle + \epsilon) \tag{2.40}$$

Therefore, from Inequalities (2.35) to (2.38), it is clear that if

$$x_k(n-1) \geq \tilde{x}_k(n-1), \tag{2.41}$$

then

$$x_k(n) \geq \tilde{x}_k(n), \tag{2.42}$$

for $n = 1, 2, \cdots$. Since $x_k(0) = \tilde{x}_k(0)$, by induction we obtain

$$x_k(n) \geq \tilde{x}_k(n), \quad n = 1, 2, \cdots . \tag{2.43}$$

By applying (2.43) to (2.34), we have

$$U_k(n) \geq \tilde{U}_k(n), \quad n = 0, 1, \cdots . \tag{2.44}$$

The number of arrivals belonging to the kth traffic flow during $[0, n]$, $A_k(n)$, is equal in both systems, i.e.,

$$A_k(n) = L_k(n) + P_k(n) + U_k(n) + x_k(n) = \tilde{L}_k(n) + \tilde{U}_k(n) + \tilde{x}_k(n), \quad n = 0, 1, \cdots . \tag{2.45}$$

Thus, from Inequalities (2.43) and (2.44),

$$L_k(n) + P_k(n) \leq \tilde{L}_k(n), \quad n = 0, 1, \cdots . \tag{2.46}$$

The number of arrivals in the two systems being equal also gives

$$\lim_{n\to\infty}[\{(L_k(n)+P_k(n)+U_k(n)\}-\{\tilde{L}_k(n)+\tilde{U}_k(n)\}]=0. \tag{2.47}$$

Hence,

$$r_k=\lim_{n\to\infty}\frac{L_k(n)+P_k(n)}{L_k(n)+P_k(n)+U_k(n)}\le\lim_{n\to\infty}\frac{\tilde{L}_k(n)}{\tilde{L}_k(n)+\tilde{U}_k(n)}=\bar{r}_k. \tag{2.48}$$

■

The Work-conserving WRR-CSVP System

The time slot reassignment operation is contained in a condition of the indicator function $u_k(n)$. Let the nth time slot be assigned to the kth traffic flow by the WRR scheduling scheme, i.e., $n\in\Theta_k$. The time slot reassignment operation reassigns the time slot to the other traffic flow, if there is no cell belonging to the kth traffic flow in the buffer. Let k_R be the traffic flow to which the time slot is reassigned. The determination process has to consider the buffer occupancies at the beginning of the nth time slot, $x_k(n-1)$'s. Thus, this determination process of time slot reassignment is expressed by the operator, ψ, such as:

$$k_R=\psi(\mathbf{x}(n-1)), \tag{2.49}$$

which appears in a condition of $u_k(n)$. The time slot reassignment operator, ψ, may also consider the virtual buffer space allocation, $(B_1,B_2,\cdots,B_K)$.

The analytical approach in [WM95] was limited to $K = 2$ since the dimension of the system increases for $K > 2$ and it becomes difficult to apply the same approach. Furthermore, the analytical complexity increases for $K > 2$ since the pushout operator, Φ, is no longer trivial. In addition, the time slot reassignment operator ψ is no longer trivial for $K > 2$. As already pointed out, when more than two traffic flows are supported, i.e., $K > 2$, there may be two or more traffic flows occupying the buffer space and thus it is necessary to select one. There can be a policy to choose a traffic flow such as reassigning a time slot to a traffic flow which is occupying the buffer space most; other policies are further discussed in Section 3.4.3. Furthermore, a set of input values to ψ may not determine a unique set of output values; several sets of potential output may result with certain probabilities. These factors contribute to make the system analytically intractable.

Using the time slot reassignment operator, the indicator function of the Work-conserving WRR scheduling scheme is further specified by:

$$u_k(n) = \begin{cases} 1, & \text{if } n \in \Theta_k \text{ and } x_k(n-1) > 0, \text{ or} \\ & \quad n \notin \Theta_k \text{ and } k = \psi(\mathbf{x}(n-1)), \\ 0, & \text{otherwise.} \end{cases} \tag{2.50}$$

This indicator function is applied in Equation (2.29) to describe the queueing behavior. It should be noted that the cell admission operator, Φ, defined in Equation (2.30) includes the indicator function, $u_k(n)$, in its arguments because in a time slot, cell admission is determined based on the state of buffer space after cell transmission.

For the aggregate process, the indicator function, $u(n)$, is simplified to

$$u(n) = \begin{cases} 1, & \text{if } x(n-1) > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{2.51}$$

This equation is equivalent to Equation (9) in [CR87]. Using this formula, the number of cells in the system is expressed as:

$$x(0) = \min\{B, a(0) + x^-\}, \tag{2.52}$$

$$x(n) = \min\{B, a(n) + [x(n-1) - 1]^+\}, \quad n = 1, 2, \cdots. \tag{2.53}$$

and the number of cells blocked or pushed out from the aggregate process is described by:

$$l(0) + p(0) = [a(0) + x^- - B]^+, \tag{2.54}$$

$$l(n) + p(n) = [a(n) + [x(n-1) - 1]^+ - B]^+, \quad n = 1, 2, \cdots. \tag{2.55}$$

These equations are equivalent to Equations (4) to (7) in [CR87]. Thus the Work-conserving WRR-CSVP system is a WCD, and the following conservation relation, shown in [CR87], holds:

Proposition 1 (Conservation Law [CR87]) *Let $\overline{\lambda}$ be the aggregate load and $\overline{\lambda}_k$, $k = 1, 2, \cdots, K$, be the average load of the kth traffic flow. Denote $\boldsymbol{\lambda} = (\overline{\lambda}_1, \overline{\lambda}_2, \cdots, \overline{\lambda}_K)$ and $\mathbf{r} = (r_1, r_2, \cdots, r_K)$. For fixed arrival statistics and buffer capacity, the aggregate cell loss ratio, r is invariant for all WCD's, and the per-*

flow cell loss ratios satisfy

$$r \cdot \lambda / \overline{\lambda} = r, \tag{2.56}$$

where "$\cdot$" is the Euclidean inner product.

This means that the aggregate cell loss ratio is constant regardless of the position of the virtual partition for a given total buffer size and input traffic. It should also be noted that the Work-conserving WRR-CS system exhibits the same aggregate cell loss ratio as the Work-conserving WRR-CSVP system, since it is also a WCD.

It was reported in [WM95] that the cell loss performance is sensitive to virtual partitioning of the CSVP scheme: it was numerically observed that if the virtually allocated buffer space becomes larger, the resultant cell loss ratio becomes smaller. This confirms our intuition that if a larger buffer space is guaranteed to a traffic flow, the cell loss performance of the traffic flow should improve. Although the system becomes analytically intractable for $K > 2$, a non-increasing relation between cell loss ratio and virtually allocated buffer space can be shown for the case of two traffic flows, i.e., $K = 2$.

Theorem 2 *Given identical input traffic, consider two buffer space allocations by the Work-conserving WRR-CSVP system, (B_1, B_2) and $(\hat{B}_1, \hat{B}_2)$, where $B_1 + B_2 = \hat{B}_1 + \hat{B}_2 = B$. If*

$$\hat{B}_k > B_k \tag{2.57}$$

and the initial condition is given by

$$x_k(0) = \hat{x}_k(0) = 0, \tag{2.58}$$

then

$$\hat{r}_k \leq r_k, \tag{2.59}$$

for $k = 1, 2$, where r_k and $\hat{r}_k$ are the cell loss ratios of the kth traffic flow in the (B_1, B_2) buffer and the $(\hat{B}_1, \hat{B}_2)$ buffer, respectively.

The proof of Theorem 2 is provided in Appendix A.

2.4 Concluding Remarks

The queueing model discussed in this chapter may be analytically intractable, and thus a closed form solution may not be available, although numerical calculation by iteration may be possible. In this work, however, an emulation approach was adopted in order to obtain insights for implementation. An emulator was constructed in the following manner: the Work-conserving WRR-CSVP resource allocation scheme was implemented as an application program on a workstation; then, peripheral software to numerically examine its performance was built around it. It should be noted that the performance of the system can be obtained numerically by iterations applied to the queueing model in this chapter since it is a discrete-time system. However, it may be as time consuming as to obtain results by emulation.

Chapter 3 provides the detailed algorithms of the Work-conserving WRR-CSVP resource allocation mechanism, including traffic flow selection for the pushout and the time slot reassignment operations. The design of an emulator is also discussed in Chapter 3.

Chapter 3

Design Of Algorithms And Emulator

3.1 Introduction

As already mentioned, an emulation approach is taken in this work. The Work-conserving WRR-CSVP resource allocation mechanism is implemented as an application program on a workstation. This software is designed so that it can easily be transformed to firmware. Then, the peripheral software, such as the cell generator representing traffic sources, the free cell pool, and some additional parts to collect statistics, is developed to constitute an emulator. The term "emulation" is used to emphasize that the Work-conserving WRR-CSVP resource allocation mechanism is implemented as a finite state machine. The software implementation of the Work-conserving WRR-CSVP resource allocation scheme provides us with some insights for implementation. In fact, the development of an emulator leads

us to the hardware design specified in Appendix C. The emulation approach also allows us to investigate various performance aspects, such as cell loss, cell delay, and cell delay variation, for more than two traffic flows.

In order to implement the Work-conserving WRR-CSVP resource allocation scheme, the operations of the CSVP buffer allocation scheme provided in [WM95] need to be refined into the detailed algorithms. Also, the operations of the Work-conserving WRR scheduling scheme need to be refined. Furthermore, it is pointed out in Chapter 2 that a mechanism is needed to handle more than two traffic flows for both the pushout and the time slot reassignment operations. It is necessary to provide a clear detailed algorithmic description of the Work-conserving WRR-CSVP resource allocation mechanism which is capable of handling more than two traffic flows.

It is also of interest to know how the buffer space of the Work-conserving WRR-CSVP resource allocation should be implemented. In order to enable the admission, pushout, and service of cells, the buffer space must be structured suitably for these operations.

In this chapter, the design of the algorithms of the Work-conserving WRR-CSVP resource allocation mechanism is discussed. The resource allocation mechanism is divided into two modules: the Cell Admission Controller module and the Service Scheduler module. The Cell Admission Controller module performs cell admission according to the CSVP buffer allocation scheme. The Service Scheduler module performs cell transmission according to the Work-conserving WRR scheduling scheme. The design in this chapter also describes the Buffer Space, which is a data structure

designed for the operations of the Work-conserving WRR-CSVP resource allocation mechanism.

A brief description of the design of an emulator is also provided in this chapter. Based on the design, an emulator is developed and the algorithms of the Work-conserving WRR-CSVP resource allocation mechanism are validated.

Hereafter, the term *Traffic Class* is used to represent the unit of traffic to which our resource allocation mechanism is applied. A Traffic Class may constructed by a superposition of traffic flows from mini-sources.

3.2 Buffer Space

The operations of the CSVP buffer allocation scheme, the admission, blocking, and pushout of the cells, are performed on a Traffic Class basis. Since it is necessary to preserve the order of arrivals for service, a queue of a Traffic Class, a VC queue, needs to be a FIFO queue. It is standard in software design to implement a FIFO queue by a logical linked list. In addition, the elasticity of a chain by a logical linked list is most suitable for the CSVP buffer allocation scheme, where the length of the chain ranges from 0 to the size of the total buffer space.

What is the suitable mechanism to select a cell to be pushed out then? The cells in the over-subscribed portion of the buffer are the cells which are admitted by taking advantages of the complete sharing of the buffer space, i.e., they are the cells which could have been discarded had the buffer been full, or had the buffer space been completely partitioned. Therefore, it is deemed appropriate to push out a cell in the over-subscribed portion of the buffer.

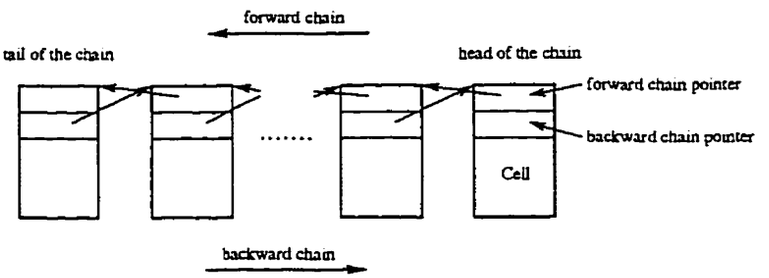

Figure 3.1: Doubly linked list

The last cell of the queue of the over-subscribing Traffic Class is in the over-scribed portion of the buffer. And the operation to deque a cell at the tail of a queue, the LIFO operation, can be easily implemented if the cells are chained by a *doubly linked list* (Figure 3.1). Thus, the Buffer Space is constructed by a set of FIFO/LIFO queues, each of which is assigned to each Traffic Class. Another advantage of using a chain structure is that unused space (free cells) can be easily managed by a queue of empty spaces. It should be noted that there are other implementations to pushout a cell in the over-subscribed portion of the buffer. For example, a single forward chain with an additional tail pointer which indicates the head of the over-subscribed portion can pushout a cell pointed by this second tail pointer [Pre97].

Let K be the number of Traffic Classes supported. The FIFO/LIFO queues of Traffic Classes are maintained by the following parameters: the total buffer size and the buffer thresholds, the amount of virtually allocated space, are the constants, B and B_k, $k = 1, 2, \cdots, K$, respectively; and the total buffer occupancy and the buffer occupancies of the Traffic Classes are stored in the variables, x and x_k,

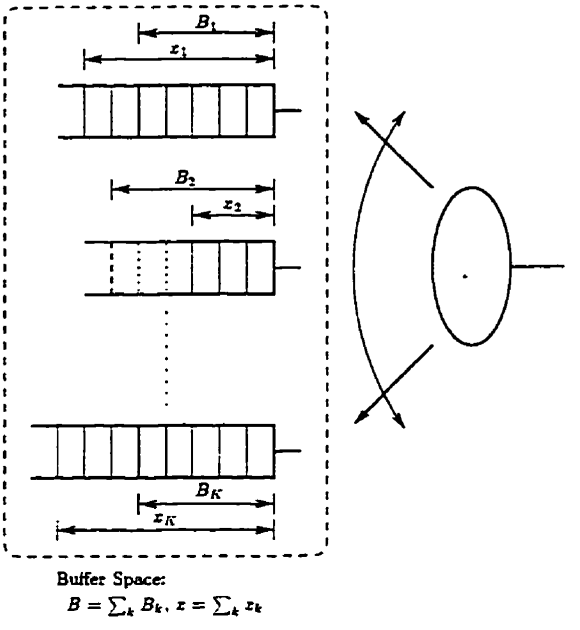

Figure 3.2: The Buffer Space

$k = 1, 2, \cdots, K$, respectively. The Buffer Space is illustrated in Figure 3.2.

3.3 Cell Admission Controller

3.3.1 Basic Operations of the Cell Admission Controller

The Buffer Allocation Controller examines the admission of the arrivals cell by cell in each time slot. It is assumed that the examination of cell admission is applied in a random order to the cells arriving in the same time slot since it is unbiased and thus deemed appropriate. The admission of a cell is examined according to the CSVP scheme and the procedure is repeated until all arrivals in the time slot are examined. Recall that k_P denotes the Traffic Class to be pushed out. The

admission rule by the CSVP buffer allocation scheme is further specified in the following:

Cell Admission by the CSVP Scheme:

1. Identify Traffic Class k to which the cell belongs.
2. If $x < B$,
   - put the cell into the FIFO/LIFO queue belonging to Traffic Class k.
   - Set $x = x + 1$ and $x_k = x_k + 1$.
3. If $x = B$,
   (a) if $x_k \geq B_k$,
   - drop the cell.

   (b) if $x_k < B_k$,
   - find a Traffic Class which is over subscribing the buffer space, i.e., find k_P such that $x_{k_P} > B_{k_P}$.
   - Pushout a cell belonging to Traffic Class k_P using the LIFO operation.
   - Set $x_{k_P} = x_{k_P} - 1$.
   - Admit the arriving cell into the queue of Traffic Class k traffic using the FIFO operation.
   - Set $x_k = x_k + 1$.

By applying this cell admission procedure repeatedly to all the arrivals in the time slot in a random order, the numbers of cells admitted, dropped, and pushed

out become clear, i.e., the output of the cell admission operator, Φ, for the time slot is obtained.

3.3.2 Pushout Class Selection Methods

As already pointed out, when there are more than two Traffic Classes, there may be two or more Traffic Classes over-subscribing the buffer space. Thus, an additional mechanism is necessary to determine the Traffic Class, k_P, to be pushed out. The random selection is defined by:

1. **Random selection:** Evaluate $x_k - B_k$, $k = 1, 2, \cdots, K$. Choose k_P at random among k's such that $x_k - B_k > 0$.

Since this method does not require comparisons, it may be computationally inexpensive, although it requires the use of a pseudo-random number generator. Therefore, this method may be preferred if it performs satisfactorily.

There may be advantages to using a more sophisticated selection method, in spite of the computational cost for comparisons. We consider policies to penalize a greedy Traffic Class. If a Traffic Class is greedy, it is likely that the Traffic Class over-subscribes the buffer space in a large amount. Therefore, the following method can be adopted:

2. **Largest Excess by Absolute Amount selection:** Evaluate $x_k - B_k$, $k = 1, 2, \cdots, K$. Choose k_P such that $x_{k_P} - B_{k_P} = \max_k(x_k - B_k)$.

However, it may be more appropriate to evaluate excess by relative amount to the allocated buffer space, i.e.,

3. **Largest Excess by Relative Amount selection:** Evaluate $x_k - B_k$, $k = 1, 2, \cdots, K$. Choose k_P such that $(x_{k_P} - B_{k_P})/B_{k_P} = \max_k (x_k - B_k)/B_k$.

This method involves floating point calculations and may not be preferable. Note that when the buffer space is evenly allocated, i.e., each Traffic Class receives the same amount of buffer space by the virtual partition, the largest excess by relative amount selection method is equivalent to the largest excess by absolute amount selection. These sophisticated methods require $K - 2$ comparisons.

Another idea is to prioritize the selection as follows:

4. **Priority selection:** Evaluate $x_k - B_k$, $k = 1, 2, \cdots, K$. Choose k_P with highest priority level, π_{k_P}, among Traffic Classes k's such that $x_k - B_k > 0$. The priority level, π_k, $k = 1, 2, \cdots, K$, is preassigned.

This method requires at most $K - 2$ comparisons.

It may be necessary to "break ties" in these sophisticated methods. In such cases, random selection is adopted to resolve the situation.

The cell loss performance of these selection methods needs to be examined to determine which one should be adopted. In Chapter 4, they are compared by emulation.

3.4 Service Scheduler

Let k_S be the Traffic Class to which the current time slot is to be assigned by the scheduling scheme. The main procedure of the Service Scheduler module in a time slot is as follows:

1. Determine Traffic Class k_S to receive the transmission.

2. Select the FIFO/LIFO queue of Traffic Class k_S.

3. Transfer the cell at the head of the queue to the transmission buffer.

4. Set $x_{k_S} = x_{k_S} - 1$; and $x = x - 1$.

3.4.1 Basic Operations of the WRR Service Scheduler

In order to develop the Service Scheduler based on the WRR scheduling scheme, the patterns of the time slot allocation within a service cycle have to be specified. *Exhaustive service* is one of the simplest time slot allocation patterns, where a Traffic Class receives the time slots consecutively until it has received its share before passing control to the next Traffic Class. There is no iterative comparisons. Usually, the WRR scheduling scheme assumes Exhaustive service (See, e.g., [KKK90, RRA95]). Let s be the number of remaining time slots in the current service cycle. Let s_k, $k = 1, 2, \cdots, K$, be the number of time slots yet to be received by Class k traffic during the current cycle. Set the initial value $s = 0$. Let κ be the parameter indicating the Traffic Class whose eligibility is examined. The algorithm to obtain k_S in a time slot is specified as follows:

Exhaustive service:

Step 1: If $s = 0$, set $s = S$ and $s_k = S_k$, $k = 1, 2, \cdots, K$; set $\kappa = 1$; and go to Step 3.

Step 2: If $s_\kappa = 0$, transfer control to the next Traffic Class by $\kappa = \kappa + 1$.

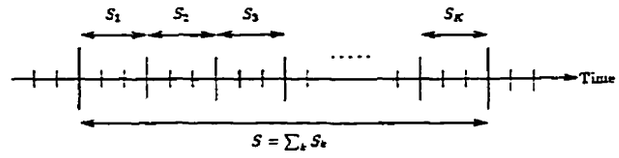

Figure 3.3: Exhaustive service

Step 3: Set $s_\kappa = s_\kappa - 1$ and $s = s - 1$.

Step 4: Set $k_S = \kappa$.

Exhaustive service is illustrated in Figure 3.3.

The granularity of the capacity allocation is limited in the WRR scheduling scheme; in order to have a better granularity, a long service cycle is necessary. However, it is known that a long service cycle can cause a larger cell loss ratio and cell delay by Exhaustive service [RRA95]. This is because WRR scheduling is a fixed time slot allocation scheme and time slots can be wasted. Therefore, it is important to have a mechanism other than Exhaustive service if we are to have a long service cycle and yet avoid cell loss and cell delay performance degradation.

Exhaustive service may also cause undesirable bursts in the output traffic. In Exhaustive service, Traffic Class k receives at most S_k consecutive time slots. The number of cells belonging to Traffic Class k in the buffer and the number of arrivals belonging to Traffic Class k may result in a situation where S_k cells belonging to Traffic Class k are emitted to the output link consecutively. This series of cells can be seen as a burst if S_k is large. And the buffer occupancies and arrival patterns of the other Traffic Classes can create the consecutive emissions of more than S_k cells. The burst may cause poor system performance at the downstream nodes.

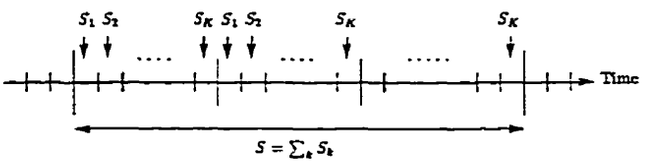

Figure 3.4: Round Robin service

Round Robin service is one of the alternative mechanisms where slots are allocated to Traffic Classes on an alternating basis [KSC91]. This can reduce slot wastage and is expected to perform well, especially when the capacity is fairly evenly allocated. In addition, Round Robin service may prevent undesirable bursts in the output traffic because there will be at most K time slots between two cell emissions belonging to Traffic Class k with Round Robin service, The algorithm of Round Robin service is specified by replacing **Step 2** in Exhaustive service:

Round Robin service:

Step 2: Transfer control to the next Traffic Class by $\kappa = \kappa + 1$. If $\kappa > K$, set $\kappa = 1$. If $s_\kappa = 0$, repeat the above until κ satisfying $s_\kappa > 0$ is found.

Round Robin service is illustrated in Figure 3.4. Note that this **Step 2** requires at most K comparisons to find κ s.t. $s_\kappa > 0$. Therefore, Round Robin service is deemed more complex than Exhaustive service.

3.4.2 Time Slot Reassignment Mechanism

The WRR scheduling scheme can be made more efficient by reassigning potentially wasted time slots to another Traffic Class. The algorithm of the time slot reassignment operation is specified in the following. Recall that k_R denotes the Traffic

Class to which the current time slot is to be reassigned. The time slot reassignment operation replaces **Step 4** of Exhaustive and Round Robin services:

Time Slot Reassignment Operation:

Step 4: If $x_\kappa > 0$, set $k_S = \kappa$; otherwise, examine the buffer occupancy of the other Traffic Classes, x_k, $k = 1, 2, \cdots, \kappa - 1, \kappa + 1, \cdots, K$, and find k_R such that $x_{k_R} > 0$. Set $k_S = k_R$.

This time slot reassignment operation is expected to improve the cell loss and cell delay performances drastically. Also, it is of interest to examine the effect of the long service cycle with Exhaustive service. The time slot reassignment operation may reduce the effect.

3.4.3 Time Slot Reassignment Class Selection Methods

It is pointed out in Chapter 2 that a mechanism must be added to the time slot reassignment operation to select the Traffic Class k_R to receive the reassigned time slot in order to handle more than two Traffic Classes. A number of selection methods can be considered such as Standby Queue in [ZF93] and Carry-Over in [SMT96]. Standby Queue is an additional queue to store all the cells in the buffer in FIFO order. When a time slot is reassigned, the cell at the head of Standby Queue is transmitted. In this scheme, the numbers of queueing and dequeueing operations are doubled, and an entry in Standby Queue has to be removed whenever a cell is transmitted according to the decision by the WRR scheduling scheme. This increases the amount of operations. Carry-Over is used to achieve finer granularity of capacity allocation and requires floating point calculations. In this work, we

consider the following methods to select a Traffic Class, which are computationally less expensive than these two methods.

The random selection is defined as follows:

1. **Random selection:** Evaluate x_k, $k = 1, 2, \cdots\cdots, K$. Choose k_R at random among k's such that $x_k > 0$.

Since this method does not require comparisons, it may be computationally inexpensive, although it requires the use of a pseudo-random number generator. Therefore, this method may be preferred if it performs satisfactorily.

There may be advantages to using a more sophisticated policy, in spite of the computational cost for comparisons. We consider policies to favor Traffic Classes in burst. When a Traffic Class is in burst, it is likely that it occupies the buffer space in a large amount. Therefore, the methods in the following can be adopted:

2. **Largest Occupancy by Absolute Amount selection:** Evaluate x_k, $k = 1, 2, \cdots, K$. Choose k_R such that $x_{k_R} = \max_k x_k$.

3. **Largest Occupancy by Relative Amount selection:** Evaluate x_k, $k = 1, 2, \cdots, K$. Choose k_R such that $x_{k_R}/B_{k_R} = \max_k x_k/B_k$.

The largest occupancy by relative amount selection method involves floating point calculations. The buffer space allocated by the virtual partition can be taken into consideration as follows:

4. **Largest Excess by Absolute Amount selection:** Evaluate $x_k - B_k$, $k = 1, 2, \cdots, K$. Choose k_R such that $x_{k_R} - B_{k_R} = \max_k(x_k - B_k)$ and $x_{k_R} - B_{k_R} >$

0. If there is no Traffic Class satisfying this condition, adopt the largest occupancy by absolute amount selection.

Note that a similar method where the excess is measured by relative amount to the buffer space virtually allocated is equivalent to the largest occupancy by relative amount method and thus excluded. Also note that when the buffer space is evenly allocated, i.e., each Traffic Class receives the same amount of buffer space by the virtual partition, the three methods above are equivalent. These sophisticated methods require $K-2$ comparisons.

Another idea is to prioritize the selection:

5. **Priority selection:** Evaluate x_k, $k = 1, 2, \cdots, K$. Choose k_R with highest priority level, ϖ_{k_R}, among k's such that $x_k > 0$. The priority level, ϖ_k, $k = 1, 2, \cdots, K$, is preassigned independently to the priority level of pushout.

This method requires at most $K-2$ comparisons.

It may be necessary to "break ties" in these sophisticated methods. In such cases, random selection is adopted to resolve the situation.

The cell loss performance of these selection mechanisms needs to be examined to determine which one should be adopted. In Chapter 4, they are compared by emulation.

It should be noted that the WRR scheduling scheme in [KSC91] is work-conserving using a different mechanism. A service cycle is divided into several subcycles in which different Traffic Classes are assigned to be eligible, which is indicated by the eligibility bits, each of which corresponds to a subcycle, in the table of VC information. This architecture allows flexibility in setting the pattern

of visit within a service cycle, including Exhaustive and Round Robin services. At the beginning of each subcycle, the eligible Traffic Classes indicated by the eligibility bits are examined for *readiness*, i.e., if they have a cell to be transmitted in the buffer and the readiness is indicated by the ready bit in the table of VC information. Only the ready Traffic Classes receive transmission in the subcycle then the next subcycle is initiated. Thus, the actual length of subcycles and service cycles varies depending on the readiness of the Traffic Classes. It should be noted that the queueing model defined in Chapter 2 is no longer applicable since the indication function, $u_k(n)$, is different.

3.5 Emulation System

3.5.1 Structure of the Emulator

The Cell Admission Controller and the Service Scheduler described in the previous sections and the Cell Departure module, which completes cell transmission, together with the data structures such as the Buffer Space and the Transmission Buffer, constitute the software implementation of the Work-conserving WRR-CSVP resource allocation mechanism, and are the core parts of the emulator. Three other modules are added to construct an emulator, which provide us with means to evaluate system performance such as cell loss, cell delay and cell delay variation. The emulation system consists of the following six modules:

1. The **Free Queue** is a queue of free cells. A cell is a data structure representing an ATM cell. The cells are taken from the Free Queue upon generation by

the cell generator. The cells are returned to the Free Queue upon discard, pushout, and departure.

2. The **Cell Generator** represents the traffic sources and generates cells in each time slot.

3. The **Random Shuffler** prepares arriving cells in each time slot in a random order so that cell admission is performed in a random order.

4. The **Cell Admission Controller** performs the CSVP buffer allocation scheme.

5. The **Service Scheduler** selects a cell to be transmitted, transfers it to the Transmission Buffer, and starts the transmission in each time slot.

6. The **Cell Departure** module completes cell transmission.

These modules are connected as follows:

- The Cell Generator module is connected to the Free Queue module to generate cells and to the Random Shuffler module by a FIFO queue.
- The Random Shuffler module is connected to the Cell Admission Controller module by a FIFO queue.
- The Cell Admission Controller module is connected to the Service Scheduler module by the Buffer Space, a set of FIFO/LIFO queues, and to the Free Queue module for dropping and pushing out cells.
- The Service Scheduler module is connected to the Buffer Space and to the Transmission Buffer.

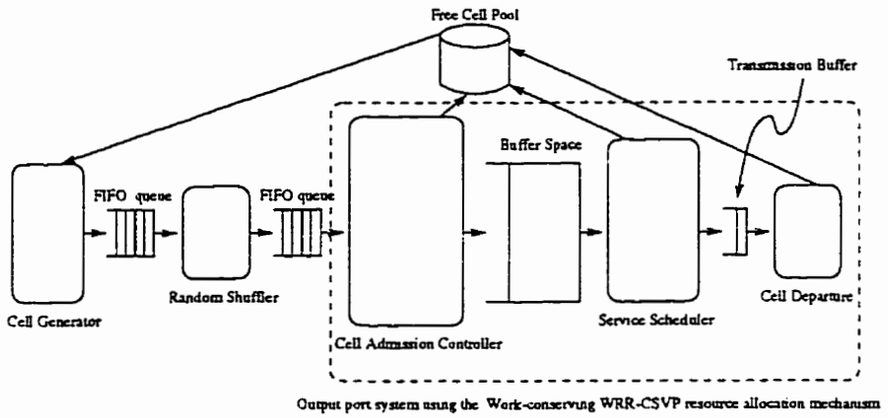

Figure 3.5: Structure of the emulator

- The Cell Departure module is connected to the Transmission Buffer and to the Free Queue.

The structure of the emulator is illustrated in Figure 3.5.

3.5.2 Basic Operations

Recall the timing structure of a time slot in Section 2.1.3. According to this timing structure, the operation of the emulation system in a time slot is specified as follows.

Operation of Emulator in a Time Slot:

Step 1: Obtain Traffic Class k_S according to the Work-conserving WRR scheduling scheme. If $x_{k_S} > 0$, transfer the cell belonging to traffic class k_S in the Buffer Space to the Transmission Buffer and set $x = x - 1$. (Service Scheduler)

Step 2: The cells are generated by the cell generator by obtaining free cells from the Free Queue. The cells are put into the FIFO queue connected to the

random shuffler. (Cell Generator)

Step 3: The Random Shuffler obtains cells out of the input FIFO queue connected to the Cell Generator and feeds them in a random order into the output FIFO queue connected to the Cell Admission Controller. (Random Shuffler)

Step 4: The Cell Admission Controller takes the cells from the input FIFO queue and processes them according to the CSVP buffer allocation scheme. Admitted cells are queued in the Buffer Space and blocked and pushed-out cells are returned to the Free Queue. (Cell Admission Controller)

Step 5: The cell in the Transmission Buffer leaves the system. The cell is returned to the Free Queue. (Cell Departure)

The Cell Admission Controller, the Service Scheduler, and the Cell Departure models, which performs the Work-conserving WRR-CSVP resource allocation mechanism, are validated by the test programs with a thorough set of test patterns. The Cell Generator and the Random Shuffler contain probabilistic operations using a pseudo-random number generator. These parts are independently validated by collecting statistical measurements such as first and second moments and comparing the values with those derived using analytical models.

Chapter 4

Performance Of A Generic Node

4.1 Introduction

The performance of the Work-conserving WRR-CSVP resource allocation mechanism is evaluated in this chapter using a single-server queueing model introduced in Chapter 2. It is of interest to observe the efficient use of resources by our resource allocation mechanism to achieve better cell loss performance, and to demonstrate the robustness of resource allocation of the mechanism to protect well-behaving traffic flows.

First, the advantages of the Work-conserving WRR scheduling scheme are examined. It is shown that the addition of the time slot reassignment operation considerably improves system performance by the efficient use of link capacity. Then, the CSVP buffer allocation scheme is examined from the view point of its efficient use of buffer space to obtain a better aggregate cell loss ratio, and its robustness of buffer allocation to provide protection, especially for non-bursty traffic, to achieve

lower cell loss for well-behaved traffic. The effect of virtual partitioning is also investigated.

Another important topic of investigation is the design issues raised in Chapter 3. The three issues raised are: the mechanics of service to realize the WRR scheduling scheme, Exhaustive and Round Robin services; the traffic class selection method for the pushout operation; and the traffic class selection method for the time slot reassignment operation. The proposed alternatives in Chapter 3 for these issues are examined and design guidelines are provided.

The main results of this chapter were also presented in [YM96].

4.2 Input Traffic

4.2.1 Traffic Models

Characterization of ATM source traffic is known to be a challenging task and an appropriate model has not been agreed upon. For example, one of the most significant characteristics of an ATM source, *burstiness*, is not yet uniquely defined. In the experiments in this chapter, a traffic model which is considered to approximate some ATM sources is used, knowing that this model may not sufficiently represent all ATM sources. A traffic source of a Traffic Class is created by a superposition of mini-sources, the models of which are defined in the following.

The bursty traffic model of a mini-source used here is an Interrupted Bernoulli Process (IBP, see e.g., [MOSM90]), which is one of the commonly used ON-OFF source models in ATM network research. An IBP is defined as follows:

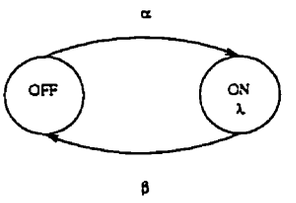

Figure 4.1: State transition diagram of an IBP

Definition 9 (Interrupted Bernoulli Process)

An IBP is defined by the 3-tuple (α, β, λ) *such that:*

1. *An IBP source is defined as a two-state, ON and OFF, discrete-time Markov model.*
2. *The state transition from OFF state to ON state occurs with probability* α *per slot and from ON state to OFF state with probability* β *per slot.*
3. *At most one cell is generated in each time slot.*
4. *While in OFF state, no cell is generated.*
5. *While in ON state, a cell is generated with probability* λ *per slot.*

(See Figure 4.1.)

A Bernoulli process is used to model a mini-source of non-bursty traffic. The definition of Bernoulli process is as follows:

Definition 10 (Bernoulli Process)

A Bernoulli process is defined by the parameter, λ_B*, and at most one cell is generated with probability* λ_B *per slot.*

Note that λ_B is the average cell generation rate of a Bernoulli process. A Bernoulli process can be seen as a special case of an IBP specified in the following:

Remark 1 *A Bernoulli process with the parameter λ_B is equivalent to an IBP with the parameters $(\alpha, \beta, \lambda) = (1, 0, \lambda_B)$.*

We construct a traffic source of a Traffic Class by a superposition of statistically identical IBP mini-sources. The statistical characteristics of an IBP mini-source are determined by three parameters, (α, β, λ); a different parameter value set creates a different type of IBP mini-source. In order to distinguish IBP mini-sources of different characteristics, the notion of *Traffic Type* is introduced. A Traffic Type determines the statistical characteristics of a mini-source. We define *Traffic Type* in the following:

Definition 11 (Traffic Type)

A Traffic Type i, $i = 1, 2, \cdots$, specifies the parameter values, $(\alpha_i, \beta_i, \lambda_i)$. An IBP mini-source is said to be Type i if its parameter values $(\alpha, \beta, \lambda) = (\alpha_i, \beta_i, \lambda_i)$.

A Traffic Class is determined by the number of IBP mini-sources and the Traffic Type to which the mini-sources belong. The definition of Traffic Class in this chapter is thus given as follows:

Definition 12 (Traffic Class)

Traffic Class k, $k = 1, 2, \cdots, K$, is determined by the number of IBP mini-sources n_k and their Traffic Type i_k. The traffic flow of Traffic Class k is generated by a superposition of n_k IBP mini-sources of Traffic Type i_k.

The Traffic Types used to create the results presented in this chapter are shown in Table 4.1. The characteristics of each Traffic Type presented in Table 4.1 are the following: α, β, and λ are the parameters of an IBP mini-source; $\overline{\lambda_i}$ denotes the average cell generation rate; and $\overline{p}_i \times \lambda_i$ is an indicator of *burstiness*, where $\overline{p}_i$ denotes the average peak duration time and λ_i is the peak cell generation rate. Other Traffic Types are also used to create the results provided in Appendix B.

Table 4.1: Traffic Types

Type	α_i	β_i	λ_i	$\overline{\lambda_i}$	$\overline{p}_i \times \lambda_i$
Type 1	1.0	0.0	0.03	0.03	–
Type 2	0.001	0.002	0.09	0.03	45
Type 4	0.0005	0.001	0.09	0.03	90
Type 8	0.001	0.002	0.14	0.0466 $\cdots$	70
Type 9	0.002	0.006	0.16	0.04	26.6 $\cdots$
Type 10	1.0	0.0	0.0225	0.0225	–
Type 11	0.001	0.002	0.0675	0.0225	33.75
Type 12	0.0005	0.001	0.0675	0.0225	67.5
Type 14	0.001	0.002	0.055	0.018	27.5

In our experiments, the values, 7, 9, 10, and 20, are chosen for the number of mini-sources in a Traffic Class, n_k. These are the lowest numbers of ON-OFF mini-sources to be superposed in order to obtain the effect of sufficient burstiness and non-trivial characteristics in the aggregate traffic (See, e.g., [SW86, DL86, HL86]). The average load of each mini-source is set so that the total average utilization becomes 90%. The average peak rates are set in the range of $10M$ bps to $15M$ bps; and the average peak duration is set in the range of 167 cell times to 1000 cell times. The average peak rate and the average peak duration used here are approximately

within the range of the video traffic used in Section 5.6, although adjustments are made to obtain 90% average utilization by using a shorter silence period.

It should be noted that $\overline{p}_i \times \lambda_i$ is shown tc be insufficient to represent burstiness in terms of resultant cell loss ratio (See Appendix B). In fact, resource dimensioning regarding cell loss performance is still an open problem. The resource dimensioning for our experiment is performed empirically. We do not delve into the resource dimensioning problem since our emulation is time consuming and thus it is difficult to obtain a sufficient amount of data to discuss resource dimensioning.

4.3 Characteristics of the Work-conserving WRR-CSVP System

The characteristics of the Work-conserving WRR-CSVP resource allocation mechanism are examined by emulation. The results in this section are produced using Exhaustive service with a short service cycle. It has been confirmed that the results with Round Robin service are approximately the same for this service cycle length. The random selection method is used for both pushout operation and time slot reassignment operations.

4.3.1 Advantages of the Work-conserving WRR Scheduling Scheme

The Work-conserving WRR scheduling can be seen as a capacity allocation by the CSVP strategy, while the WRR scheduling scheme can be seen as a capacity

allocation by the CP strategy. Since the Work-conserving WRR scheduling scheme maximally utilizes the link capacity, we expect that it exhibits a better aggregate cell loss performance than the WRR scheduling scheme. As discussed in Chapter 1, our interest is in the VC based scheduling schemes and thus the capacity allocation by the CS strategy, first-come-first-served, is not considered.

It is important to examine the Work-conserving WRR-CSVP resource allocation mechanism in a heterogeneous traffic situation, since the actual ATM network situation is expected to be heterogeneous. Let us examine the cell loss improvement by the Work-conserving WRR scheduling scheme using a heterogeneous traffic situation, Case 1, shown in Table 4.2. Each Traffic Class receives a buffer space of 300 cells by the CSVP buffer allocation scheme, i.e., $B_k = 300$, $k = 1, 2, 3$; and 1/3 of the total link capacity by the Work-conserving WRR and the WRR scheduling schemes, i.e., $S = 9$ and $S_k = 3$, $k = 1, 2, 3$.

Table 4.2: Heterogenous traffic situation

	Class 1	Class 2	Class 3
Case 1	Type 1 × 10	Type 2 × 10	Type 4 × 10

Table 4.3 shows that the Work-conserving WRR scheduling scheme achieves much better cell loss performance than the WRR scheduling scheme. Both the cell delay and the cell delay variation performances are also improved by the Work-conserving WRR scheduling scheme (Table 4.4). Evidently, the Work-conserving WRR scheduling scheme has a clear advantage. The better performances of the Work-conserving WRR scheduling scheme are achieved by utilizing a fairly large

Table 4.3: Cell loss performance ($\times 10^{-3}$, 95% confidence interval)

	Cell loss ratio	
	WRR	WC-WRR
Aggregate	4.1 ± 0.35	0.25 ± 0.11
Class 1	(undetected)	(undetected)
Class 2	2.9 ± 0.26	0.15 ± 0.10
Class 3	9.5 ± 0.97	0.60 ± 0.27

Table 4.4: Cell delay and cell delay variation performances ($\times 10^{2}$, 95% confidence interval)

(cell time)	Mean cell delay		Standard deviation	
	WRR	WC-WRR	WRR	WC-WRR
Aggregate	3.5 ± 0.12	0.96 ± 0.054	5.3 ± 0.13	2.2 ± 0.12
Class 1	0.16 ± 0.0010	0.081 ± 0.0015	0.14 ± 0.0016	0.100 ± 0.0022
Class 2	3.9 ± 0.11	1.1 ± 0.058	4.6 ± 0.14	1.9 ± 0.086
Class 3	6.4 ± 0.27	1.7 ± 0.012	6.7 ± 0.14	3.1 ± 0.020

amount of slots wasted by the WRR scheduling scheme (Table 4.5). These results

Table 4.5: Wasted slot rates ($\times 10^{-1}$, 95% confidence interval)

	Wasted slot rate
Aggregate	1.0 ± 0.020
Class 1	0.99 ± 0.0050
Class 2	1.0 ± 0.012
Class 3	1.1 ± 0.052

indicate that the Work-conserving WRR scheduling scheme is preferable for a better performance achieved by the efficient use of link capacity, if the Work-conserving WRR scheduling scheme is implementable. Similar results for the different cases can be found in Appendix B.

4.3.2 Advantages of the CSVP Scheme

Let us now examine the CSVP buffer allocation scheme. The most important advantages of the CSVP buffer allocation scheme previously discussed are the efficient use of buffer space and the robust allocation of buffer space. These advantages are demonstrated by evaluating cell loss performance. In addition, another advantage is discovered from the observations of cell delay and cell delay variation performances for different virtual partition.

Buffer Efficiency

If the buffer space is completely shared such as in the CS or the CSVP buffer allocation schemes, it is known to achieve better buffer efficiency because in such a

buffer allocation scheme, an arriving cell is admitted as long as there is an empty space in the buffer and thus the entire buffer space is maximally utilized. The buffer efficiency of the CSVP scheme is demonstrated by comparing its aggregate cell loss performance with that of the CP scheme, where the buffer space is partitioned and not shared. This aspect of the CSVP scheme was observed numerically in [WM95] for a fluid-flow model. Here, it is shown that our discrete traffic system is also buffer efficient.

Table 4.6 compares the aggregate cell loss performance of the CP, the CS and the CSVP buffer allocation schemes. The aggregate cell loss ratios of the CS and

Table 4.6: The aggregate cell loss performance of the CP, CS, and CSVP buffer allocation schemes ($\times 10^{-3}$, 95% confidence interval)

	CP	CS	CSVP
Aggregate	3.0 ± 0.11	0.28 ± 0.078	0.25 ± 0.11

the CSVP schemes are less than a tenth of that of the CP scheme due to the buffer efficiency. It should be noted that the Work-conserving WRR-CSVP mechanism and the Work-conserving WRR-CS mechanism exhibit the same aggregate cell loss ratio due to Conservation Law (Proposition 1 in Chapter 2).

In order to achieve the level of cell loss performance of Traffic Classes 2 and 3 achieved by the CSVP scheme, where buffer space of 300 cells is allocated to each Traffic Class, buffer spaces of about 600 cells to Traffic Class 2 and 700 cells to Traffic Class 3 are necessary by the CP scheme. The CSVP scheme requires approximately half of the buffer space than the CP scheme for these Traffic Classes. No cell losses were detected for Traffic Class 1 in either scheme. These results show

Table 4.7: The cell loss performance of the CP and CSVP buffer allocation schemes ($\times 10^{-3}$, 95% confidence interval)

	CP (600 cells to Class 2)	CP (700 cells to Class 3)	CSVP
Class 2	0.16 ± 0.10	-	0.15 ± 0.10
Class 3	-	0.56 ± 0.27	0.60 ± 0.27

the efficient use of buffer space of the CSVP scheme and indicate that the CSVP scheme is effective when the buffer space is limited and bursty traffic is supported.

Robustness of Buffer Allocation

The CS and CSVP buffer allocation schemes exhibit a better aggregate cell loss ratios than the CP scheme. What are the cell loss ratios experienced by the Traffic Classes? Who benefits from the efficient use of buffer space? The CP scheme may not be buffer efficient. However, it guarantees a minimum buffer space to each Traffic Class and thus achieves robustness of buffer allocation. The buffer space allocated to each Traffic Class is guaranteed in the CP scheme and cannot be violated by the other Traffic Classes. On the other hand, in the CS scheme, there is no guarantee. Therefore, although the CS scheme may be buffer efficient and attain a better aggregate cell loss performance, some Traffic Classes may suffer from a larger cell loss due to temporarily reduced buffer space. The CSVP scheme guarantees a minimum buffer space to each Traffic Class by the virtual partition and the pushout operation, while achieving a completely shared buffer space.

Table 4.8 compares the cell loss ratio by the CP, the CS and the CSVP buffer allocation schemes using Case 1. In Case 1, Traffic Class 1 is non-bursty and thus

Table 4.8: The cell loss performance of the CP, CS, and CSVP buffer allocation schemes ($\times 10^{-3}$, 95% confidence interval)

	CP	CS	CSVP
Class 1	(undetected)	0.21 ± 0.058	(undetected)
Class 2	1.9 ± 0.35	0.29 ± 0.08	0.15 ± 0.10
Class 3	7.2 ± 0.29	0.35 ± 0.10	0.60 ± 0.27

should not suffer high cell loss ratio if a buffer space of 300 cells is guaranteed. In fact, for the CP scheme, no cell losses were detected for Traffic Class 1. However, this class suffers from a high cell loss ratio under the CS scheme. By the CSVP scheme, no cell losses for Traffic Class 1 were detected. The CSVP scheme guarantees a buffer space of 300 cells for Traffic Class 1. The cell loss ratio of Traffic Class 2 is also slightly improved by the CSVP scheme. This better cell loss performance for Traffic Classes 1 and 2 is actually accomplished by limiting the buffer subscription of Traffic Class 3, the most bursty traffic; the cell loss ratio of Traffic Class 3 by the CSVP scheme is larger than that of the CS scheme. The CSVP buffer allocation scheme limits the buffer occupancy of bursty traffic and protects non-bursty traffic, while achieving better aggregate cell loss performance for a given total buffer size.

These results show that the CSVP scheme is suitable for heterogeneous traffic situations, where bursty and non-bursty traffic coexist, to protect non-bursty traffic. Undetected cell loss is guaranteed for non-bursty traffic if a sufficient amount of buffer is virtually allocated.

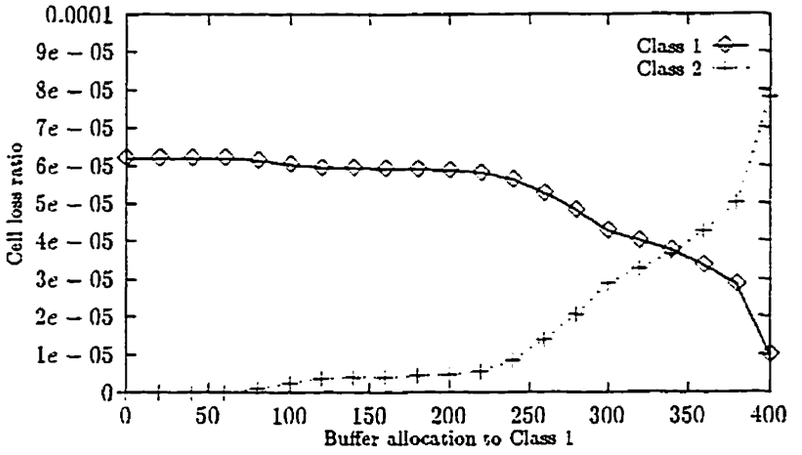

Figure 4.2: Virtual partitioning vs. cell loss ratio (Case 2)

Virtual Partitioning

It is shown in [WM95] that cell loss performance is sensitive to the virtual partitioning of the CSVP buffer allocation scheme. For the case of two Traffic Classes, a non-increasing relation between the virtually allocated buffer space and the cell loss ratio is shown (Theorem 2). Figure 4.2 illustrates this relation using Case 2 shown in Table 4.9. The total buffer size is 400 cells, i.e., $B = 400$; and 3/5 of

Table 4.9: Two Traffic Class case

	Class 1	Class 2
Case 2	Type 8 × 9	Type 9 × 7

link capacity is allocated to Traffic Class 1 and 2/5 to Traffic Class 2, i.e., $S = 5$, and $S_1 = 3$ and $S_2 = 2$.

Here, we investigate the case of three Traffic Classes. Figure 4.3 shows the cell

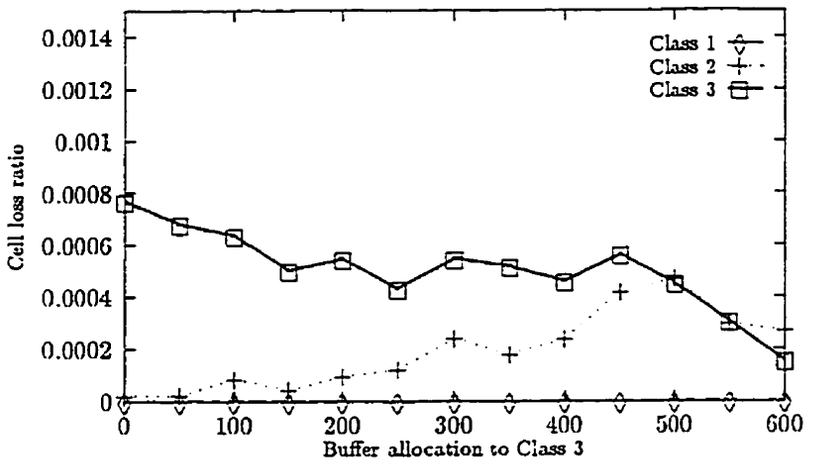

Figure 4.3: Virtual partitioning vs. cell loss ratio (Case 1)

loss performance with various virtual partitionings using Case 1. The sensitivity of cell loss is evident but not the non-increasing relation. The fluctuation of the cell loss ratios between allocations 150 and 500 is within the range of the confidence interval. However, the cell loss ratio of Traffic Class 2 from allocations 500 to 600 shows some singularity. It appears that the non-increasing relation may not hold at singular points, such as when a buffer space of 600 cells is allocated to Traffic Class 1 in this case. An analytical approach may be necessary to understand this phenomenon.

Are the cell delay and the cell delay variation performances sensitive too? Interestingly, they seem insensitive to the virtual partitioning (Figure 4.4). The same tendency is observed for Case 2 (See Appendix B). Cell delay and cell delay variation performance is determined by the service scheduling scheme if an infinite buffer is assumed. In our experiments, the buffer space is sufficiently large to achieve very

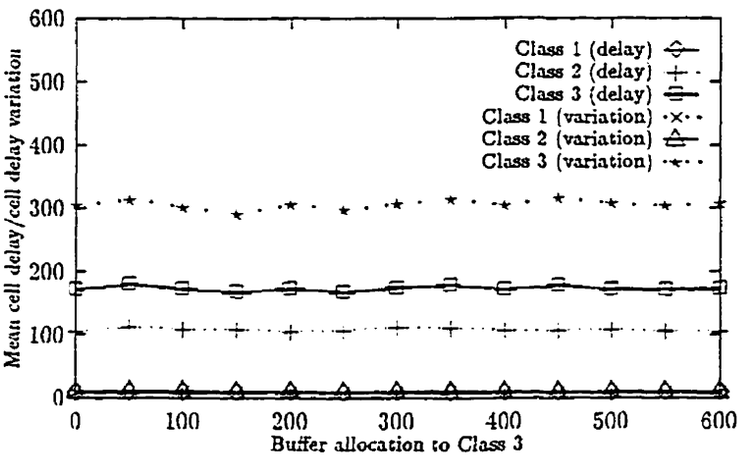

Figure 4.4: Virtual partitioning vs. mean cell delay and cell delay variation (Case 1)

small cell loss ratios. Therefore, the impact of the buffer size to cell delay and cell delay variation may be minimal, and the effect of virtual partitioning on cell delay and cell delay variation is small.

The insensitivity of cell delay and cell delay variation to virtual partitioning suggests that the cell loss performance of Traffic Classes can be adjusted by an appropriate virtual partitioning without affecting the cell delay and the cell delay variation performances. The virtual partition can be a useful tool to tune cell loss performance.

4.4 Design Guidelines

The following design issues were left inconclusive in the discussion of the algorithm design in Chapter 3:

1. mechanics of service to implement the WRR scheduling scheme.

2. traffic class selection method for

   (a) the pushout operation of the CSVP buffer allocation scheme and

   (b) the time slot reassignment operation of the Work-conserving WRR scheduling scheme.

Two mechanics of service, Exhaustive service and Round Robin service, are proposed to implement the WRR scheduling scheme, and various methods are suggested for traffic class selection. In this section, these alternatives are compared for cell loss performance in order to provide design guidelines.

4.4.1 Mechanics of Service

As discussed in Section 3.4.1, for the WRR scheduling scheme, Exhaustive service is known to have a performance degradation problem. Round Robin service solves the problem if the capacity is evenly allocated, where the pattern of service is the same for any length of service cycle. When the link capacity is unevenly distributed, however, Round Robin service may not be able to provide sufficient compensation for a longer service cycle. Case 3 shown in Table 4.10 is introduced to create an uneven capacity allocation with an uneven traffic situation. Traffic Types 10 to 12 are created by modifying the peak generation rate, λ_i, of Traffic Types 1, 2, and 4, which comprise Case 1. The average cell generation rate of Traffic Class 3 is twice as much as that of the other Traffic Classes. Thus, twice as much capacity and buffer space is given to Traffic Class 3. Each of Traffic Classes 1 and 2 receives 1/4

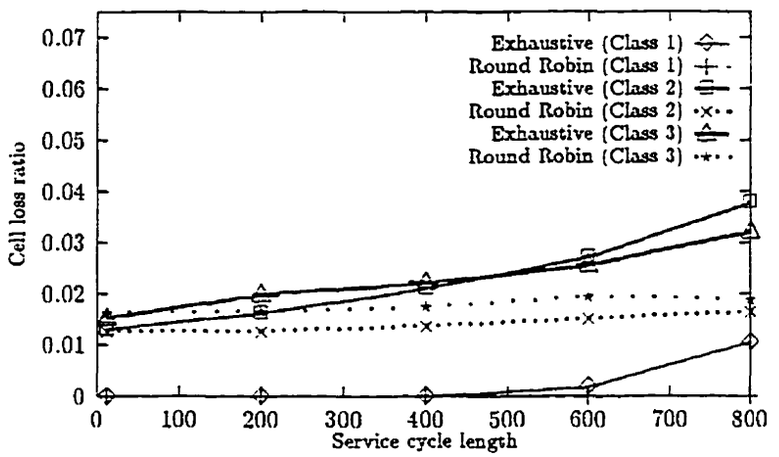

Figure 4.5: Service cycle length vs. cell loss ratio (WRR)

of the total capacity and Traffic Class 3 receives 1/2 by $S = 12$, and $S_1 = S_2 = 3$ and $S_3 = 6$. The total buffer size is 400 cells; and 100 cells are allocated to each of Traffic Classes 1 and 2 and 200 cells to Traffic Class 3, i.e., $B_1 = B_2 = 100$ and $B_3 = 200$. In the following, the two services are examined using Case 3. Similar

Table 4.10: Uneven traffic situation

	Class 1	Class 2	Class 3
Case 3	Type 10 × 10	Type 11 × 10	Type 12 × 20

results for another uneven case are provided in Appendix B.

Let us first examine the cell loss performance. It is evident in Figure 4.5 that, with Exhaustive service, the cell loss ratio increases with the length of service cycle. Round Robin service reduces the cell loss ratio somewhat.

Figure 4.6 indicates that not only the cell loss performance but also the cell

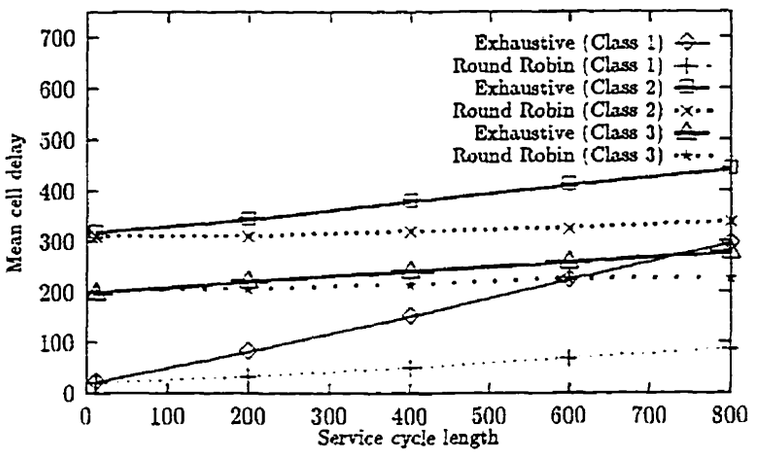

Figure 4.6: Service cycle length vs. mean cell delay (WRR)

delay performance degrades with Exhaustive service when the service cycle is longer. Round Robin service reduces the problem. On the other hand, the effect on the cell delay variation performance is not straightforward (Figure 4.7). The cell delay variation of Traffic Class 1, the non-bursty class, increases, while that of Traffic Classes 2 and 3, the bursty classes, decreases slightly by Exhaustive service. Round Robin service reduces the problem.

Does the same problem occur for the Work-conserving WRR scheduling scheme? Figure 4.8 shows the effect of the length of service cycle on the cell loss performance when the Work-conserving WRR scheduling scheme is adopted. The cell loss performance degradation with Exhaustive service for a long service cycle is much smaller than that of the WRR scheduling scheme. The problems of a long service cycle seem to be somewhat compensated for by the time slot reassignment operation.

With Exhaustive service, the mean cell delay of Traffic Class 1 increases for a

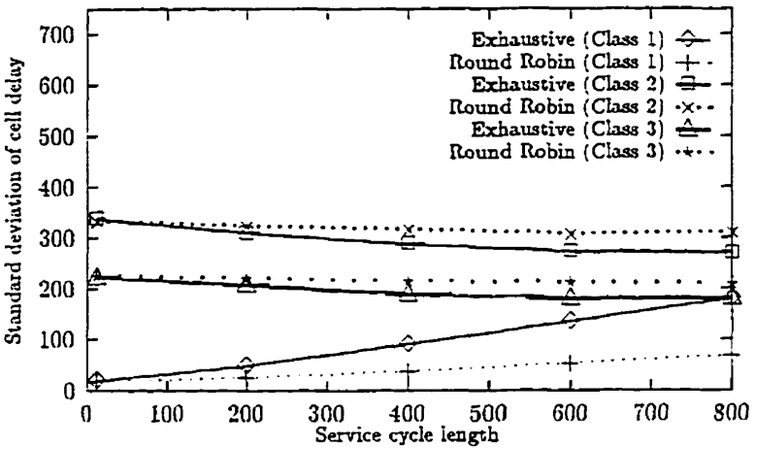

Figure 4.7: Service cycle length vs. cell delay variation (WRR)

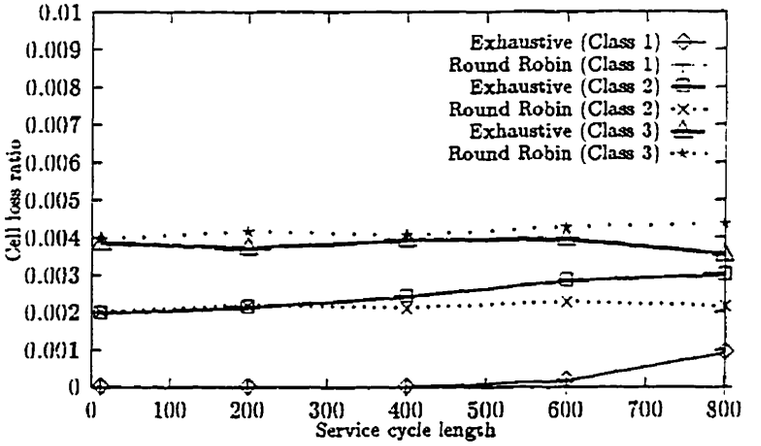

Figure 4.8: Service cycle length vs. cell loss ratio (Work-conserving WRR)

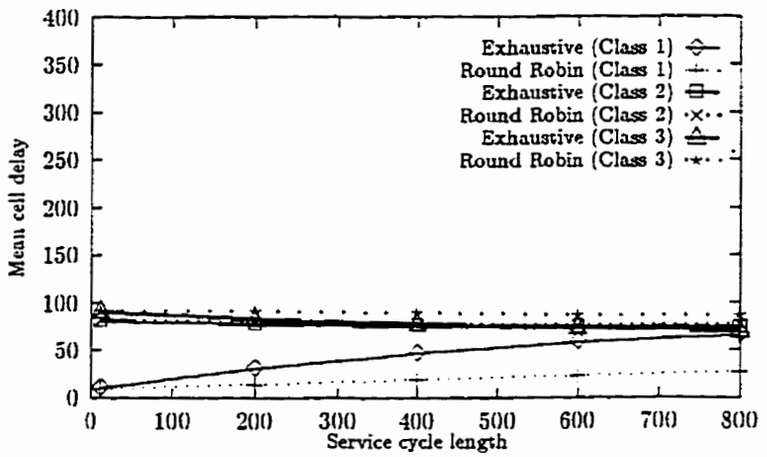

Figure 4.9: Service cycle length vs. mean cell delay (Work-conserving WRR)

longer service cycle but that of the other Traffic Classes does not seem to change (Figure 4.9). Round Robin service provides sufficient compensation. Figure 4.10 shows that the cell delay variation of Traffic Class 1 increases with Exhaustive service, and Round Robin service reduces the problem.

These observations show that, when the Work-conserving scheduling scheme is adopted, Exhaustive service functions sufficiently well to provide low cell loss performance. Therefore, Exhaustive service may be adopted for its simpler operation if cell loss is more important performance measure. However, Round Robin service may be necessary for a longer service cycle if the cell delay or the cell delay variation is of concern. In addition, as already pointed out in Section 3.4.1, Exhaustive service may cause undesirable bursts in the output traffic. Round Robin service may still be preferable.

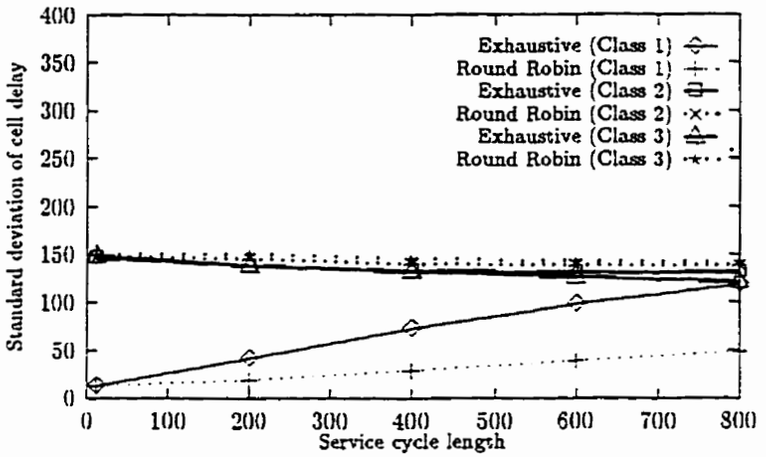

Figure 4.10: Service cycle length vs. cell delay variation (Work-conserving WRR)

4.4.2 Traffic Class Selection Methods

The traffic class selection methods proposed for the pushout and the time slot reassignment operations in Chapter 3 are categorized into the following three kinds:

1. random selection method,
2. methods based on buffer occupancy, and
3. prioritization method.

The methods belonging to the second category for the pushout operation are the following two:

1. Largest Excess by Absolute Amount, and
2. Largest Excess by Relative Amount,

and for the the time slot reassignment, the following three:

1. Largest Occupancy by Absolute Amount,

2. Largest Occupancy by Relative Amount, and

3. Largest Excess by Absolute Amount.

The random selection method is used as a point of reference; the methods belonging to the second and the third categories are examined in comparison to the random selection method. Recall that random selection may be necessary to break ties for the methods of the second and the third categories. Therefore, the random selection method may have to be implemented after all. Cell loss performance is sensitive to virtual partitioning while cell delay and cell delay variation are insensitive. In fact, these measures are observed to be insensitive to the methods discussed here. Therefore, we focus on the cell loss performance. The results in the following are produced using Exhaustive service with a short service cycle. It has been confirmed that the results with Round Robin service are approximately the same for this service cycle length.

Methods Based on Buffer Occupancy

Three steps are taken to investigate the traffic class selection methods based on buffer occupancy. First, we examine the pushout operation. Next, the time slot reassignment operation is dealt with. Then, a combination of the pushout and the time slot reassignment operations is discussed. Recall that when the buffer space is equally allocated to Traffic Classes, the methods based on buffer occupancy are equivalent. Therefore, we examine the methods using an uneven traffic situation,

Case 3, where the buffer space is unevenly allocated. Similar results for another uneven case can be found in Appendix B.

Pushout The largest excess by absolute amount and the largest excess by relative amount traffic class selection methods are examined. Recall that these methods require $K-2$ comparisons. It is expected that the bursty traffic, especially the burstiest Traffic Class 3, is pushed out most frequently and experiences an increased cell loss ratio with the methods based on buffer occupancy. The random selection method may have a similar effect since with this method, the frequency of over-subscription of the buffer space by a Traffic Class determines the frequency with which the Traffic Class is pushed out. Table 4.11 shows that neither the largest excess by absolute amount nor the largest excess by relative amount increases the cell loss ratio noticeably from that of random selection. Therefore, the random

Table 4.11: Cell loss ratio of different traffic class selection methods for the pushout operation ($\times 10^{-3}$, 95% confidence interval)

	Random	Absolute excess	Relative excess
Class 1	(undetected)	(undetected)	(undetected)
Class 2	2.0 ± 0.35	1.9 ± 0.26	2.2 ± 0.31
Class 3	3.9 ± 0.57	4.2 ± 0.51	4.3 ± 0.50

selection method, which requires the use of a pseudo-random number generator, may be preferred to avoid increasing number of comparisons when the number of VCs, K, increases.

Time Slot Reassignment For the time slot reassignment operation, the largest occupancy by absolute amount, largest occupancy by relative amount, and largest excess by absolute amount traffic class selection methods are examined. Recall that these methods require $K-2$ comparisons. We expect that the most bursty traffic, Traffic Class 3, receives a larger amount of time slot reassignments and thus shows a smaller cell loss ratio with these methods. The random selection method may perform similarly since with this method, the frequency of subscription of the buffer space by a Traffic Class determines the frequency with which the Traffic Class receives a reassigned time slot. Table 4.12 shows that these methods yield similar cell loss performance. Therefore, the random selection method, which requires the

Table 4.12: Cell loss ratio of different traffic class selection methods for the time slot reassignment operation ($\times 10^{-3}$, 95% confidence interval)

	Random	Absolute occup.	Relative occup.	Absolute excess
Class 1	(undetected)	(undetected)	(undetected)	(undetected)
Class 2	2.0 ± 0.35	2.3 ± 0.34	2.2 ± 0.23	2.2 ± 0.32
Class 3	3.9 ± 0.57	4.0 ± 0.62	4.1 ± 0.51	4.0 ± 0.48

use of a pseudo-random number generator, may be preferred to avoid increasing number of comparisons when the number of VCs, K, increases.

Combination of the Two The combination of the different traffic class selection methods for the pushout and the time slot reassignment may intensify the effect of these operations and create meaningful differences in the cell loss performance. Table 4.13 compares various combinations. As seen in Table 4.13, no combination of the traffic class selection methods for the pushout and the time slot reassignment

Table 4.13: Cell loss ratio of different combinations of traffic class selection methods for the pushout and the time slot reassignment operations ($\times 10^{-3}$, 95% confidence interval)

time slot	Random	Absolute occupancy	
pushout	Random	Absolute excess	Relative excess
Class 1	(undetected)	(undetected)	(undetected)
Class 2	2.0 ± 0.35	2.6 ± 0.45	2.9 ± 0.21
Class 3	3.9 ± 0.57	4.5 ± 0.44	4.6 ± 0.48

time slot	Relative occupancy		Absolute excess	
pushout	Absolute excess	Relative excess	Absolute excess	Relative excess
Class 1	(undetected)	(undetected)	(undetected)	(undetected)
Class 2	2.2 ± 0.30	2.2 ± 0.36	2.5 ± 0.47	2.4 ± 0.28
Class 3	4.2 ± 0.45	4.4 ± 0.58	4.4 ± 0.39	4.3 ± 0.64

operations creates sufficiently large differences in the cell loss performance to adopt these methods. This result further supports the random selection method.

Prioritization

Our next question is whether prioritization of the traffic class selection for the pushout and time slot reassignment operations creates any effect on cell loss performance. In order to assess the effect of prioritization, we create Case 4 where five identical Traffic Classes, Classes 1 to 5, are supported by the resource allocation mechanism. Each Traffic Class consists of a superposition of ten Type 14 IBP mini-sources. The buffer space and the link capacity are equivalently allocated, i.e., $B_k = 150$, $k = 1, 2, \cdots, 5$, and $S = 15$ and $S_k = 3$, $k = 1, 2, \cdots, 5$. The same three steps are taken: first, we examine the prioritization of the pushout operation; next, that of the time slot reassignment operation; and then, the combination of

the pushout and time slot reassignment operations is discussed.

Pushout The prioritization of the traffic class selection for the pushout operation is examined while the random selection method is adopted for the time slot reassignment operation. We consider the priority settings shown in Table 4.14. The cell loss performances resulting from these priority settings are shown in Table 4.15. When no priority is given, the cell loss performances of the Traffic Classes

Table 4.14: Priority settings of the pushout operation

	Class 1	Class 2	Class 3	Class 4	Class 5
Setting 1	Random selection (no priority)				
Setting 2	high	low	low	low	low
Setting 3	high	high	high	high	low

Table 4.15: Cell loss ratios by different prioritization of the pushout operation ($\times 10^{-3}$, 95% confidence interval)

	Setting 1	Setting 2	Setting 3
Class 1	0.90 ± 0.20	1.2 ± 0.28	0.97 ± 0.29
Class 2	0.90 ± 0.19	0.71 ± 0.015	0.96 ± 0.23
Class 3	0.97 ± 0.24	0.80 ± 0.21	1.0 ± 0.31
Class 4	0.81 ± 0.28	0.71 ± 0.18	1.0 ± 0.24
Class 5	0.91 ± 0.24	0.95 ± 0.31	0.62 ± 0.23

are approximately the same. In Setting 2, Traffic Class 1 receives the priority to be pushed out over Traffic Classes 2 to 5. As a result, Traffic Class 1 experiences a higher cell loss ratio. In Setting 3, Traffic Classes 1 to 4 receive the priority to be pushed out over Traffic Class 5. Thus, Traffic Class 5 experiences a smaller cell

loss ratio than others. However, these differences in cell loss performance are not large enough to make a significant impact. Considering that the priority method requires at most $K-2$ comparisons, and thus becomes computationally expensive when K is large, the contribution of this method to cell loss performance is small.

Time Slot Reassignment Adopting the random selection mechanism for the pushout, we examine the prioritization of traffic class selection for the time slot reassignment operation. The priority settings considered are shown in Table 4.16. Table 4.17 shows the cell loss performance resulting from these various prioritiza-

Table 4.16: Priority settings of the time slot reassignment operation

	Class 1	Class 2	Class 3	Class 4	Class 5
Setting 1	Random selection (no priority)				
Setting 4	high	low	low	low	low
Setting 5	high	high	high	high	low

tions. When no priority is given, the cell loss performances of the Traffic Classes

Table 4.17: Cell loss ratios by different prioritization of the time slot reassignment operation ($\times 10^{-3}$, 95% confidence interval)

	Setting 1	Setting 4	Setting 5
Class 1	0.90 ± 0.20	0.44 ± 0.12	0.79 ± 0.25
Class 2	0.90 ± 0.19	0.83 ± 0.30	0.70 ± 0.15
Class 3	0.97 ± 0.24	0.87 ± 0.25	0.70 ± 0.13
Class 4	0.81 ± 0.28	0.92 ± 0.34	0.81 ± 0.22
Class 5	0.91 ± 0.24	0.86 ± 0.20	1.2 ± 0.34

are approximately the same. In Setting 4, Traffic Class 5 has the priority to re-

ceive reassigned time slots over Traffic Classes 1 to 4. As a result, Traffic Class 5 experiences a smaller cell loss ratio than others. In Setting 5, Traffic Classes 1 to 4 have the priority to receive reassigned time slots over Traffic Class 5. Thus, Traffic Class 5 experiences a larger cell loss ratio than others. However, these differences are not large enough to make a significant impact. Considering that the priority method requires at most $K-2$ comparisons, and thus becomes computationally expensive when K is large, the contribution of this method to cell loss performance is small.

Combination of the Two Our question now is if the combination of the prioritizations of the pushout and the time slot reassignment can do any better. Table 4.18 shows the different priority setting combinations of the traffic class selection of the pushout and the time slot reassignment operations. As we can see in Table 4.19.

Table 4.18: Priority settings of the pushout and the time slot reassignment operations (p: pushout, t: time slot reassignment)

		Class 1	Class 2	Class 3	Class 4	Class 5
Setting 1	p	Random selection (no priority)				
	t	Random selection (no priority)				
Setting 6	p	high	low	low	low	low
	t	high	low	low	low	low
Setting 7	p	high	low	low	low	low
	t	low	high	high	high	high
Setting 8	p	high	high	high	high	low
	t	high	high	high	high	low
Setting 9	p	high	high	high	high	low
	t	low	low	low	low	high

the contribution of these settings to the cell loss performance is not sufficiently

large considering its computational cost for comparisons. Therefore, it is recom-

Table 4.19: Cell loss ratios by different prioritization of the pushout and the time slot reassignment operations ($\times 10^{-3}$, 95% confidence interval)

	Setting 1	Setting 6	Setting 7	Setting 8	Setting 9
Class 1	0.90 ± 0.20	0.88 ± 0.20	0.73 ± 0.19	0.96 ± 0.21	0.47 ± 0.15
Class 2	0.90 ± 0.19	0.81 ± 0.19	0.67 ± 0.10	1.0 ± 0.22	1.4 ± 0.25
Class 3	0.97 ± 0.24	0.89 ± 0.33	0.60 ± 0.17	0.98 ± 0.25	1.1 ± 0.20
Class 4	0.80 ± 0.28	0.88 ± 0.23	0.66 ± 0.20	1.1 ± 0.29	1.3 ± 0.27
Class 5	0.91 ± 0.24	0.65 ± 0.16	1.4 ± 0.31	0.94 ± 0.24	1.2 ± 0.29

mended to adopt the random selection method for both the pushout and the time slot reassignment operations.

4.5 Concluding Remarks

The work-conserving WRR scheduling scheme considerably improves cell loss, cell delay, and cell delay variation performance compared to the WRR scheduling scheme by the efficient use of link capacity. The improvement is large enough to consider the addition of the time slot reassignment operation in spite of the increased computational cost. Note that minimum bandwidth guarantee is inherent to the Work-conserving WRR scheduling scheme. The buffer efficiency of the CSVP buffer allocation scheme is clearly observed. It is also demonstrated that the CSVP scheme guarantees a minimum buffer space to each Traffic Class. Thus, the Work-conserving WRR-CSVP mechanism exhibits the efficient use of resources and the robustness of resource allocation suitable for an output port of an ATM switch to perform VC based resource allocation.

Another advantage of the CSVP scheme observed is that cell loss performance is sensitive to virtual partitioning, while cell delay and cell delay variation performances are not. This suggests that cell loss performance can be adjusted by virtual partitioning without affecting cell delay and cell delay variation. Further investigation is conducted in Chapter 5 regarding the effect of virtual partitioning on the end-to-end performance.

The two service mechanics of the WRR scheduling scheme, Exhaustive and Round Robin services, are compared. The adoption of the time slot reassignment operation reduced the drawback of Exhaustive service in cell loss performance. Therefore, Exhaustive service may be acceptable for its less computational cost. However, a problem still exists where cell delay and cell delay variation are concerned. Round Robin service showed better performance. Also, Exhaustive service may cause undesirable bursts in the output traffic and increase the cell loss at the downstream nodes. Thus, if feasible, Round Robin service may be recommended. We attempt to capture this burst problem of Exhaustive service by end-to-end performance evaluation in Chapter 5.

The traffic class selection methods for the pushout and the time slot reassignment operations are also examined. None of the traffic class selection methods based on buffer occupancy result in significantly different cell loss performance from the random selection method. The contribution of prioritization is also small. The random selection method does not require comparisons and thus may be computationally less expensive, although the use of a pseudo-random number generator is necessary. In addition, the random selection method may be necessary to break

ties after all. Therefore, the adoption of the random selection method for both the pushout and and the time slot reassignment operations is recommended.

The Work-conserving WRR-CSVP resource allocation mechanism performs well and possesses some preferable characteristics. In Chapter 5, we further investigate the characteristics of the resource allocation mechanism from the perspective of end-to-end performance. With all these advantages, the Work-conserving WRR-CSVP resource allocation mechanism is recommendable for adoption at an output port of an ATM switch if the computational cost can be overcome. In order to cope with the speed of ATM networks, it is preferred to implement the resource allocation mechanisms in hardware. We discuss the implementation strategies for the Work-conserving WRR-CSVP resource allocation mechanism and examine the feasibility in Chapter 6.

Chapter 5

End-To-End Network Performance

5.1 Introduction

It is shown by emulation in Chapter 4 that the Work-conserving WRR-CSVP resource allocation mechanism achieves efficient use of resources and robustness in resource allocation at a single generic node. However, as mentioned in Section 1.1.5, QoS has to be provided on an end-to-end basis in ATM networks. Thus, it is important to examine the Work-conserving WRR-CSVP resource allocation mechanism in a network environment and to evaluate its end-to-end performance. For this purpose, a network emulator is developed and the end-to-end performance is evaluated in this chapter.

As pointed out in Section 3.4.1, Exhaustive service may cause undesirable bursts in the output traffic when the service cycle is long. We attempt to demonstrate

this problem.

One of the advantages of the Work-conserving WRR-CSVP resource allocation mechanism discussed in Section 4.3.2 is the effect of virtual partitioning. Virtual partitioning of buffer space affected cell loss performance but did not show noticeable effect on cell delay and cell delay variation. If this is also true for end-to-end performance, then virtual partitioning can be used to adjust cell loss without affecting cell delay and cell delay variation. We examine the effect of virtual partitioning on the end-to-end performance and the performance at the downstream nodes.

It is also of interest to examine our resource allocation mechanism for practical traffic. The Motion Pictures Expert Group (MPEG) coding scheme [Gal91] is an accepted standard coding scheme that is currently in use. The traffic created by this coding method is known to be very bursty. We have created traffic flows from existing MPEG coded pictures as test sources to further examine the advantages of the Work-conserving WRR-CSVP resource allocation mechanism in an end-to-end context.

5.2 Network Model

5.2.1 Topology of the Network

An ATM network is a mesh network of ATM switches with the ATM multiplexers at the network entry points and the ATM demultiplexers at the network exit points. In Chapter 2, we modeled an ATM multiplexer or an output port of a non-blocking output-buffering ATM switch using a single-server model. An end-to-end network

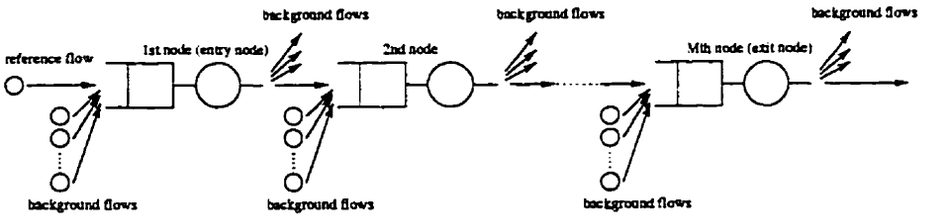

Figure 5.1: The path of the reference flow

partition is modeled by a tandem connection of single-server queues, as shown in Figure 5.1. The traffic flow that traverses this network from the entry node to the exit node is referred to as the *Reference Class*; and the other traffic flows supported by the nodes on the path are referred to as *Background Classes*.

It should be noted that, in reality, there may be a strong correlation between a Background Class at one node and a Background Class at another node. For example, a Background Class may take the same path as the Reference Class. However, for simplicity of the emulator, we assume that all Background Classes are independent.

5.2.2 Network Emulator

An emulator of an end-to-end network is developed by connecting the generic node emulators in tandem. A few functions are added to the generic node emulator to construct a tandem network. At the interior and the exit nodes, an input link replaces one of the traffic sources; an upstream node is connected to this link. When a cell belonging to the Reference Class departs an upstream node, the cell is transferred to the link connecting to the downstream node. The downstream node

receives this cell in the same manner as it receives cells from the Cell Generator. At the exit node, end-to-end performance statistics are collected for cells belonging to the Reference Class. In each time slot, the set of the operations at a generic node, service scheduling, cell generation, cell admission, and cell departure, is performed from the entry node to the exit node along the path. Also, MPEG data stream sources are added to the Cell Generator. A superposition of MPEG data stream sources constitutes a Traffic Class to obtain a sufficient amount of average load for our emulator, since the average load of each of the MPEG data streams available for us was very low.

5.2.3 Performance Measures

Let M be the number of nodes on the path including the entry and the exit nodes. Therefore, the entry node is the first node and the exit node is the Mth. Let $L^m(n)$, $m = 1, 2, \cdots, M$, be the number of cells belonging to the Reference Class that are blocked at the mth node during $[0, n]$, and let $P^m(n)$, $m = 1, 2, \cdots, M$, be the number of cells belonging to the Reference Class that are pushed out at the mth node during $[0, n]$. Let $U^M(n)$ be the number of cells belonging to the Reference Class that departed from the Mth node during $[0, n]$.

End-to-end Cell Loss Ratio

The end-to-end cell loss ratio is defined as follows:

Definition 13 (End-to-end Cell Loss Ratio)

The end-to-end cell loss ratio, r_E, is defined by:

$$r_E = \lim_{n\to\infty} \frac{\sum_{m=1}^{M}\{L^m(n) + P^m(n)\}}{\sum_{m=1}^{M}\{L^m(n) + P^m(n)\} + U^M(n)}. \tag{5.1}$$

By a stationarity assumption, we have

$$r_E \approx \frac{\sum_{m=1}^{M}\{L^m(N) + P^m(N)\}}{\sum_{m=1}^{M}\{L^m(N) + P^m(N)\} + U^M(N)}. \tag{5.2}$$

for $N \gg 0$.

End-to-end Cell Delay and Cell Delay Variation

End-to-end cell delay is defined only for those cells departing from the exit node. Let d_i^m, $m = 1, 2, \cdots, M$, be the cell delay of the ith departure among the $U^M(n)$ departures of the Reference Class from the mth node. The propagation delay is ignored since it is constant. The end-to-end cell delay is defined by the sum of the cell delays along the path.

Definition 14 (End-to-end Cell Delay)

The end-to-end cell delay, $d_{E,i}$, for the ith departure of the Reference Class from the Mth node is defined as follows:

$$d_{E,i} = \sum_{m=1}^{M} d_i^m \tag{5.3}$$

The mean end-to-end cell delay is defined as follows:

Definition 15 (Mean End-to-end Cell Delay)

The mean end-to-end cell delay, $\bar{d}_E$, is given by:

$$\bar{d}_E = \lim_{n\to\infty} \frac{\sum_{i=1}^{U^M(n)} d_{E,i}}{U^M(n)}. \tag{5.4}$$

The standard deviation around the mean end-to-end cell delay is used to measure the end-to-end cell delay variation.

Definition 16 (End-to-end Cell Delay Variation)

The standard deviation of the end-to-end cell delay , σ_E^d, is given by:

$$\sigma_E^d = \lim_{n\to\infty} \sqrt{\frac{\sum_{i=1}^{U^M(n)} (d_{E,i} - \bar{d}_E)^2}{U^M(n)}}. \tag{5.5}$$

By a stationarity assumption, we have

$$\bar{d}_E \approx \frac{\sum_{i=1}^{U^M(N)} d_{E,i}}{U^M(N)}, \tag{5.6}$$

and

$$\sigma_E^d \approx \sqrt{\frac{\sum_{i=1}^{U^M(N)} (d_{E,i} - \bar{d}_E)^2}{U^M(N)}}, \tag{5.7}$$

for $N \gg 0$.

5.3 Exhaustive Service vs. Round Robin Service

It is pointed out in Section 3.4.1 that Exhaustive service may cause an undesirable burst in the output traffic if the service cycle is long. Exhaustive service allocates consecutive S_k time slots to Traffic Class k in a service cycle. Therefore, a combination of backlog of cells in the buffer and the arrival pattern of a Traffic Class k can result in the emissions of S_k cells consecutively, which can be seen as a burst of length S_k. And the buffer occupancies and arrival patterns of the other Traffic Classes can make this burst longer.

The burst should be able to be detected by observing cell loss performance. Assume that the Background Classes are non-bursty and constantly occupy their share of buffer space. Then, if the Reference Class is bursty, the buffer space needed to support the burst will not be available and result in a higher cell loss ratio. Also, the buffer space for the Background Classes may be violated by the burst of the Reference Class and the Background Classes may experience higher cell loss ratios as well.

In order to create such a situation, the following scenario is prepared. We construct a two-node network. The Reference Class is generated by a superposition of ten statistically identical Bernoulli processes with $\lambda = 0.033$. The two Background Classes at each node are also created with the same traffic model as the Reference Class. Therefore, the average load of each node is 99% and 1/3 of the link capacity is allocated to each Traffic Class. At the first node, a buffer space of 300 cells is allocated to each Traffic Class to enable sufficient backlog in the buffer space. A buffer space of 135 cells is allocated to each class at the second node.

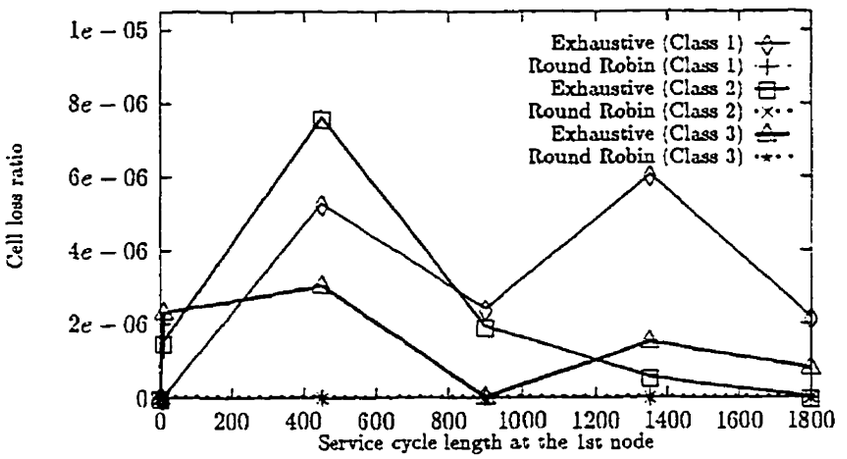

Figure 5.2: Cell loss ratio at the 2nd node

Table 5.1: Cell loss ratio at the 2nd node ($\times 10^{-5}$, 95% confidence interval)

	Exhaustive (service cycle = 450)
Class 1	0.53 ± 0.74
Class 2	0.76 ± 1.2
Class 3	0.30 ± 0.49

which is the minimum buffer space to obtain nearly undetected cell loss. The cell loss performance at the second node is observed to detect the burst.

When Exhaustive service is adopted, some cell loss are detected at the second node for all three Traffic Classes, while no cell loss is detected with Round Robin service (Figure 5.2). Unfortunately, however, this result is inconclusive due to the large confidence interval by our emulator shown in Table 5.1.

Although the creation of bursts by Exhaustive could not be demonstrated, Round Robin service is adopted throughout the remainder of this chapter in order

to eliminate the possibility.

5.4 Efficiency and Robustness

The Work-conserving WRR-CSVP resource allocation mechanism is shown to utilize buffer space efficiently and provide robust buffer allocation at a generic node in Section 4.3.2. Here, we examine if these advantages are extended to the end-to-end performance.

5.4.1 Buffer Efficiency

In order to demonstrate the buffer efficiency of the CSVP buffer allocation scheme, the cell loss performance of the CP and the CSVP schemes are compared in Section 4.3.2. The buffer efficiency of the CSVP scheme results in a better aggregate cell loss ratio at a generic node. The bursty traffic benefits from the buffer efficiency and experiences a lower cell loss ratio. If this is true throughout the network, the aggregate cell loss ratio at each node on the path should be lower.

The network of the CP buffer allocation scheme is created by connecting the generic nodes with the Work-conserving WRR-CP resource allocation mechanism in tandem. We construct a three-node network and use traffic situation Case 1 in Section 4.3.2 as follows. The burstiest traffic, Traffic Class 3, is chosen as the Reference Class. Traffic Classes 1 and 2 are used as the Background at each node, i.e., all nodes have the same Background Class traffic condition. The buffer and capacity allocation conditions at each node are also the same as the generic node of Case 1 in Section 4.3.2.

Table 5.2: The aggregate cell loss ratio at each node ($\times 10^{-3}$, 95% confidence interval)

	CP	CSVP
1st node	3.2 ± 0.29	0.26 ± 0.11
2nd node	1.3 ± 0.12	0.076 ± 0.042
3rd node	0.95 ± 0.28	0.044 ± 0.046

Table 5.3: The end-to-end cell loss ratio ($\times 10^{-3}$, 95% confidence interval)

CP	CS	CSVP
9.6 ± 0.91	0.47 ± 0.14	0.76 ± 0.24

Table 5.2 shows that the aggregate cell loss ratio by the CSVP scheme is much smaller than that by the CP scheme at all nodes. Buffer efficiency of the CSVP scheme is effective throughout the network.

The bursty Reference Class benefits from the efficient use of buffer space, as further discussed in the following subsection. Therefore, the end-to-end cell loss ratio is also smaller by the CSVP scheme (Table 5.3). This means that buffer efficiency of the CSVP scheme is effective also on the end-to-end cell loss performance to support bursty traffic. Note that the CS scheme achieves the best end-to-end cell loss ratio among the three schemes. This low cell loss is achieved by violating the buffer space for the other Traffic Classes as shown in the following.

Table 5.4: The cell loss ratios of Traffic Class 1 ($\times 10^{-3}$, 95% confidence interval)

	CP	CS	CSVP
1st node	(undetected)	0.20 ± 0.050	(undetected)
2nd node	(undetected)	0.13 ± 0.079	(undetected)
3rd node	(undetected)	0.043 ± 0.042	(undetected)

5.4.2 Robustness of Buffer Allocation

In the CS buffer allocation scheme, the buffer space is not guaranteed. Therefore, our emulation study in Section 4.3.2 indicated that the buffer space for the non-bursty traffic was violated by the bursty traffic and resulted in an unnecessarily high cell loss ratio at a generic node. Bursty traffic may violate the buffer space for non-bursty traffic and cause higher cell loss ratio for the non-bursty traffic if the CS scheme is adopted.

The network of the CS buffer allocation scheme is created by connecting the generic nodes with the Work-conserving WRR-CS resource allocation mechanism in tandem. We examine the robustness of buffer allocation by the CSVP scheme, i.e., capability to guarantee a minimum buffer space, by comparing the cell loss performance with that of the CS and the CP schemes.

Table 5.4 shows the cell loss ratios of Traffic Class 1 for Case 1, which is used to examine buffer efficiency in the previous subsection. The non-bursty traffic, Traffic Class 1, suffers from a high cell loss ratio at all nodes with the CS scheme. The buffer space for Traffic Class 1 is protected by the CSVP scheme and no cell loss is detected. The burstiest Reference Class, Traffic Class 3, took advantages of the CS scheme, where no regulation of buffer subscription is adopted, and resulted in a

Table 5.5: The cell loss ratio of Traffic Class 3 as the Reference Class ($\times 10^{-3}$, 95% confidence interval)

	CP	CS	CSVP
End-to-end	(undetected)	0.49 ± 0.12	(undetected)
1st node	(undetected)	0.20 ± 0.078	(undetected)
2nd node	(undetected)	0.15 ± 0.078	(undetected)
3rd node	(undetected)	0.14 ± 0.047	(undetected)

better end-to-end cell loss performance shown in Table 5.3 at the cost of increased cell loss for Traffic Class 1.

When non-bursty traffic flow traverses a network where bursty traffic exists at the nodes on the path, it is necessary to protect the buffer space allocated to the non-bursty traffic throughout the path. Let the non-bursty traffic, Traffic Class 1, be the Reference Class. The bursty Traffic Classes 2 and 3 are used as the Background Classes at each node on the path. With the CS scheme, the cell loss ratio of the Reference Class significantly degrades at each node and a high end-to-end cell loss ratio is seen in Table 5.5. By the CSVP scheme, the buffer space allocated to the Reference Class is guaranteed and protected at all nodes and thus no cell loss is detected. The robustness of buffer allocation by the CSVP scheme protects non-bursty traffic from the violation by bursty traffic, which is one of the most significant properties of our resource allocation mechanism, since the traffic situation in ATM networks is expected to be heterogeneous.

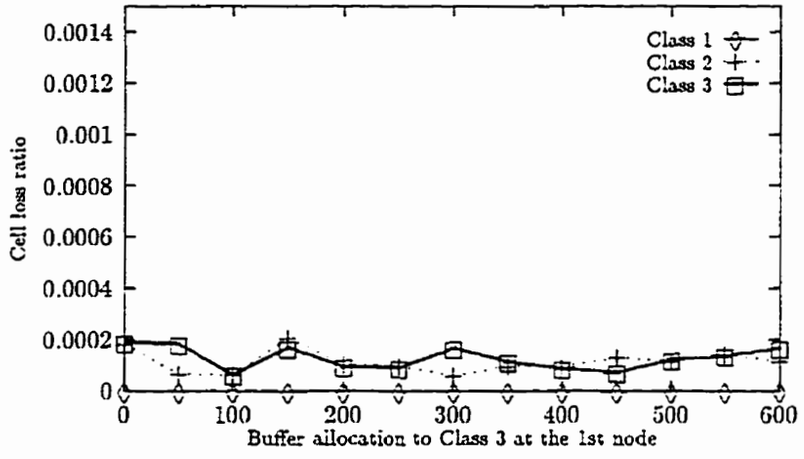

Figure 5.3: Virtual partitioning vs. cell loss ratio at the 2nd node

5.5 Virtual Partitioning

It was shown in Section 4.3.2 that the cell loss performance is sensitive to virtual partitioning, whereas the cell delay and the cell delay variation performances are insensitive. What is the effect of virtual partitioning on the end-to-end performance? If the end-to-end cell loss ratio is significantly affected, while the effect on the end-to-end cell delay and delay variation is minimal, the virtual partition will be a strong tool to adjust the end-to-end cell loss performance. The three-node network used in the previous section is used to examine the effect of virtual partitioning. Traffic Class 3 is chosen to be the Reference Class.

The effect of the virtual partitioning at the first node is already observed in Section 4.3.2. Let us examine if the virtual partitioning at the first node has any impact on the second node. Figure 5.3 shows the effect on the cell loss performance at the second node. The effect is too small to detect. The effect of the virtual

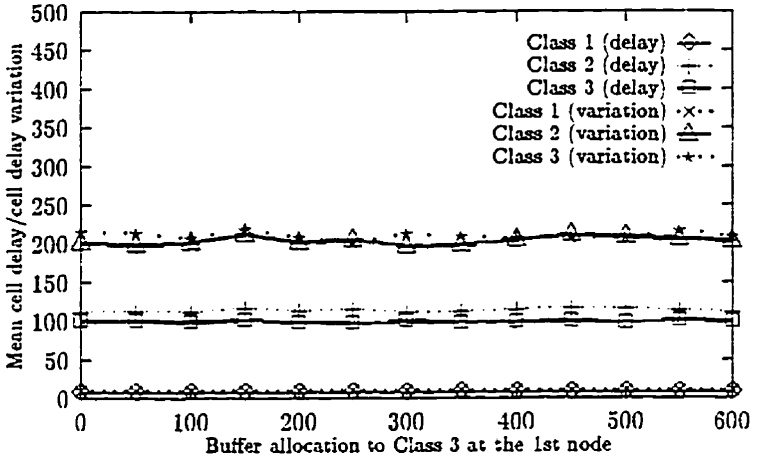

Figure 5.4: Virtual partitioning vs. mean cell delay and cell delay variation at the 2nd node

partitioning at the first node on the cell delay and the cell delay variation at the second node is shown in Figure 5.4. There appears to be no effect.

The impact of cell loss ratio at the first node is strong on the end-to-end cell loss performance, i.e., the first node is the bottleneck node. Thus, the effect of the virtual partitioning on the end-to-end cell loss performance is apparent in Figure 5.5. However, the cell delay and the cell delay variation performances do not seem to be affected (Figure 5.6). These results strongly support the virtual partition as one of the tools to adjust the cell loss service quality.

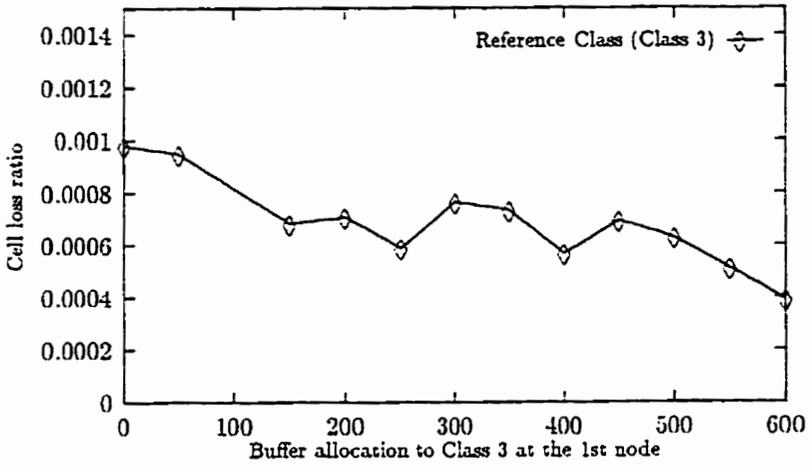

Figure 5.5: Virtual partitioning vs. end-to-end cell loss ratio

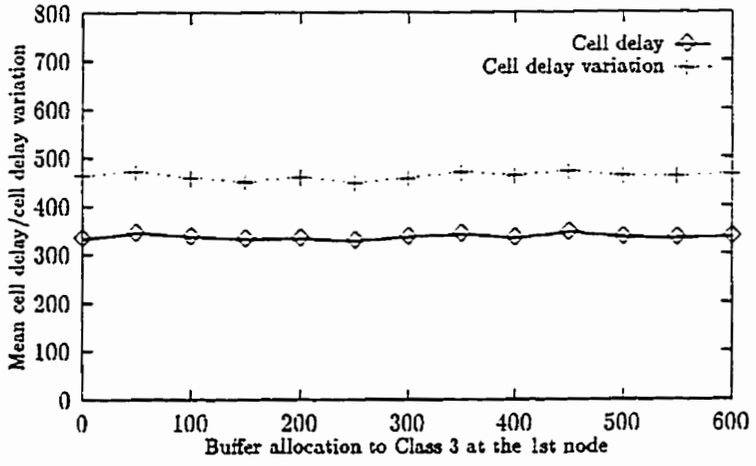

Figure 5.6: Virtual partitioning vs. mean end-to-end cell delay and cell delay variation

5.6 MPEG Video Data Stream

5.6.1 MPEG Video Data

The MPEG first-phase (MPEG-1) video compression [ISO94] is an international standard for video compression, primarily developed for the storage of video data [Gal91]. For actual data transmission over a network, where some losses of data may occur, the MPEG second-phase (MPEG-2) standard [ISO95] was developed and is expected to be adopted for the broadcasting of High Definition Television (HDTV) [Min95, ACD+95]. Public-domain software referred to as *MPEGTool* [UAH+94] was developed to create MPEG coded data and to obtain some statistics.

In order to transmit MPEG coded data, the data stream has to be segmented into ATM cells. The ATM cell segmentations of MPEG-1 has been studied, although MPEG-1 may not be designed for data transmission. The cell segmentation of MPEG coded video data can be performed for different coding layers of MPEG, frame layer, slice layer, and macroblock layer. Macroblock layer cell segmentation is discussed in [GH93]. The characteristics of cell streams generated by MPEG sources for different coding layers are examined in [PZ94]. The cell delay and cell delay variation performance using MPEG-2 coded video data is studied by simulation for tandem FIFO queues in [NLG96].

In this section, MPEG-1 coded video data are used as the test sources and the cell loss, cell delay and cell delay variation are examined. A tool referred to as *mpegcell* [Che96] is applied to create cell streams from the MPEG data; *mpegcell* uses the information obtained by MPEGTool. In *mpegcell*, frame layer cell segmentation

is adopted and no spacing between cell emissions is applied. The cell stream created by frame level cell segmentation can be very bursty if the cells belonging to the same frame are emitted to the network without spacing. Therefore, in practice, some spacing between cell emissions may be introduced to reduce burstiness, and/or slice level cell segmentation may be used. Here, however, the Work-conserving WRR-CSVP resource allocation scheme is tested using the highly bursty traffic of frame level cell segmentation without spacing. It should be noted that frame layer cell segmentation may require a maximum delay of 2 frame times for cell segmentation and assembly.

5.6.2 Performance Evaluation

The MPEG coded video data used in the emulation study are listed in Table 5.6. These are the data with sufficiently long duration and high intensity to be suitable for our emulator. We consider a traffic situation Case 5, shown in Table 5.7. Traffic Class 1 is the Traffic Class of the MPEG video sources. This Traffic Class is used as the Reference Class and traverses the network. The other Traffic Classes are the Background Classes: Traffic Class 2 is bursty traffic created by a superposition of IBP mini-sources; and Traffic Class 3 is non-bursty traffic created by a superposition of Bernoulli process mini-sources. This setting can be seen as a situation where a video channel shares the resources with data and voice channels. Each channel supports a number of sources.

A seven-node network is created. The resource allocation conditions are the same at all nodes and the resource dimensioning was performed empirically. At

Table 5.6: MPEG sources

Name	frames/s	Avg. rate (b/s)	Description
RedsNightmare	25	597776.45	A short animation movie with scene changes.
earth-cif	25	974899.86	An animation of the earth rotating. No scene change.
pca	30	551481.36	Scenes of aircrafts flying. Both animation and actual movie. Contains some scene changes.
canyon	30	237193.72	An animation of the scene of a canyon seen from a flying aircraft. No scene change.
zoom	30	945774.17	An animation zooming some geometrical patterns. No scene change.
btpintro	25	442061.71	An introduction of a movie with some scene changes.
jet	30	182210.99	Similar to 'canyon'. No scene change.
genesisp	24	444903.84	A part of one of the Startrek movies. Contains some scene changes.

Table 5.7: Traffic Classes of Case 5

	Source
Class 1	MPEG video sources
Class 2	Type 2 $\times 10$
Class 3	Type 1 $\times 10$

Table 5.8: The aggregate cell loss ratios ($\times 10^{-3}$)

	CP	CSVP
1st node	2.0 ± 0.83	0.18 ± 0.14
2nd node	0.32 ± 0.15	(undetected)
3rd node	0.38 ± 0.12	(undetected)
4th node	0.44 ± 0.18	(undetected)
5th node	0.53 ± 0.16	(undetected)
6th node	0.46 ± 0.16	(undetected)
7th node	0.36 ± 0.14	(undetected)

each node, the total buffer size is 550 cells and the allocation is: 300 cells to Traffic Class 1; and 125 cells to each of Traffic Classes 2 and 3, i.e., $B_1 = 300$ and $B_2 = B_3 = 125$. For the capacity allocation, 1/2 of the total link capacity is allocated to Traffic Class 1 and 1/4 to each of Traffic Classes 2 and 3, i.e., $S = 20$, and $S_1 = 10$ and $S_2 = S_3 = 5$.

Let us examine the buffer efficiency first. A network with the Work-conserving WRR-CP resource allocation mechanism is constructed for comparison purposes. The aggregate cell loss ratio shown in Table 5.8 indicates the buffer efficiency of the CSVP scheme. The aggregate cell loss ratio is much better with the CSVP scheme at all nodes. Therefore, the CSVP scheme is more buffer efficient than the CP scheme.

Table 5.9 shows the end-to-end cell loss ratio and the cell loss ratio of the bursty Reference Class at each node. The end-to-end cell loss ratio is smaller with the CSVP scheme than the CP scheme. This is mainly due to the cell loss performance at the first node, which has a significant impact on the end-to-end cell loss ratio. The buffer efficiency of the CSVP scheme contributes to obtain a lower end-to-end

Table 5.9: The end-to-end cell loss ratio and the cell loss ratios at each node of Traffic Class 1 ($\times 10^{-3}$, 95% confidence interval)

	CP	CS	CSVP
End-to-end	29.7 ± 13.1	2.3 ± 2.1	3.6 ± 2.6
1st node	29.7 ± 13.1	2.3 ± 2.1	3.6 ± 2.6
2nd node	(undetected)	(undetected)	(undetected)
3rd node	(undetected)	(undetected)	(undetected)
4th node	(undetected)	(undetected)	(undetected)
5th node	(undetected)	(undetected)	(undetected)
6th node	(undetected)	(undetected)	(undetected)
7th node	(undetected)	(undetected)	(undetected)

Table 5.10: The cell loss ratios of Traffic Class 2 ($\times 10^{-3}$, 95% confidence interval)

	CP	CS	CSVP
1st node	0.98 ± 0.50	0.070 ± 0.047	0.00033 ± 0.00075
2nd node	0.67 ± 0.32	(undetected)	(undetected)
3rd node	0.79 ± 0.24	(undetected)	(undetected)
4th node	0.92 ± 0.38	(undetected)	(undetected)
5th node	1.1 ± 0.34	(undetected)	(undetected)
6th node	0.98 ± 0.34	(undetected)	(undetected)
7th node	0.76 ± 0.30	(undetected)	(undetected)

cell loss ratio for bursty traffic. Another bursty class, Traffic Class 2 also benefits at all nodes and experiences a much lower cell loss (Table 5.10). Bursty Background traffic also benefits from the buffer efficiency of the CSVP scheme.

Is the non-bursty class, Traffic Class 3, protected? Table 5.11 shows the cell loss experienced by Traffic Class 3 at each node. At the first node, where the bursty MPEG coded video traffic enters the network, Traffic Class 1 is greedy for the buffer space. Thus, this bursty Reference Class and another bursty class, Traffic Class 2,

Table 5.11: The cell loss ratios of Traffic Class 3 ($\times 10^{-3}$, 95% confidence interval)

	CP	CS	CSVP
1st node	(undetected)	0.066 ± 0.048	(undetected)
2nd node	(undetected)	(undetected)	(undetected)
3rd node	(undetected)	(undetected)	(undetected)
4th node	(undetected)	(undetected)	(undetected)
5th node	(undetected)	(undetected)	(undetected)
6th node	(undetected)	(undetected)	(undetected)
7th node	(undetected)	(undetected)	(undetected)

cause unnecessary cell loss for the non-bursty traffic, Traffic Class 3, with the CS scheme. The undetected cell loss for Traffic Class 3 shown in Table 5.11 shows that the CSVP scheme guarantees a minimum buffer space. The robustness of buffer allocation of the CSVP scheme is effective.

From the second node to the seventh node, no cell loss is detected with any buffer allocation scheme. It may be the Work-conserving WRR scheduling scheme with Round Robin service that reduces the burstiness of the Reference Class at the first node and results in undetected cell loss at the downstream nodes. The emulation using Case 1 in the previous section shows the same tendency. Recall that Exhaustive service may increase the burstiness and cause a higher cell loss ratio at the downstream nodes. These observations suggest that scheduling schemes may have a significant impact on the cell loss performance at the downstream nodes.

Let us examine the cell delay and the cell delay variation now. The end-to-end cell delay and cell delay variation performances are shown in Table 5.12. Recall that frame layer cell segmentation requires a maximum delay of 2 frame times for cell segmentation and assembly. An interframe time is 1/30 seconds, which is

Table 5.12: The end-to-end cell delay and cell delay variation performances ($\times 10^2$, 95% confidence interval)

(cell time)	CP	CS	CSVP
Cell delay	1.8 ± 0.11	1.9 ± 0.13	1.9 ± 0.15
Cell delay variation	1.5 ± 0.12	1.9 ± 0.16	1.8 ± 0.17

approximately 12347 cell times. The largest burst is caused by the largest type of frame in the MPEG coding, intra-frame (I-frame), and the size of the burst is around 300 consecutive cells on average for our MPEG coded video sources (The largest burst can be over 500 cells). The end-to-end cell delay and delay variation observed in our emulation are much smaller than the cell segmentation and assembly delay, even if the size of the burst is taken in to account. Therefore, the end-to-end delay and delay variation observed here may be tolerable for some applications.

The Work-conserving WRR-CSVP resource allocation achieves a better aggregate cell loss ratio throughout the path of MPEG coded video traffic due to its efficient use of resources. The efficiency benefits bursty traffic to obtain a lower end-to-end cell loss performance. The resource allocation mechanism also protects non-bursty traffic at all nodes from potential high cell loss caused by sharing the resources with bursty MPEG video traffic without a regulation. This shows the robustness of resource allocation of the Work-conserving WRR-CSVP resource allocation mechanism. Our resource allocation mechanism is, thus, suitable for the situations where bursty traffic, such as MPEG video data, and non-bursty traffic coexist.

It should be noted that the largest portions of the end-to-end cell delay and cell

Table 5.13: The cell delay performance at each node ($\times 10^2$, 95% confidence interval)

(cell time)	CP	CS	CSVP
1st node	1.5 ± 0.10	1.6 ± 0.12	1.7 ± 0.14
2nd node	0.98 ± 0.012	0.11 ± 0.018	0.10 ± 0.012
3rd node	0.043 ± 0.0035	0.044 ± 0.0051	0.041 ± 0.0037
4th node	0.028 ± 0.0015	0.030 ± 0.0034	0.029 ± 0.0021
5th node	0.023 ± 0.00082	0.024 ± 0.0012	0.023 ± 0.0010
6th node	0.022 ± 0.00041	0.022 ± 0.00057	0.021 ± 0.00050
7th node	0.020 ± 0.00022	0.021 ± 0.00037	0.020 ± 0.00026

Table 5.14: The cell delay variation performance at each node ($\times 10^2$, 95% confidence interval)

(cell time)	CP	CS	CSVP
1st node	1.5 ± 0.11	1.8 ± 0.14	1.7 ± 0.17
2nd node	0.17 ± 0.032	0.21 ± 0.051	0.19 ± 0.035
3rd node	0.066 ± 0.015	0.067 ± 0.015	0.061 ± 0.0097
4th node	0.029 ± 0.0041	0.041 ± 0.016	0.035 ± 0.0082
5th node	0.018 ± 0.0029	0.022 ± 0.0046	0.021 ± 0.0033
6th node	0.015 ± 0.00090	0.015 ± 0.0018	0.014 ± 0.0012
7th node	0.012 ± 0.00034	0.013 ± 0.00074	0.012 ± 0.00058

delay variation are incurred at the first node (Tables 5.13 and 5.14). The impact of the first node is also significant for cell loss performance (Table 5.9). These results suggest that the control at such a bottleneck node is significant to achieve the required QoS, such as cell loss, cell delay, and cell delay variation.

5.7 Concluding Remarks

The Work-conserving WRR-CSVP resource allocation mechanism has been investigated from the network viewpoint. An end-to-end network is modeled by a tandem connection of single-server queues and an emulator of such a network partition is developed by connecting a number of generic node emulators in tandem.

An attempt is made to demonstrate bursts in the output traffic caused by Exhaustive service. Although the result is inconclusive due to the limited accuracy of our emulator, it is recommended to adopt Round Robin service to avoid this potential problem.

The efficient use of resources and the robustness of resource allocation inherent in the Work-conserving WRR-CSVP resource allocation mechanism are shown to have significant impacts on the end-to-end cell loss performance. These aspects of our resource allocation mechanism are significant, especially in heterogeneous traffic situations expected in ATM networks.

Another advantage of the Work-conserving WRR-CSVP resource allocation mechanism is the effect of virtual partitioning. It is demonstrated that the end-to-end cell loss performance is sensitive to virtual partitioning at the bottleneck node while the end-to-end cell delay and the end-to-end cell delay variation are insensitive to it. This suggests that the virtual partition can be a tool to tune the end-to-end cell loss performance without affecting the end-to-end cell delay and the end-to-end cell delay variation.

The Work-conserving WRR-CSVP resource allocation scheme is applied to handle MPEG coded video data streams. It is shown that our resource allocation

scheme is effective when highly bursty traffic such as MPEG coded video data shares the network resources with other bursty and/or non-bursty traffic.

Chapter 6

Implementation Strategies And Feasibility

6.1 Introduction

The effectiveness of the Work-conserving WRR-CSVP resource allocation mechanism presented in Chapters 2 and 3 has been demonstrated by emulation results in Chapters 4 and 5. In Chapter 3, the detailed algorithms to implement the Work-conserving WRR-CSVP mechanism as an application software is discussed. Here, the algorithms are divided into several processing entities to achieve parallel processing.

As discussed in Section 3.2, the implementation of the buffer space is an important issue. A suitable memory structure needs to be investigated. The design of the random selection method is another question to be examined.

In this chapter, we provide design and implementation strategies for the Work-

conserving WRR-CSVP mechanism. The feasibility of the proposed design is examined by assessing the size of the memory space and the computational costs of the operations. The design specification is provided in Appendix C: the precise structures of the memory space and the detailed operations of the processing entities are described in the specification. In the following discussion, the focus is on the cell admission and the service mechanisms.

6.2 General Structure

The operations of the Work-conserving WRR-CSVP resource allocation mechanism are divided into five processing entities to achieve parallel processing. We refer to a processing entity as a *state machine* (SM). The five SMs are: the Pre-Buffering SM (PB-SM); the Cell Buffering SM (CB-SM); the Buffer Allocation SM (BA-SM); the Service Scheduling SM (SS-SM); and the Cell Transmission SM (CX-SM).

The actual buffer space is realized by the Buffer Space (BS) memory segment. The Pre-Buffer (PB) segment and the Transmission Buffer (XB) segment are added to cope with the speed mismatch among the SMs and the switching fabric. The VCs supported at an output port are managed with the data kept in the VC Table (VCT) memory segment.

The general structure of the system is illustrated in Figure 6.1, with the functionalities described in the following section.

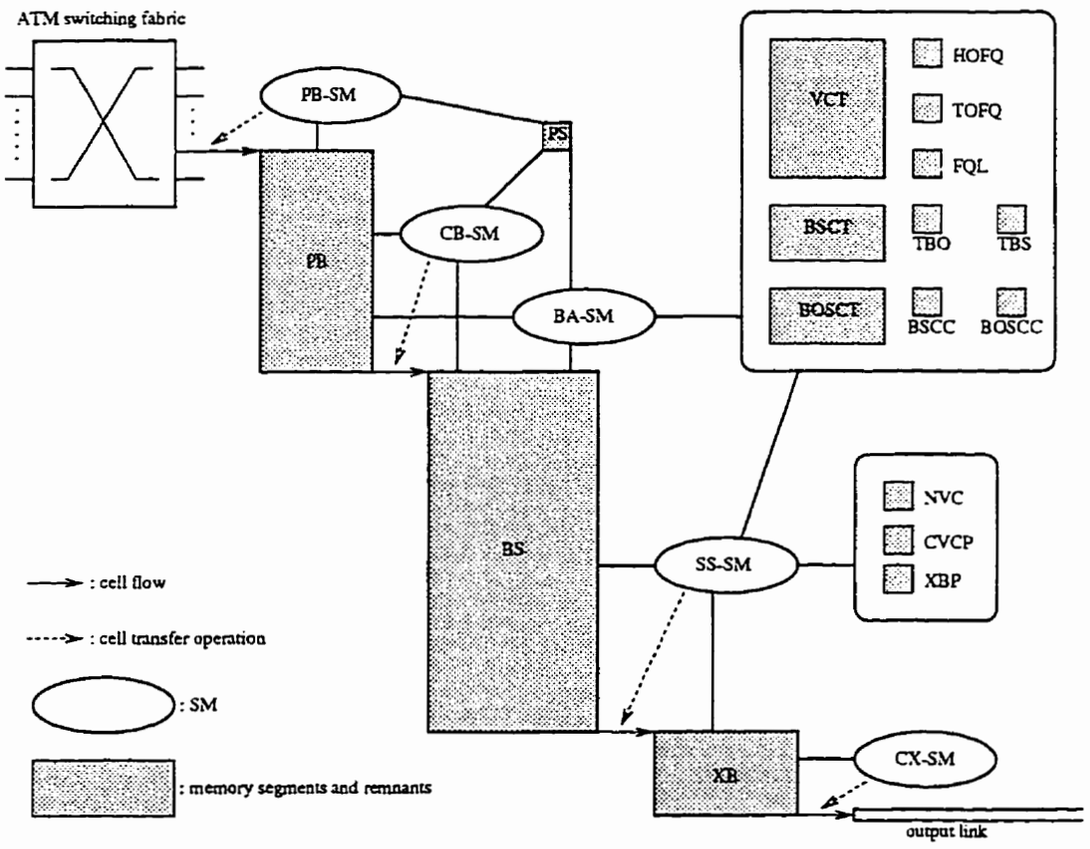

Figure 6.1: General structure of the Work-conserving WRR-CSVP resource allocation scheme

6.3 Implementation Strategies

6.3.1 Parallel Processing

In the following, the functionalities of the SMs and the PB, BS, and XB memory segments are described. The descriptions are given by following the flow of cells indicated by arrows in Figure 6.1.

The speed of the switching fabric is expected to be very high. In order to allow sufficient time for the BA-SM to perform complex cell admission, the PB segment is placed between the switching fabric and the actual buffer space, the BS segment. The PB-SM transfers the cells from the switching fabric and temporarily stores them in the PB segment. Such an architecture can also be seen in [THP94].

The BA-SM performs the cell admission according to the CSVP buffer allocation scheme. If the cell can be admitted, the BA-SM finds an empty space in the BS or creates an empty space with a pushout operation. The random method is adopted for the traffic class selection of the pushout operation. In order to avoid the load of time consuming cell copy operation for the BA-SM, the CB-SM is added to copy the admitted cells in the PB to the BS.

The SS-SM selects a cell to be transmitted according to the Work-conserving WRR scheduling scheme. The random method is adopted for the traffic class selection of the time slot reassignment operation. In order to avoid the complex memory contention problem between the SS-SM and the BA-SM, they are designed to work in sequence, i.e., the SS-SM activates the BA-SM upon its completion of the operation in a time slot. However, this may require very high speed operations

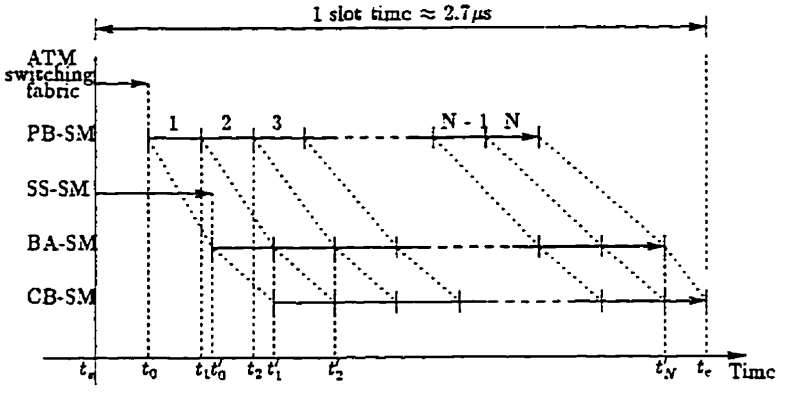

Figure 6.2: Timing structure of the SMs

of the SS-SM and the BA-SM.

It takes one full cell time to transmit a cell to the output link. Therefore, a transmission buffer, the XB segment, is placed between the BS segment and the output link. The scheduling entity, the SS-SM, transfers a cell to the XB, and the CX-SM transmits a cell in the XB to the output link. The CX-SM operates independent of the other SMs and the switching fabric without memory contention.

The switching fabric and the other four SMs, the PB-SM, the CB-SM, the BA-SM, and the SS-SM, have timing dependencies. Let N be the number of the output ports of the switching fabric, i.e., at most N cells arrive in a time slot. The timing structure of the SMs to achieve parallel processing is illustrated in Figure 6.2. At t_0, the ATM switching fabric starts emitting the first cell. The PB-SM starts as soon as the switching fabric starts emitting the first cell and completes the transfer of the first cell to the PB at t_1 and the nth cell at t_n. The SS-SM starts its job in the time slot at t_s, completes it at t'_0, and activates the BA-SM. The BA-SM cannot start its job on the nth cell until the nth cell is transferred from the switching fabric

by the PB-SM at t_n. Thus, the BA-SM starts processing the 1st cell at $t'_0 \geq t_1$ and the nth cell at $t'_{n-1} \geq t_n$, and completes the job on the last cell at t'_N. The CB-SM cannot start its job on the nth cell until the BA-SM completes its job on the nth cell at t'_n. However, it should not be difficult for the CB-SM to complete its job on the nth cell by t'_{n+1} since the operation of the CB-SM between t'_n and t'_{n+1} is essentially copying of a cell and thus deemed simpler than the operation of the BA-SM on a cell between t'_n and t'_{n+1}. Therefore, the only overhead of the CB-SM is the processing of the last cell, the Nth cell, i.e., $t_e - t'_N$. Note that 1 slot time is approximately $2.7\mu s$.

6.3.2 Buffer Space Management

As discussed in Section 3.2, the full sharing of buffer space and the FIFO/LIFO operations of the Work-conserving WRR-CSVP mechanism can be easily implemented by doubly linked lists. A FIFO VC queue by a logical linked list is implemented in [KSC91]. Hardware implementation of a forward chain can also be seen in the memory shared switch architecture in [EKO+93]. A more complex memory management using a doubly linked list is achieved by a Memory Controller (MEC) chip developed by a group from Sumitomo Electric Industries, Ltd. for their Fiber Distributed Data Interface (FDDI) LAN products [KTT+89]. Other advantages of a doubly linked list include memory efficiency, flexibility, and easy recovery of a broken chain. Also, an advantage is that the empty space can be maintained easily. The doubly linked list of the empty space in the buffer is referred to as the Free Queue in this chapter and Appendix C.

Another possible architecture is to utilize address queues. Such an architecture is used in the ATM switch designs in [Cha91a, CU95, LS96a]. This may be a simpler architecture but it is not suitable for the CSVP buffer allocation scheme for the following reason. The size of address is 4 bytes and the size of a cell is 53 bytes. Let B be the maximum number of cells the buffer can contain, i.e., the size of buffer space. A VC may occupy the entire buffer space so the size of an address queue is $4 \times B$ bytes while the total buffer size is $53 \times B$ bytes. Since each VC requires one address queue and the number of VCs supported is expected to be large, the memory space necessary for the address queues can easily surpass the size of the memory space for the actual buffer space. However, the memory space for the address queues is sparsely used because only a maximum of B cells are permitted to occupy the actual buffer space. Therefore, this architecture is memory inefficient and the buffer efficiency of the CSVP scheme cannot be effective.

6.3.3 Random Selection Method

In order to implement the random selection method for the pushout and the time slot reassignment operations, it is necessary to identify the eligible VCs. The Buffer Subscribing Connection Table (BSCT) and the Buffer Over-Subscribing Connection Table (BOSCT), are designed to contain the list of eligible VCs in the proposed design in Appendix C. The random selection can be made by selecting a VC from these tables at random using a pseudo-random number generator. The tables have to be updated whenever the eligibilities of VCs change. The procedures to maintain the BSCT and the BOSCT are also included in the design specification presented

in Appendix C.

6.4 Discussion on Feasibility

6.4.1 Design Parameters

The following design parameters are used. Let the size of the ATM switching fabric be $N \times N$, i.e., the ATM switching fabric possesses N input ports and N output ports. Let K be the maximum number of VCs that this output port can support. The value of K may be larger or smaller than the number of input ports of the switching fabric, N. Let B be the size of the buffer space in number of cells, i.e., a maximum of B cells can be stored in the output port buffer.

6.4.2 Necessary Amount of Memory Space

Let us first examine the size of memory needed in this design. The size of the memory segments and remnants are as follows. (Bits have been rounded up to bytes.)

- The PB segment: $58 \times N$ bytes.
- The BS segment: $62 \times B$ bytes.
- The XB segment: 54×3 bytes.
- The VCT segment: $20 \times K$ bytes.
- The BSCT segment: $4 \times K$ bytes.

- The BOSCT segment: $4 \times (K-1)$ bytes.
- The memory remnants: 31 bytes.

In total, $58 \times N + 62 \times B + 28 \times K + 191$ bytes of memory are necessary. Assume that we have a 1024×1024 ATM switching fabric, i.e., $N = 1024$, and that one output port supports a maximum of 1024 VCs, i.e., $K = 1024$. Since the buffer space is shared by 1024 VCs, each VC may not need a large space. Let us assume that we provide 10 cells of buffer space for each VC at an output port. This gives a total buffer size of $B = 10240$, i.e., at most 10240 cells can be buffered, which is reasonably large for an output port, considering that the average load of the input traffic is lower than the capacity of the output link. Under these assumptions, the amount of necessary memory space is 723135 bytes, approximately 723K bytes. In order to have an ATM switch with the Work-conserving WRR-CSVP resource allocation mechanism at its output ports, 723K bytes of memory is necessary at each output port and 723M bytes for the entire switch. The current technology of dual-port random access memory (RAM) can meet this requirement to enable concurrent access, which may be necessary for the implementation of the PB, the BS, and the XS segments. The actual memory space for storing cells in the buffer is $53 \times B$ bytes, and thus, the overhead is $58 \times N + 9 \times B + 28 \times K + 191$ bytes. In our example, the overhead is only approximately 193K bytes, which is approximately 27% of the entire memory space. This ratio approachs 15% when the buffer size B becomes larger.

6.4.3 Computational Costs

The operations of the PB-SM, the CB-SM, and the CX-SM are very simple and straightforward (See, Figures C.8, C.9, and C.10 in Appendix C) and the algorithmic state machines (ASM) for these can easily be developed. Considering the timing structure in Figure 6.2, the speed requirements for these SMs may not be too difficult to meet. Thus, we concentrate our discussion on the SS-SM and the BA-SM.

The ideal situation concerning timing is that the SS-SM completes its job by t_1, i.e., $t'_0 = t_1$. This allows the BA-SM a maximum time to operate since it can only start its job at t_1 at the earliest.

Is it possible to implementation our Work-conserving WRR scheduling scheme in a very large-scale integration (VLSI) chip? Let us first consider the implementation of the operation of the WRR scheduling scheme. According to our design in Appendix C, the procedure of the WRR scheduling scheme contains two loops: the renewal of a service amount allocated to each VC, RC_k, $k = 1, 2, \cdots, K$, requires at most K iterations of restoration of the original values upon the re-initialization of each service cycle (See Figure C.12); and Round Robin service requires at most K comparisons to find next VC (See Figure C.14). A complementary metal-oxide semiconductor (CMOS) chip of a 4×4 shared memory switch is designed in [KSC91]. This chip handles 256 VCs at each output port. As described in Section 3.4.3, the WRR scheme by Katevenis' group allows various patterns of service including Exhaustive and Round Robin services. Therefore, it is possible to adopt their architecture to implement our WRR scheme, although the mechanism to make it

work-conserving is different.

The time slot reassignment operation added to the WRR scheduling scheme may increase the difficulties of VLSI implementation. Reassignment of time slot and renewal of the BSCT and the BOSCT are the key additional functions to the WRR scheme. Although reassignment of a time slot is a straightforward operation (Figure C.15), the operation of the SS-SM also includes the removals of an entry from BSCT and BOSCT, which require at most K and $K-1$ comparisons, respectively (See Section C.3.7). Computational cost increases due to these loops as the number of supported VCs, K, increases.

The SS-SM also performs copying of a cell, which should be easily matched with the similar operation performed by the PB between t_1 and t_2. Therefore, if the operations other than copying can be completed by t_1, i.e., this part of the SS-SM matches the speed of the switching fabric, the SS-SM becomes ideal for our timing situation.

The BS-SM is the most complex entity and possibly the bottleneck for fast operation, and thus is the most critical part of implementation. The ideal case is that the BA-SM can process a cell using as much as or less time than the PB-SM to process the subsequent cell, i.e., $t'_n - t'_{n-1} \leq t_{n+1} - t_n$. This will result in $t'_n = t_{n+1}$. Otherwise, the operating time difference of the two SMs will accumulate.

The CS scheme is already implemented by Katevenis' group [KSC91]. Their design includes VPI-VCI matching to identify the VC in the VCT, which requires at most K comparisons. It also maintains logical linked lists of VC queues. What are the additional functions to make a CS scheme to be a CVSP scheme? They

are the branch in the flowchart in Figure C.18 where '$BO_k \geq BT_k$' is examined for pushout, and the additions of an entry to the BSCT and the BOSCT. Note that the operation of cell admission by pushout, the additional branch in Figure C.18, may seem straightforward in Figure C.20, but it includes the removals of an entry from BSCT and BOSCT, which require at most K and $K-1$ comparisons, respectively. Another addition to the CS scheme by Katevenis' group is the maintenance of doubly linked lists of VC queues, which requires twice as many operations as that of logical linked lists.

Let us summarize the operational complexity relative to the design parameters. According to the flowcharts in Appendix C, the operation of the SS-SM contains at most $4K$ loops, and the operation of the BS-SM requires at most $3K-1$ loops to process a cell, and in total, at most $(3K-1) \times N$ loops. Therefore, switch size, N, and the number of supported VCs, K, are the deciding factors of computational cost. Considering the existent implementation such as the switch by Katevenis' group, it may be possible to implement the Work-conserving WRR-CSVP resource allocation mechanism in a 4×4 switch, where 256 VCs are supported at each output port, i.e., $N = 4$ and $K = 256$. When switch size N increases, however, we may have to wait for the progress of hardware technology. It should be noted that the use of the reduced instruction set computer (RISC) microprocessor to implement the SS-SM and the BS-SM is a possibility. In such a case, the design specification provided in Appendix C is immediately applicable.

Chapter 7

Conclusions and Further Study

7.1 Overview

One of the significant challenges in ATM networks is to provide QoS guarantees and to achieve efficient use of resources to obtain better cell loss performance. With a non-blocking output-buffering ATM switch, the resources at an output port of an ATM switch, a combination of buffer space and link capacity, have to be carefully managed for these purposes. Since the traffic is expected to be bursty, a complete sharing approach can realize considerable efficiency. However, it is necessary to have some robustness of resource allocation to guarantee a minimum amount of resources. The CSVP resource allocation strategy introduced by Wu and Mark [WM95] provides such efficiency and robustness of resource allocation. The CSVP strategy was applied to buffer allocation for the case of two traffic flows in [WM95] and the cell loss performance was studied using fluid-flow approximation. The analysis was limited to the case of two traffic flows due to the analytical

intractability of the queueing model.

It was intended in this work to propose a resource allocation mechanism that manages both buffer space and capacity allocation, and is feasible at an output port of an ATM switch. The mechanism has to provide efficient use of resources to obtain better cell loss performance and robust allocation of resources to guarantee a minimum amount of resources to each traffic flow to protect well-behaved traffic. To this end, the CSVP strategy was applied also to capacity allocation to form the Work-conserving WRR scheduling scheme, and was combined with the CSVP buffer allocation scheme to constitute the Work-conserving WRR-CSVP resource allocation mechanism.

In Chapter 2, the Work-conserving WRR-CSVP resource allocation mechanism was defined and a queueing model was established. By analysis, the cell loss ratio of the WRR-CSVP mechanism was shown to be smaller or equal to that of the WRR-CP mechanism (Theorem 1). A non-increasing relation between cell loss ratio and virtually allocated buffer space in the Work-conserving WRR-CSVP mechanism for the case of two traffic flows is also derived (Theorem 2).

The Work-conserving WRR-CSVP resource allocation mechanism was implemented as an application program on a workstation to obtain insights for actual implementation. Then, peripheral modules were developed to construct an emulator in order to examine the characteristics of our resource allocation mechanism. In Chapter 3, the detailed algorithms of the Work-conserving WRR-CSVP resource allocation mechanism were described. The algorithms included the handling of more than two traffic flows. The implementation of the buffer space was also discussed.

Three design issues were raised by proposing some options: the mechanics of the WRR scheduling scheme, Exhaustive or Round Robin services; the operations of the buffer allocation scheme to select a traffic flow to be pushed out when more than two traffic flows are supported; and the operation of the scheduling scheme to select a traffic flow to receive a reassigned time slot. It was necessary to examine these options to determine the best choice.

In Chapter 4, an emulation study was conducted to investigate the advantages of the Work-conserving WRR-CSVP resource allocation mechanism. An attempt was made to demonstrate the efficiency of capacity allocation by the Work-conserving WRR scheduling scheme, and the buffer efficiency and the robustness of buffer allocation of the CSVP buffer allocation scheme by emulation. Also, the effect of virtual partitioning on cell loss and cell delay was investigated. The cell loss performance was sensitive to virtual partitioning but the cell delay and the cell delay variations were insensitive. The design issues pointed out in Chapter 3 were examined by emulation. The random selection method for pushout time slot reassignment operation was shown to perform sufficiently.

The Work-conserving WRR-CSVP resource allocation mechanism was examined on an end-to-end basis in Chapter 5. In theory, Exhaustive service may create bursts in the output traffic, but this phenomenon could not be demonstrated. However, we still recommend Round Robin services to avoid potential bursts. The buffer efficiency and the robustness of buffer allocation of the Work-conserving WRR-CSVP scheduling scheme were demonstrated by the end-to-end performance. The effects of virtual partitioning on the downstream nodes and the end-to-end performance

were also examined. Virtual partitioning has a noticeable impact on the end-to-end cell loss but the effect on the end-to-end cell delay and cell delay variation is minimal. The efficiency of the Work-conserving WRR-CSVP mechanism was shown to be significant in handling very bursty traffic such as MPEG coded video signals, while the robustness of resource allocation was also shown to protect well-behaved traffic from such bursty traffic.

In Chapter 6, a hardware design strategy was discussed. The Work-conserving WRR-CSVP resource allocation mechanism is divided into the five parallel processing entities with three main memory segments. The memory size and the computational cost were examined and the feasibility was discussed. The detailed algorithms of the processing entities and the structure of the memory space are provided in Appendix C. Computational cost may still be a problem in implementing a large switch but the proposed mechanism may be feasible for small switches such as 4×4.

7.2 Contributions

The Work-conserving WRR-CSVP resource allocation mechanism is proposed to allocate resources, a combination of buffer space and link capacity, at an output port of an ATM switch. By this mechanism, the resources are utilized efficiently and a minimum amount of resources is guaranteed to each traffic flow. Our mechanism is effective especially for heterogeneous traffic situations where bursty and non-bursty traffic flows coexist. Another advantage of the Work-conserving WRR-CSVP mechanism is that the cell loss performance is sensitive to virtual partitioning of the CSVP buffer allocation scheme but the cell delay and cell delay variation

performances are not. This is also true in the case of end-to-end performance. Therefore, the virtual partition may be a useful tool to adjust cell loss performance only. These advantages indicate the importance of the proposed resource allocation mechanism to handle heterogeneous traffic situations of ATM networks with bursty traffic flows.

The Work-conserving WRR-CSVP resource allocation mechanism was implemented as an application software on a workstation. Some implementation insights were obtained to provide design guidelines. This led to the hardware design specified in Appendix C. Our hardware design offers a practical possibility. Once the speed matching difficulty is overcome by the advanced hardware technology, the proposed resource allocation mechanism has definite advantages.

7.3 Suggestions for Further Study

The Work-conserving WRR scheduling scheme has some disadvantages. The lack of granularity of capacity allocation may become a problem. The unpredictability of the characteristics of the output traffic may incur some problems. What constitutes the most suitable scheduling scheme for ATM switches remains an issue.

Another issue of discussion is the level of control. We considered a VC based control. However, the Work-conserving WRR-CSVP resource allocation mechanism may be more suitable for a coarser level of control. The VCs of similar traffic characteristics can be bundled to form a class of traffic and the Work-conserving WRR-CSVP resource allocation mechanism can be applied to these traffic classes. This will reduce the complexity drastically, and the actual implementation may not

be too difficult. Note that the Traffic Classes used in our emulation are actually superpositions of mini-sources. Thus, they can be seen as a traffic class created by bundling VCs.

An empirical resource dimensioning is used in the emulation work. Resource dimensioning, especially buffer dimensioning, is a difficult yet important issue for further study.

Appendix A

Analysis Of Virtual Partitioning

A.1 Introduction

The aggregate cell loss ratio of the Work-conserving WRR-CSVP resource allocation mechanism is constant regardless of the position of the virtual partition (Proposition 1 in Chapter 2). However, it affects cell loss performance of traffic flows (See Section 4.3.2). It is shown in this appendix that a non-increasing relation between the virtually allocated buffer space and the resultant cell loss ratio exists for the case of two traffic flows: a larger buffer space virtually allocated to a traffic flow results in a smaller or equal cell loss ratio of the traffic flow. The following arguments take advantage of the simplicity of the pushout and the time slot reassingment operations when there are only two traffic flows.

Two traffic flows, Traffic Classes 1 and 2, are multiplexed, i.e., $K = 2$. Let B be the total buffer size. We compare two different virtual partitionings, Systems B and $\hat{B}$. The first virtual partitioning, System B, allocates B_1 and B_2 buffer spaces

to Traffic Classes 1 and 2 respectively. The second virtual partitioning, System $\hat{B}$, allocates $\hat{B}_1$ and $\hat{B}_2$ buffer spaces to Traffic Classes 1 and 2 respectively. Note that $B_1+B_2 = \hat{B}_1+\hat{B}_2 = B$. Let $x_k(n)$ and $\hat{x}_k(n)$, $k = 1, 2$, denote the buffer occupancies of Traffic Class k at the nth time slot in Systems B and $\hat{B}$ respectively. Let r_k and $\hat{r}_k$ denote the cell loss ratios of Traffic Class k in Systems B and $\hat{B}$ respectively.

We first examine the effect of the service scheduling in a time slot. Then, the effect of the cell admission to the buffer occupancy in a time slot is examined. These observations lead to the non-increasing relation, i.e., if $\hat{B}_k > B_k$, then $\hat{r}_k \leq r_k$.

A.2 Buffer Occupancy After Service Scheduling

Let us first examine the effect of the service scheduling on the buffer occupancies. Recall the timing structure in Section 2.3.1. In the nth time slot, the service scheduling decision according to the Work-conserving WRR scheduling scheme is made at $n\triangle - \epsilon$ and the cell is transferred to the transmission buffer immediately. Thus, a cell space is created in the buffer space by $n\triangle - \delta$. The scheduling decision is made at $n\triangle - \epsilon$, where the buffer occupancy is equal to that of the $(n-1)$th time slot. Recall that the following relation holds since the Work-conserving WRR-CSVP resource allocation mechanism is a WCD [CR87]:

$$x(n) = x_1(n) + x_2(n) = \hat{x}_1(n) + \hat{x}_2(n), \tag{A.1}$$

at any slot time n. Therefore,

Remark 2

$$x_1(n\triangle - \epsilon) + x_2(n\triangle - \epsilon) = \hat{x}_1(n\triangle - \epsilon) + \hat{x}_2(n\triangle - \epsilon) = x(n-1), \tag{A.2}$$

at any slot time n.

Let $u_k(n)$ and $\hat{u}_k(n)$, $k = 1, 2$, be the indicator functions of Systems B and $\hat{B}$, respectively. The buffer occupancy and the indicator function of each Traffic Class satisfy the following relations:

Lemma 1 *When only two Traffic Classes, Traffic Classes 1 and 2, are supported by the Work-conserving WRR-CSVP resource allocation mechanism, i.e., $K = 2$, the indicator functions of Systems B and $\hat{B}$ in the n^{th} time slot satisfy one of the following 3 cases:*

1. *If*

$$\hat{x}_k(n\triangle - \epsilon) = x_k(n\triangle - \epsilon), \tag{A.3}$$

then

$$\hat{u}_k(n) = u_k(n). \tag{A.4}$$

2. *If*

$$\hat{x}_k(n\triangle - \epsilon) < x_k(n\triangle - \epsilon), \tag{A.5}$$

then

$$\hat{u}_k(n) \leq u_k(n). \tag{A.6}$$

3. If

$$\hat{x}_k(n\triangle - \epsilon) > x_k(n\triangle - \epsilon), \tag{A.7}$$

then

$$\hat{u}_k(n) \geq u_k(n). \tag{A.8}$$

Proof: In the first case, the buffer occupancy at the service scheduling decision making is identical in both systems. Therefore, we have identical values for the indicator functions.

In the second case, the only situation that cause $\hat{u}_k(n) > u_k(n)$, i.e., Traffic Class k receives the service in System $\hat{B}$ while the other Traffic Class receives the service in System B, is when $x_k(n\triangle - \epsilon) = 0$. However, this contradicts with Inequality (A.5). Therefore, Inequality (A.6) holds.

The third case can be easily derived by replacing x_k with $\hat{x}_k$ and $\hat{x}_k$ with x_k in the argument of the second case.

■

Since the indicator functions, $u_k(n)$ and $\hat{u}_k(n)$, $k = 1, 2$, take the value of 0 or 1, the following relations can be easily derived from Lemma 1.

Corollary 1 *1. If*

$$\hat{x}_k(n\triangle - \epsilon) = x_k(n\triangle - \epsilon), \tag{A.9}$$

then

$$\hat{x}_k(n\triangle - \delta) = x_k(n\triangle - \delta). \tag{A.10}$$

2. If

$$\hat{x}_k(n\triangle - \epsilon) < x_k(n\triangle - \epsilon), \tag{A.11}$$

then

$$\hat{x}_k(n\triangle - \epsilon) \leq x_k(n\triangle - \epsilon) \tag{A.12}$$

and

$$x_k(n\triangle - \delta) - \hat{x}_k(n\triangle - \delta) \leq x_k(n\triangle - \epsilon) - \hat{x}_k(n\triangle - \epsilon). \tag{A.13}$$

3. If

$$\hat{x}_k(n\triangle - \epsilon) > x_k(n\triangle - \epsilon), \tag{A.14}$$

then

$$\hat{x}_k(n\triangle - \delta) \geq x_k(n\triangle - \delta) \tag{A.15}$$

and

$$\hat{x}_k(n\triangle - \delta) - x_k(n\triangle - \delta) \leq \hat{x}_k(n\triangle - \epsilon) - x_k(n\triangle - \epsilon). \tag{A.16}$$

Note that due to Remark 2, the following equation holds:

Remark 3

$$\begin{aligned} x_1(n\triangle - \delta) + x_2(n\triangle - \delta) &= \hat{x}_1(n\triangle - \delta) + \hat{x}_2(n\triangle - \delta) \\ &= [x_1(n\triangle - \epsilon) + x_2(n\triangle - \epsilon) - 1]^+, \end{aligned} \tag{A.17}$$

at any time slot n.

A.3 Buffer Occupancy After Cell Admission

Let us now examine the effect of the cell admissions to the buffer occupancies. Recall that $a_k(n)$, $k = 1, 2$, denotes the number of arrivals belonging to Traffic Class k in the nth time slot and that $a(n) = a_1(n) + a_2(n)$. The following properties hold for the buffer occupancies.

Lemma 2 *When only two Traffic Classes, Traffic Classes 1 and 2, are supported by the Work-conserving WRR-CSVP resource allocation mechanism, and the buffer*

space is allocated to satisfy

$$\hat{B}_k > B_k, \tag{A.18}$$

given identical input traffic, if

$$\hat{x}_k(n\triangle - \delta) \geq x_k(n\triangle - \delta) \tag{A.19}$$

and

$$\hat{x}_k(n\triangle - \delta) - x_k(n\triangle - \delta) \leq \hat{B}_k - B_k, \tag{A.20}$$

then

$$\hat{x}_k(n\triangle + \epsilon) \geq x_k(n\triangle + \epsilon) \tag{A.21}$$

and

$$\hat{x}_k(n\triangle + \epsilon) - x_k(n\triangle + \epsilon) \leq \hat{B}_k - B_k, \tag{A.22}$$

for any time slot n.

Proof: We assume that $k = 1$ without loss of generality. The possible situations are the following 3 cases:

1. $x_1(n\triangle - \delta) \geq B_1$ and $\hat{x}_1(n\triangle - \delta) \geq \hat{B}_1$,
2. $x_1(n\triangle - \delta) \geq B_1$ and $\hat{x}_1(n\triangle - \delta) < \hat{B}_1$, and

3. $x_1(n\triangle - \delta) < B_1$ and $\hat{x}_1(n\triangle - \delta) < \hat{B}_1$,

and the case where $x_1(n\triangle - \delta) < B_1$ and $\hat{x}_1(n\triangle - \delta) \geq \hat{B}_1$ does not occur due to Inequalities (A.19) and (A.20). In the following, each case is examined and the relations are shown to hold.

If there is no cell blocking or pushout, i.e.,

$$a(n) = a_1(n) + a_2(n) \leq B - x(n\triangle - \delta), \tag{A.23}$$

then,

$$\begin{cases} x_1(n\triangle + \epsilon) &= x_1(n\triangle - \delta) + a_1(n), \qquad \text{and} \\ x_2(n\triangle + \epsilon) &= x_2(n\triangle - \delta) + a_2(n), \end{cases} \tag{A.24}$$

and

$$\begin{cases} \hat{x}_1(n\triangle + \epsilon) &= \hat{x}_1(n\triangle - \delta) + a_1(n), \qquad \text{and} \\ \hat{x}_2(n\triangle + \epsilon) &= \hat{x}_2(n\triangle - \delta) + a_1(n). \end{cases} \tag{A.25}$$

Therefore, obviously Inequalities A.21 and A.22 hold. Hence, we only consider the case where the cell blockings and/or pushouts occurs, i.e.,

$$a(n) = a_1(n) + a_2(n) > B - x(n\triangle - \delta), \tag{A.26}$$

where from Remark 3,

$$x(n\triangle - \delta) \leq B - 1. \tag{A.27}$$

Note that the buffer is full after the cell admissions in both systems, i.e.,

$$x_1(n\triangle+\epsilon)+x_2(n\triangle+\epsilon)=\hat{x}_1(n\triangle+\epsilon)+\hat{x}_2(n\triangle+\epsilon)=B. \tag{A.28}$$

Case 1 ($x_1(n\triangle-\delta)\geq B_1$ and $\hat{x}_1(n\triangle-\delta)\geq\hat{B}_1$)**:** Traffic Class 1 is subscribing equal to or larger than the allocated buffer space in both systems. Therefor one of the following 3 cases occurs:

- If $a_2(n)<\hat{x}_1(n\triangle-\delta)-\hat{B}_1$,

$$\begin{cases} x_1(n\triangle+\epsilon) &= x_1(n\triangle-\delta)-a_2(n)>B_1, \quad \text{and} \\ \hat{x}_1(n\triangle+\epsilon) &= \hat{x}_1(n\triangle-\delta)-a_2(n)>\hat{B}_1. \end{cases} \tag{A.29}$$

- If $\hat{x}_1(n\triangle-\delta)-\hat{B}_1\leq a_2(n)<x_1(n\triangle-\delta)-B_1$,

$$\begin{cases} x_1(n\triangle+\epsilon) &= x_1(n\triangle-\delta)-a_2(n)>B_1, \quad \text{and} \\ \hat{x}_1(n\triangle+\epsilon) &= \hat{B}_1. \end{cases} \tag{A.30}$$

- If $a_2(n)>x_1(n\triangle-\delta)-B_1$,

$$\begin{cases} x_1(n\triangle+\epsilon) &= B_1, \quad \text{and} \\ \hat{x}_1(n\triangle+\epsilon) &= \hat{B}_1. \end{cases} \tag{A.31}$$

Hence, Inequalities (A.21) and (A.22) hold.

Case 2 ($x_1(n\Delta - \delta) \geq B_1$ and $\hat{x}_1(n\Delta - \delta) < \hat{B}_1$)**:** Traffic Class 1 is subscribing the buffer equal to or larger than the allocated space, B_1, in System B while it is under-subscribing the buffer space in System $\hat{B}$. This leads to the following 4 cases:

- If $a_1(n) < \hat{B}_1 - \hat{x}_1(n\Delta - \delta)$ and $a_2(n) < x_1(n\Delta - \delta) - B_1$,

$$\begin{cases} x_1(n\Delta + \epsilon) &= x_1(n\Delta - \delta) - a_2(n) > B_1, \quad \text{and} \\ \hat{x}_1(n\Delta + \epsilon) &= \hat{x}_1(n\Delta - \delta) + a_1(n) < \hat{B}_1. \end{cases} \tag{A.32}$$

- If $a_1(n) < \hat{B}_1 - x_1(n\Delta - \delta)$ and $a_2(n) \geq x_1(n\Delta - \delta) - B_1$.

$$\begin{cases} x_1(n\Delta + \epsilon) &= B_1, \quad \text{and} \\ \hat{x}_1(n\Delta + \epsilon) &= \hat{x}_1(n\Delta - \delta) + a_1(n) < \hat{B}_1. \end{cases} \tag{A.33}$$

- If $a_1(n) \geq \hat{B}_1 - x_1(n\Delta - \delta)$ and $a_2(n) < x_1(n\Delta - \delta) - B_1$,

$$\begin{cases} x_1(n\Delta + \epsilon) &= x_1(n\Delta - \delta) - a_2(n) > B_1, \quad \text{and} \\ \hat{x}_1(n\Delta + \epsilon) &= \hat{B}_1. \end{cases} \tag{A.34}$$

- If $a_1(n) \geq \hat{B}_1 - x_1(n\Delta - \delta)$ and $a_2(n) \geq x_1(n\Delta - \delta) - B_1$,

$$\begin{cases} x_1(n\Delta + \epsilon) &= B_1, \quad \text{and} \\ \hat{x}_1(n\Delta + \epsilon) &= \hat{B}_1. \end{cases} \tag{A.35}$$

Hence, Inequalities (A.21) to (A.22) hold.

Case 3 ($x_1(n\triangle - \delta) < B_1$ and $\hat{x}_1(n\triangle - \delta) < \hat{B}_1$)**:** Traffic Classes 1 is under-subscribing the buffer space in both systems. Thus, one of the following 3 cases occurs:

- If $a_1(n) < B_1 - x_1(n\triangle - \delta)$,

$$\begin{cases} x_1(n\triangle + \epsilon) &= x_1(n\triangle - \delta) + a_1(n) < B_1, \quad \text{and} \\ \hat{x}_1(n\triangle + \epsilon) &= \hat{x}_1(n\triangle - \delta) + a_1(n) < \hat{B}_1. \end{cases} \tag{A.36}$$

- If $B_1 - x_1(n\triangle - \delta) \leq a_1(n) < \hat{B}_1 - \hat{x}_1(n\triangle - \delta)$,

$$\begin{cases} x_1(n\triangle + \epsilon) &= B_1, \quad \text{and} \\ \hat{x}_1(n\triangle + \epsilon) &= \hat{x}_1(n\triangle - \delta) + a_1(n) < \hat{B}_1. \end{cases} \tag{A.37}$$

- If $a_1(n) \geq \hat{B}_1 - \hat{x}_1(n\triangle - \delta)$,

$$\begin{cases} x_1(n\triangle + \epsilon) &= B_1, \quad \text{and} \\ \hat{x}_1(n\triangle + \epsilon) &= \hat{B}_1. \end{cases} \tag{A.38}$$

Hence, Inequalities (A.21) and (A.22) hold.

■

Note that due to Remark 3 and the proof of Lemma 2, the following equation holds:

Remark 4

$$x_1(n\triangle + \epsilon) + x_2(n\triangle + \epsilon) = \hat{x}_1(n\triangle + \epsilon) + \hat{x}_2(n\triangle + \epsilon), \tag{A.39}$$

at any time slot n.

A.4 Non-increasing Relation

The following property holds for the system at $n-1$ and n.

Lemma 3 *Assume that*

$$\hat{B}_k > B_k. \tag{A.40}$$

If

$$x_k(n-1) \leq \hat{x}_k(n-1) \tag{A.41}$$

and

$$\hat{x}_k(n-1) - x_k(n-1) \leq \hat{B}_k - B_k, \tag{A.42}$$

then

$$x_k(n) \leq \hat{x}_k(n) \tag{A.43}$$

and

$$\hat{x}_k(n) - x_k(n) \leq \hat{B}_k - B_k. \tag{A.44}$$

Proof: By definition, $x_k(n-1) = x_k(n\triangle - \epsilon)$ and $\hat{x}_k(n-1) = \hat{x}_k(n\triangle - \epsilon)$, $k = 1, 2$. Also by definition, $x_k(n) = x_k(n\triangle + \epsilon)$ and $\hat{x}_k(n) = \hat{x}_k(n\triangle + \epsilon)$, $k = 1, 2$. Therefore, it is straightforward form Corollary 1 and Lemma 2 that Inequalities (A.43) and (A.44) hold.

■

Assuming that the buffer is empty at the initiation of the system, the following property of the cell loss ratio in the two systems is derived.

Theorem 2 *Given identical input traffic, consider two buffer space allocations by the Work-conserving WRR-CSVP scheme,* (B_1, B_2) *and* $(\hat{B}_1, \hat{B}_2)$*, where* $B_1 + B_2 = \hat{B}_1 + \hat{B}_2 = B$*. If*

$$\hat{B}_k > B_k \tag{A.45}$$

and the initial condition is given by

$$x_k(0) = \hat{x}_k(0) = 0, \tag{A.46}$$

then

$$\hat{r}_k \leq r_k, \tag{A.47}$$

for $k = 1, 2$.

Proof: By induction, Lemma 3 with the initial condition, Equation (A.46), gives that

$$x_k(n) \leq \hat{x}_k(n) \tag{A.48}$$

at any time slot n.

Denote the total number of cells transmitted during $[0, n]$ by $U_k(n)$ and $\hat{U}_k(n)$ in Systems B and $\hat{B}$, respectively, i.e., $U_k(n) = \sum_{i=0}^{n} u_k(i)$ and $\hat{U}_k(n) = \sum_{i=0}^{n} \hat{u}_k(i)$. From Lemma 1 and Inequality (A.48), it is obvious that

$$U_k(n) \leq \hat{U}_k(n). \tag{A.49}$$

Let $L_k(n)$ and $\hat{L}_k(n)$ be the number of cells blocked during $[0, n]$ in Systems B and $\hat{B}$, respectively. And let $P_k(n)$ and $\hat{P}_k(n)$ be the number of cells pushed out during $[0, n]$ in Systems B and $\hat{B}$, respectively.

The number of arrivals belonging to Traffic Class k during $[0, n]$, $A_k(n) = \sum_{m=0}^{n} a_k(m)$, is equal in both systems, i.e.,

$$A_k(n) = L_k(n) + P_k(n) + U_k(n) + x_k(n) = \hat{L}_k(n) + \hat{P}_k(n) + \hat{U}_k(n) + \hat{x}_k(n). \tag{A.50}$$

at any time slot n. Thus, from Inequalities (A.48) and (A.49),

$$L_k(n) + P_k(n) \geq \hat{L}_k(n) + \hat{P}_k(n), \tag{A.51}$$

at any time slot n. The number of arrivals in the two systems being equal also gives

$$\lim_{n\to\infty}[\{(L_k(n)+P_k(n)+U_k(n)\}-\{\hat{L}_k(n)+\hat{P}_k(n)+\hat{U}_k(n)\}]=0. \tag{A.52}$$

Hence

$$r_k=\lim_{n\to\infty}\frac{L_k(n)+P_k(n)}{L_k(n)+P_k(n)+U_k(n)}\geq\lim_{n\to\infty}\frac{\hat{L}_k(n)+\hat{P}_k(n)}{\hat{L}_k(n)+\hat{P}_k(n)+\hat{U}_k(n)}=\hat{r}_k. \tag{A.53}$$

■

Appendix B

Additional Data Of Single Node Case

B.1 Traffic Types

The Traffic Types used in our emulation are listed in Table B.1. Types 1 to 7 are chosen to possess the same average load: Type 1 is a Bernoulli process and thus is nonbursty; and Types 2 to 7 are bursty with different parameter values so that they have different burstiness. Types 8 and 9 are used to create a traffic situation of two Traffic Classes. Uneven traffic load situations are created using Types 10 to 13. Type 14 is used to examine the prioritization of the pushout and the time slot reassignment operations.

Let us examine Traffic Types 1 to 7 which possess the same average load. By creating homogeneous traffic situations using these and comparing the performance, the difference of these Traffic Types can be assessed. The homogeneous traffic sit-

Table B.1: Traffic Types

Type	α_i	β_i	λ_i	$\overline{\lambda_i}$	$\overline{p}_i \times \lambda_i$
1	1.0	0.0	0.03	0.03	–
2	0.001	0.002	0.09	0.03	45
3	0.0008	0.004	0.18	0.03	45
4	0.0005	0.001	0.09	0.03	90
5	0.0004	0.002	0.18	0.03	90
6	0.0002	0.001	0.18	0.03	180
7	0.0001875	0.015	0.27	0.03	180
8	0.001	0.002	0.14	$0.04666\cdots$	70
9	0.002	0.006	0.16	0.04	$26.66\cdots$
10	1.0	0.0	0.0225	0.0225	–
11	0.001	0.002	0.0675	0.0225	33.75
12	0.0005	0.001	0.0675	0.0225	67.5
13	0.001	0.005	0.135	0.0225	27
14	0.001	0.002	0.055	0.018	27.5

uations are created as follows: construct three statistically identical Traffic Classes each of which is generated by a superposition of ten identical mini-sources; the homogeneous cases, Cases i, $i = 1, \cdots, 7$, are created by the mini-sources belonging to Traffic Type i; each Traffic Class receives 300 cell buffer space and 1/3 of total link capacity by the Work-conserving WRR-CSVP resource allocation mechanism.

The cell loss, cell delay, and cell delay variation performances of these cases are presented in Table B.2. Let us discuss the measure of burstiness. If the traffic flows with the same average cell generation rate experience different cell loss ratios, the difference must be caused by the different *burstiness* of the traffic flows. Therefore, the measure of burstiness should reflect the different cell loss ratios, i.e., a good measure of burstiness should allow us to estimate the cell loss ratio. Our experiment

Table B.2: System performance with homogeneous traffic situation (cell loss ratio: $\times 10^{-3}$; cell delay and cell delay variation: $\times 10^{2}$ slot time, 95% confidence interval)

	Cell loss ratio	Cell delay	Delay variation
Case 1	(undetected)	0.053 ± 0.00019	0.058 ± 0.00036
Case 2	0.37 ± 0.10	1.1 ± 0.046	2.0 ± 0.080
Case 3	3.0 ± 0.38	2.0 ± 0.056	2.9 ± 0.061
Case 4	3.5 ± 0.26	1.8 ± 0.086	3.1 ± 0.075
Case 5	31 ± 2.6	3.2 ± 0.100	4.4 ± 0.057
Case 6	13 ± 1.2	2.6 ± 0.076	3.7 ± 0.062
Case 7	45 ± 1.5	3.4 ± 0.050	4.3 ± 0.039

shows that the Traffic Classes which possess the same value of $\overline{p}_i \times \lambda_i$ result in different cell loss ratios. This measure does not reflect the resultant cell loss ratios. It is clear that some other measures of burstiness are necessary to estimate the cell loss performance, and to obtain a good resource dimensioning to develop a better call admission control mechanism. Table B.3 presents the wasted slot rates of each class when the WRR scheduling scheme is adopted. It shows that the bursty

Table B.3: Wasted slot rates of WRR in homogeneous traffic situation ($\times 10^{-1}$, 95% confidence interval)

	Wasted slot rate
Case 1	0.99 ± 0.0030
Case 2	1.0 ± 0.025
Case 3	1.1 ± 0.017
Case 4	1.1 ± 0.018
Case 5	1.7 ± 0.027
Case 6	1.4 ± 0.023
Case 7	1.9 ± 0.027

traffic causes larger waste of time slots. The adoption of the Work-conserving WRR

scheduling scheme is especially important when the traffic is bursty.

B.2 Advantages of the Work-conservation WRR Scheduling Scheme

Considering the different cell loss ratios in Table B.2 and take the measure of burstiness, $\overline{p}_i \times \lambda_i$, into account, we choose Traffic Types 1, 2, 4, and 6 to represent the different level of burstiness in order to further examine the advantages of the Work-conserving WRR scheduling scheme.

The heterogeneous traffic situations, Cases 8 to 10, are created by the various combinations of Traffic Types 1, 2, 4, and 6, which represent different level of burstiness (Table B.4). The result for Case 8 is presented in Chapter 4 as Case 1.

Table B.4: Heterogenous traffic situations

	Class 1	Class 2	Class 3
Case 8	Type 1 × 10	Type 2 × 10	Type 4 × 10
Case 9	Type 1 × 10	Type 2 × 10	Type 6 × 10
Case 10	Type 1 × 10	Type 4 × 10	Type 6 × 10
Case 11	Type 2 × 10	Type 4 × 10	Type 6 × 10

Here, the rest of the cases, Cases 9 to 11, are shown. The amount of allocated resource is the same as Case 1 in Chapter 4.

Table B.5 shows the cell loss performance for Cases 9 to 11. The advantages of the Work-conserving WRR scheduling scheme is evident.

The cell delay and cell delay performances for Cases 9 to 11 are shown in Ta-

Table B.5: Cell loss performance ($\times 10^{-3}$, 95% confidence interval)

	Case 9		Case 10	
	WRR	WC-WRR	WRR	WC-WRR
Aggregate	21 ± 2.0	6.3 ± 0.48	25 ± 1.4	8.4 ± 0.76
Class 1	(undetected)	(undetected)	(undetected)	(undetected)
Class 2	4.9 ± 0.69	0.75 ± 0.17	15 ± 1.4	3.4 ± 0.63
Class 3	59 ± 5.4	18 ± 1.3	60 ± 4.2	22 ± 1.7

	Case 11	
	WRR	WC-WRR
Aggregate	33 ± 1.2	8.4 ± 0.76
Class 1	6.6 ± 0.81	1.0 ± 0.26
Class 2	15 ± 1.3	3.4 ± 0.63
Class 3	72 ± 3.3	22 ± 1.7

bles B.6 to B.8, respectively. This also shows the advantages of the Work-conserving WRR scheduling scheme.

Table B.6: Cell delay and cell delay variation performances (Case 9, $\times 10^2$, 95% confidence interval)

(cell time)	Mean cell delay		Standard deviation	
	WRR	WC-WRR	WRR	WC-WRR
Aggregate	4.4 ± 0.069	2.0 ± 0.052	6.6 ± 0.078	4.2 ± 0.061
Class 1	0.16 ± 0.00053	0.086 ± 0.0014	0.14 ± 0.0011	0.10 ± 0.0013
Class 2	3.7 ± 0.13	1.1 ± 0.049	4.2 ± 0.16	2.9 ± 0.051
Class 3	9.6 ± 0.19	4.9 ± 0.12	8.4 ± 0.083	6.1 ± 0.066

The wasted slot rates are shown in Table B.9.

Table B.7: Cell delay and cell delay variation performances (Case 10, $\times 10^2$, 95% confidence interval)

(cell time)	Mean cell delay		Standard deviation	
	WRR	WC-WRR	WRR	WC-WRR
Aggregate	4.9 ± 0.068	2.2 ± 0.10	6.9 ± 0.061	4.3 ± 0.11
Class 1	0.16 ± 0.00086	0.085 ± 0.0015	0.14 ± 0.0015	0.10 ± 0.00097
Class 2	5.4 ± 0.19	1.6 ± 0.13	5.8 ± 0.17	3.0 ± 0.17
Class 3	9.3 ± 0.24	4.7 ± 0.17	8.2 ± 0.093	6.0 ± 0.100

Table B.8: Cell delay and cell delay variation performances (Case 11, $\times 10^2$, 95% confidence interval)

(cell time)	Mean cell delay		Standard deviation	
	WRR	WC-WRR	WRR	WC-WRR
Aggregate	5.6 ± 0.065	2.3 ± 0.11	6.1 ± 0.052	4.0 ± 0.10
Class 1	3.5 ± 0.076	1.1 ± 0.081	3.9 ± 0.056	2.0 ± 0.12
Class 2	5.0 ± 0.19	1.6 ± 0.091	5.3 ± 0.14	2.8 ± 0.12
Class 3	8.3 ± 0.15	4.3 ± 0.21	7.4 ± 0.075	5.5 ± 0.12

Table B.9: Wasted slot rates ($\times 10^{-1}$, 95% confidence interval)

	Case 9	Case 10	Case 11
Aggregate	1.2 ± 0.020	1.2 ± 0.016	1.3 ± 0.017
Class 1	0.99 ± 0.0036	0.99 ± 0.0051	1.0 ± 0.024
Class 2	1.0 ± 0.025	1.1 ± 0.032	1.2 ± 0.034
Class 3	1.5 ± 0.060	1.5 ± 0.049	1.6 ± 0.055

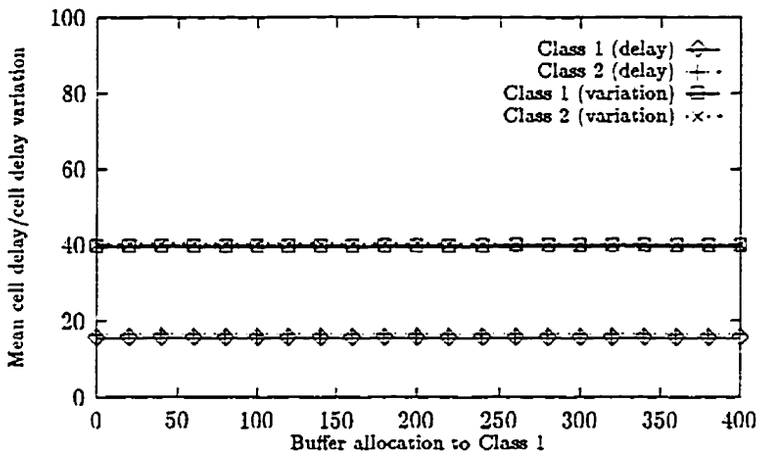

Figure B.1: Virtual partitioning vs. mean cell delay and cell delay variation

B.3 Virtual Partitioning

The relationship between virtual partitioning and cell delay and cell delay variation performance for the case of two Traffic Classes are shown in Figure B.1.

B.4 Exhaustive Service vs. Round Robin Service

Another uneven case, Case 12 is shown in Table B.10. Traffic Type 12 generating

Table B.10: Uneven traffic situation, Case 12

	Class 1	Class 2	Class 3
Case 12	Type 10 × 10	Type 11 × 10	Type 13 × 20

Traffic Class 3 in the uneven traffic situation in Chapter 4 is replaced by another bursty traffic, Traffic Type 13. The amount of resource allocated is the same as

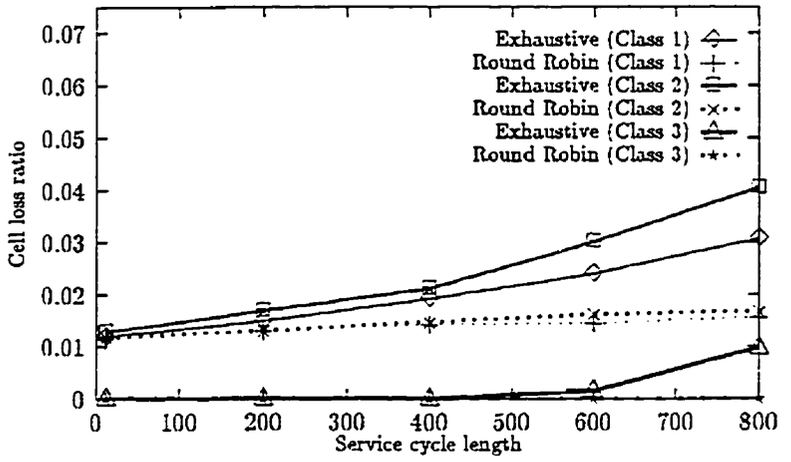

Figure B.2: Service cycle length vs. cell loss ratio (WRR)

Case 3 in Chapter 4.

The cell loss, cell delay, and cell delay variation performances for the different service cycle length of the WRR scheduling scheme are presented in Figures B.2 to B.4, respectively.

The cell loss, cell delay, and cell delay variation performances for the different service cycle length of the Work-conserving WRR scheduling scheme are presented in Figures B.5 to B.7, respectively.

B.5 Traffic Class Selection Methods

An uneven traffic situation shown in Table B.11 is created. The average cell generation rate of Traffic Class 2 is twice as much as that of the other Traffic Classes. Thus, twice as much capacity and buffer space is given to Traffic Class 2. Each of Traffic Classes 1 and 3 receives 1/4 of the total capacity and Traffic Class 2 receives

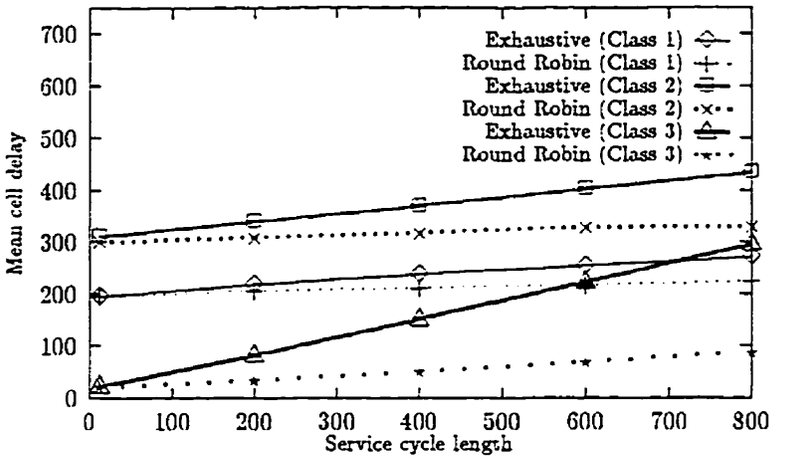

Figure B.3: Service cycle length vs. mean cell delay (WRR)

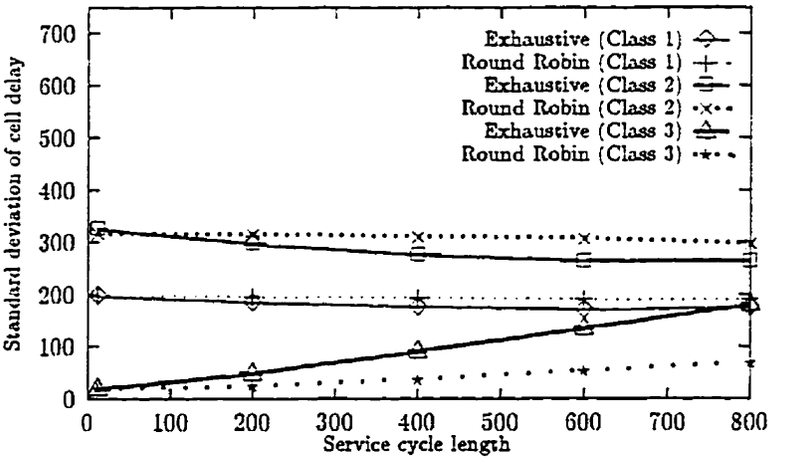

Figure B.4: Service cycle length vs. cell delay variation (WRR)

Table B.11: Uneven traffic situation, Case 13

	Class 1	Class 2	Class 3
Case 13	Type 10 × 10	Type 11 × 20	Type 12 × 10

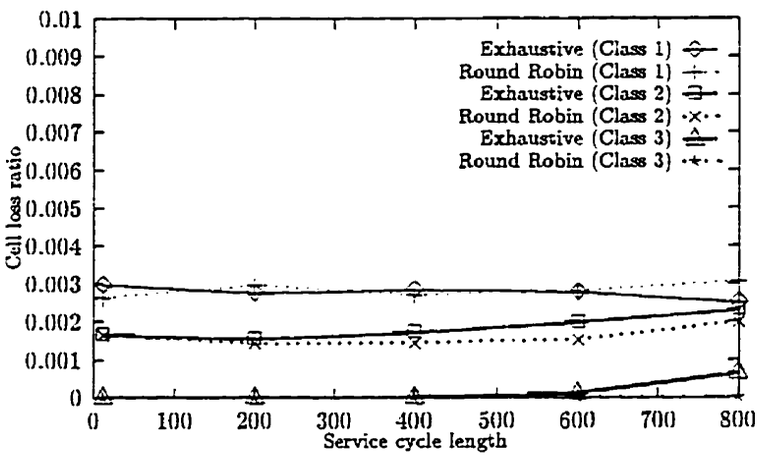

Figure B.5: Service cycle length vs. cell loss ratio (Work-conserving WRR)

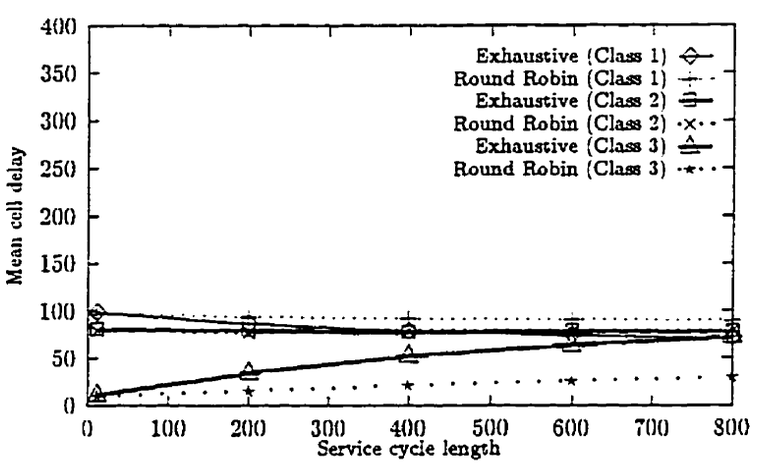

Figure B.6: Service cycle length vs. mean cell delay (Work-conserving WRR)

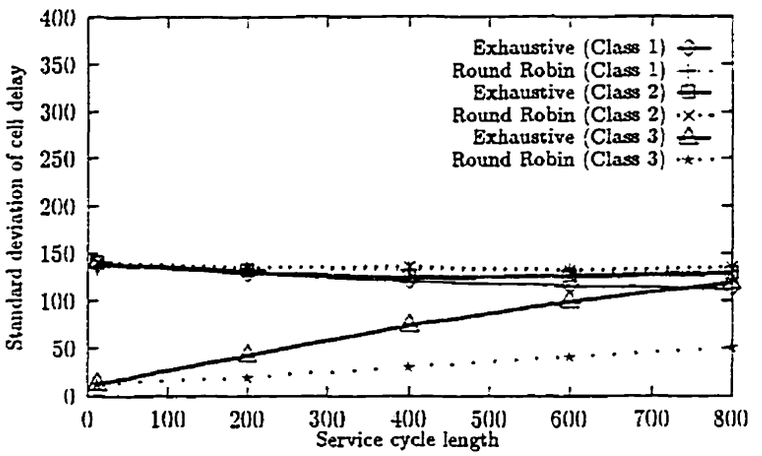

Figure B.7: Service cycle length vs. cell delay variation (Work-conserving WRR)

1/2 by $S = 12$, and $S_1 = S_3 = 3$ and $S_2 = 6$. The total buffer size is 400 cells; and 100 cells are allocated to each of Traffic Classes 1 and 3, and 200 cells to Traffic Class 2, i.e., $B_1 = B_3 = 100$ and $B_2 = 200$. In Tables B.12 to B.14, the similar results discussed in Section 4.4.2 can be seen.

Table B.12: Cell loss ratio of different traffic class selection methods for the pushout operation ($\times 10^{-3}$, 95% confidence interval)

	Random	Absolute excess	Relative excess
Class 1	(undetected)	(undetected)	(undetected)
Class 2	1.2 ± 0.24	1.3 ± 0.28	1.2 ± 0.16
Class 3	3.6 ± 0.25	3.7 ± 0.73	3.5 ± 0.40

Table B.13: Cell loss ratio of different traffic class selection methods for the time slot reassignment operation ($\times 10^{-3}$, 95% confidence interval)

	Random	Absolute occup.	Relative occup.	Absolute excess
Class 1	(undetected)	(undetected)	(undetected)	(undetected)
Class 2	1.2 ± 0.24	1.3 ± 0.17	1.5 ± 0.24	1.3 ± 0.22
Class 3	3.6 ± 0.25	4.0 ± 0.46	3.4 ± 0.38	3.8 ± 0.49

Table B.14: Cell loss ratio of different combinations of traffic class selection methods for the pushout and the time slot reassignment operations ($\times 10^{-3}$, 95% confidence interval)

time slot	Random	Absolute occupancy	
pushout	Random	Absolute excess	Relative excess
Class 1	(undetected)	(undetected)	(undetected)
Class 2	1.2 ± 0.24	1.1 ± 0.25	1.3 ± 0.29
Class 3	3.6 ± 0.25	3.4 ± 0.69	4.1 ± 0.50

time slot	Relative occupancy		Absolute excess	
pushout	Absolute excess	Relative excess	Absolute excess	Relative excess
Class 1	(undetected)	(undetected)	(undetected)	(undetected)
Class 2	1.4 ± 0.24	1.6 ± 0.14	1.3 ± 0.22	1.4 ± 0.22
Class 3	3.6 ± 0.40	4.0 ± 0.43	3.6 ± 0.59	4.0 ± 0.60

Appendix C

Design Specification Of The Work-conserving WRR-CSVP

C.1 General Scope

This specification deals with the design of the memory space and the processing entities to implement the Work-conserving WRR-CSVP resource allocation mechanism at an output port of an ATM switch. The general structure is illustrated in Figure 6.1 and the implementation strategies are discussed in Section 6.2. The data structures designed on the memory space are specified in detail here but the actual implementation of the memory space is left to the implementer's decision. The larger memory segments may require RAM but some memory remnants may be implemented as registers. The algorithms are specified using flowcharts of pseudo-assembler but the actual implementation of the SMs such as CMOS design is not discussed. However, it should not be too difficult to develop ASMs for the PB-SM

and the CB-SM since these are essentially copying operations of cells. The CX-SM performs a standard electrical-to-optical (E/O) conversion and transmits a cell into the output link. Therefore, this also is easily developed as an E/O module and its peripheral. The rest of the SMs, the SS-SM and the BA-SM, may be implemented using a microprocessor.

This design specification considers a VC based management of buffer space and capacity allocations. The cells belonging to the same VC in the buffer are maintained by VC queues. Call admission is outside of scope of this design specification and assumed to be given.

The design of memory space is described in Section C.2. The specifications of the SMs are found in Section C.3. The SMs may share the same memory space. Therefore, there are potential memory contention problems. The solutions to these problems are briefly discussed in Section C.4.

C.2 Specification of Memory Space Design

C.2.1 Addressing of the Memory Space

The size of address space should be examined. Let us assume that a maximum number of VCs is $K = 1024$. For example, if 2 cells are allocated to each VC, we have a total buffer size of $B = 2048$. Since the size of a cell is 53 bytes, the BS must be at least 110592 bytes. Thus, at least 3 bytes are necessary to handle this address space. We adopt more commonly used 4-byte address. This should provide sufficient address space. In order to simplify some operations and enable the use

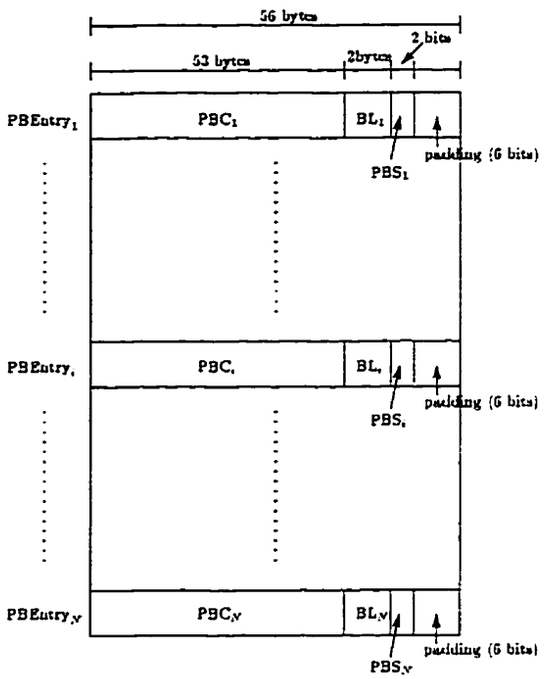

Figure C.1: The Pre-Buffer (PB)

of commonly available packages, the PB and the BS segments are padded to make the size of each entry of the segments in a multiple of bytes.

C.2.2 Memory Segments to Store Cells

Pre-Buffer (PB) segment

The PB segment is illustrated in Figure C.1. The cells from the output port of the switching fabric is temporarily stored in the PB segment by the PB-SM before being stored in the actual buffer space in the BS segment. The PB segment is comprised of N entries, where N is the switch size. Each entry, PBEntry_i, $i = 1, 2, \cdots, N$,

consists of three fields specified in the following. The subscript i of these fields indicates that the field belongs to the PBEntry_i. The size of an entry is 58 bytes including padding.

PBC (Pre-Buffer Cell) field The PBC_i, $i = 1, 2, \cdots, N$, field is a field to store a cell from the switching fabric temporarily. The PB-SM writes a cell in this field; the BA-SM reads the header information of the cell to examine the admission; and the CB-SM copys the cell to the BS. Since the size of an ATM cell is 53 bytes, the size of the PBC_i field is 53 bytes.

BL (Buffer Location) field The BL_i, $i = 1, 2, \cdots, N$, field is a 4-byte field to store a location in the BS. The BA-SM finds a space in the BS for the cell in PBC_i according to the CSVP buffer allocation scheme and writes the address to this field; and the CB-SM reads this field and transfers the cell in PBC_i to the location.

PBS (Pre-Buffer Status) field The PBS_i, $i = 1, 2, \cdots, N$, field is a 2-bit field which indicates the state of the PBEntry_i. The value of this field indicates the following states:

- If PBS_i = '00', the PBEntry_i is idle.
- If PBS_i = '01', a cell from the switching fabric is stored in the PBC_i by the PB-SM.
- If PBS_i = '10', the cell in the PBC_i is admitted to the buffer and the space in the BS is prepared by the BA-SM. The location is stored in the BL_i field.

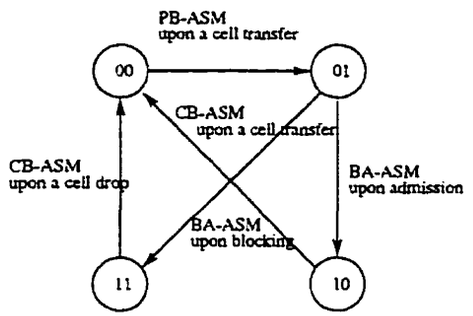

Figure C.2: The state transition of the PBS_i field

- If PBS_i = '11', the admission decision of the cell in the PBC_i by the BA-SM is to discard.

The PB-SM changes this field from '00' to '01' upon the completion of cell transfer: the BA-SM changes this field from '01' to '10' after filling the BL_i field, or from '10' to 11' upon the decision to discard; and the CB-SM changes this field from '10' or '11' to '00'. The state transition diagram is shown in Figure C.2.

Buffer Space (BS) segment

The BS segment is illustrated in Figure C.3. The BS segment is the implementation of the actual output buffer space. Since the size of buffer is B, the BS segment is comprised of B entries. Each entry, $BSEntry_j$, $j = 1, 2, \cdots, B$, consists of four fields specified in the following. These fields include the chain information to maintain the doubly link lists of the VC queues. The subscript j of these fields indicates that the field belongs to the $BSEntry_j$. The size of an entry is 62 bytes including padding.

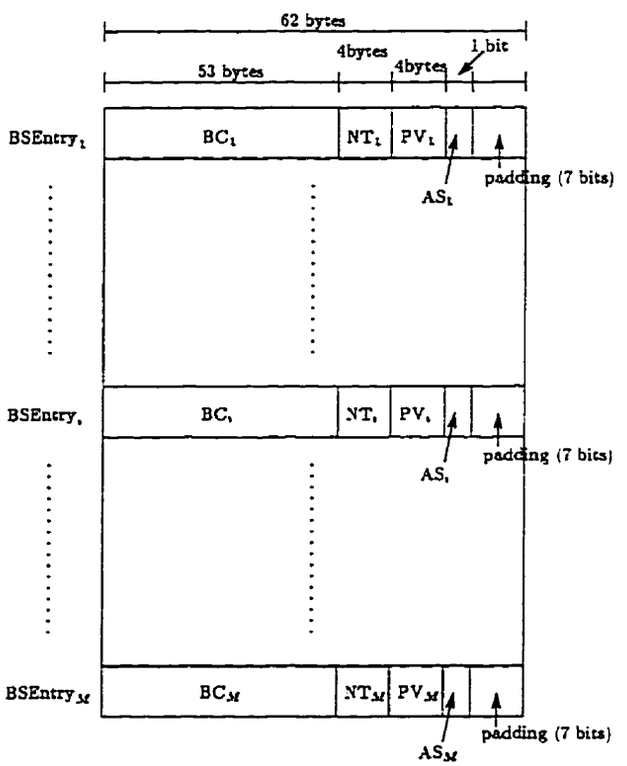

Figure C.3: The Buffer Space (BS)

BC (Buffer Cell) field The BC_i, $i = 1, 2, \cdots, B$, field is a field to store a cell for output buffering. The CB-SM transfers a cell from the PB to this field; and the SS-SM moves the cell in this field to the XB. Since the size of an ATM cell is 53 bytes, the size of the BC_i field is 53 bytes.

NT (Next) field The NT_i, $i = 1, 2, \cdots, B$, field is a 4-byte field to store the address of the next cell in the VC queue.

PV (Previous) field The PV_i, $i = 1, 2, \cdots, B$, field is a 4-byte field to store the address of the previous cell in the VC queue.

AS (Availability Status) field The AS_i, $i = 1, 2, \cdots, B$, field is a 1-bit field used by the enqueue and dequeue functions of the Free Queue of the BSEntry's to indicate the status of the $BSEntry_i$. The value of the AS_i field indicates the following states:

- If AS_i = '0', the $BSEntry_i$ is in the free queue.
- If AS_i = '1', the $BSEntry_i$ is active.

Transmission Buffer (XB) segment

The XB segment is illustrated in Figure C.4. In order to avoid memory contention problem between the SS-SM and the CX-SM, the XB segment is placed between the BS segment and the output link. The XB segment is a ring buffer consisting of three entries. By having three entries, the complete independence of the two SMs is achieved even if there is a subtle timing problem. It should be noted, however,

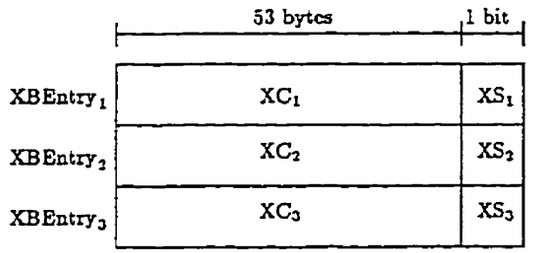

Figure C.4: The Transmission Buffer (XB)

that a sophisticated hardware implementation can avoid the problem with two entries. The SS-SM selects a cell in the BS according to the Work-conserving WRR scheduling scheme and copys the cell to the XB; and the CX-SM transmits the cell in the XB to the output link. Each entry of the XB segment, XBEntry_i, $i = 1, 2, 3$, consists of two fields specified in the following. The subscript i of these fields indicates that the field belongs to the XSEntry_i. The size of an entry is 54 bytes including padding.

XC (Transmission Cell) field The XC_i, $i = 1, 2, 3$, field is a field to store a cell to be transmitted. The SS-SM transfers a cell in the BS to this field; and the CX-SM transmits the cell in this field. Since the size of an ATM cell is 53 bytes, the size of the XC_i field is 53 bytes.

Transmission Status (XS) field The XS_i, $i = 1, 2, 3$, field is a 1-bit field to indicate the state of the XBEntry_i. The value of the XS_i field indicates the following states:

- If XS_i = '0', the BSEntry_i is idle.

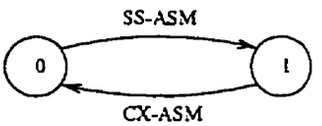

Figure C.5: The state transition of the XS_i field

- If XS_i = '1', a cell is set in the XC_i for transmission.

The SS-SM changes this field from '0' to '1'; and the CX-SM changes it from '1' to '0'. The state transition diagram is shown in Figure C.5.

C.2.3 Memory Segment to Manage The VCs

The VCs are managed by the VCT segment. In the following, 2 bytes are assumed for the counters, such as the size of allocated buffer, the buffer occupancy, the number of time slots to be allocated within a service cycle, and the time slots consumed. For these counters, a 2-byte integer is sufficiently large.

VCT (Virtual Circuit Table) segment

The VCT segment is illustrated in Figure C.4. The VCT contains the information necessary to maintain and operate the VCs at the output port system. This includes the field to maintain the VC queues. The VCT is comprised of K entries, where K is a maximum number of VCs supported at this output port. Each entry, VCTEntry_k, $k = 1, 2, \cdots, K$, consists of eight fields specified in the following. The subscript k of these fields indicates that the field belongs to the VCTEntry_k. The size of an entry is 20 bytes. Since call admission is assumed to be given, we assume that the

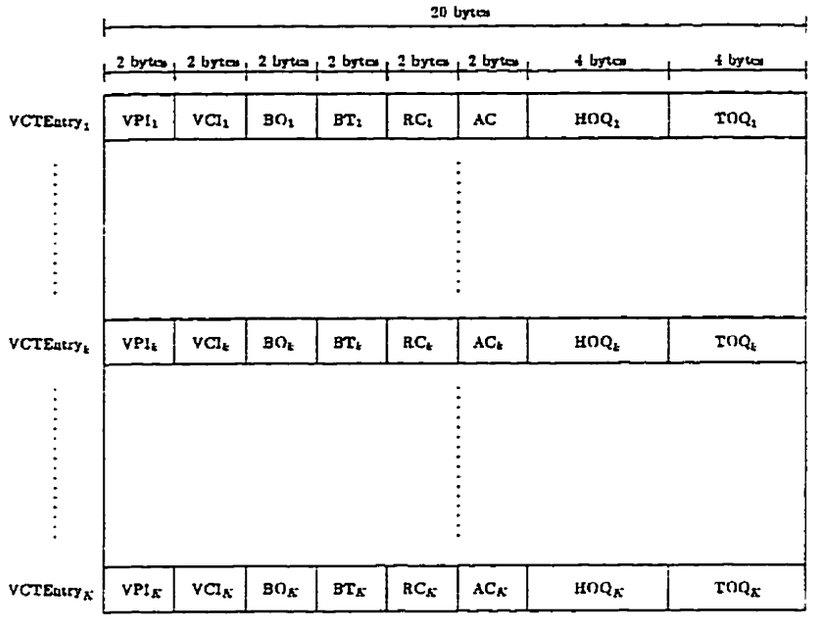

Figure C.6: The Virtual Circuit Table (VCT)

VPI, VCI, BT, and AC fields are set.

The WRR scheduling scheme is performed using the VCT, where the active VCs are listed from the top consecutively. This implementation is simple and sufficient.

VPI (Virtual Path Identifier) field The VPI_k, $k = 1, 2, \cdots, K$, field is a field to store VPI of the ATM connection. Although the size of the VPI_k field is 12 bits, 2 bytes are allocated to this field to make it a multiple of bytes.

VCI (Virtual Channel Identifier) field The VCI_k, $k = 1, 2, \cdots, K$, field is a 2-byte field to store VCI of the ATM connection. The combination of the VPI_k field and this field is used by the BA-SM to identify the entry of the VCT representing the VC to which the cell in the PB belongs.

BO (Buffer Occupancy) field The BO_k, $k = 1, 2, \cdots, K$, field is a 2-byte field to store the number of cells belonging to the VC of the $VCTEntry_k$ in the BS. The SS-SM decrements the value of this field upon a service; and the BA-SM increments upon admission of a cell and decrements upon a pushout of a cell.

BT (Buffer Threshold) field The BT_k, $k = 1, 2, \cdots, K$, field is a 2-byte field to store the size of the allocated buffer space in the number of cells by the virtual partition of the CSVP buffer allocation scheme to the VC of the $VCTEntry_k$.

RC (Remaining Capacity) field The RC_k, $k = 1, 2, \cdots, K$, field is a 2-byte field to store the amount of remaining time slots for the VC of the $VCTEntry_k$ in the current service cycle. The SS-SM renews the value of this field at the beginning of a service cycle and decrements each time the WRR scheduling scheme selects the VC of the $VCTEntry_k$ for the time slot.

AC (Allocated Capacity) field The AC_k, $k = 1, 2, \cdots, K$, field is a 2-byte field to store the amount of allocated capacity in the number of time slots in a service cycle to the VC of the $VCTEntry_k$.

HOQ (Head of Queue) field The HOQ_k, $k = 1, 2, \cdots, K$, field is a 4-byte field to store the address of the cell at the head of the VC queue.

TOQ (Tail of Queue) field The TOQ_k, $k = 1, 2, \cdots, K$, field is a 4-byte field to store the address of the cell at the tail of the VC queue.

C.2.4 Memory Segments for the VC Selection Mechanism

Buffer Subscribing Connection Table (BSCT) segment

The BSCT segment consists of K entries. Each entry, BSCTEntry_k, $k = 1, 2, \cdots, K$ contains the addresses of the VCTEntry_k's of the VCs currently subscribing the buffer space, i.e., $\text{BO}_k > 0$. Thus the size of each entry is 4 bytes. The table is filled from the top consecutively. The detailed operation on this segment is provided in Section C.3.7.

Buffer Over-Subscribing Connection Table (BOSCT) segment

The BOSCT segment consists of $K - 1$ entries. Each entry, BOSCTEntry_k, $k = 1, 2, \cdots, K-1$, contains the address of the VCTEntry_k's of the VCs currently over-subscribing the buffer space, i.e., $\text{BO}_k > \text{BT}_k$. Thus the size of each entry is 4 bytes. The table is filled from the top consecutively. The detailed operation on this segment is provided in Section C.3.7.

C.2.5 Memory Remnants

Pre-buffering Status (PS) remnant The PS remnant is a 1-bit remnant to indicate the state of the pre-buffering process. The value of the PS remnant indicates the following states:

- If PS = '0', the pre-buffering of the cells from the switching fabric is currently undertaken by the PB-SM.

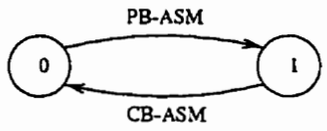

Figure C.7: The state transition of the PS_i remnant

- If PS = '1', the pre-buffering of the current time slot is completed by the PB-SM.

The PB-SM changes the value of this field from '0' to '1' when it completes the pre-buffering of the current time slot before halting; and the CB-SM changes the value of this field from '1' to '0' when it completes its operation for the current time slot. The state transition diagram is shown in Figure C.7.

Head of Free Queue (HOFQ) remnant The empty space in the BS is maintained by the Free Queue. The HOFQ remnant is a 4-byte remnant to store the address of the cell at the head of the Free Queue.

Tail of Free Queue (TOFQ) remnant The TOFQ remnant is a 4-byte remnant to store the address of the cell at the tail of the Free Queue.

Free Queue Length (FQL) remnant The FQL remnant is a 2-byte remnant to store the number of empty spaces in the BS, i.e., the length of the Free Queue.

Total Buffer Occupancy (TBO) remnant The TBO remnant is a 2-byte remnant to store current total buffer occupancy in the number of cells. The BA-SM increases the value of this field; and the SS-SM decreases it.

Total Buffer Size (TBS) remnant The TBS remnant is a 2-byte remnant to store the size of the buffer in the number of cells. The BA-SM uses this value for the cell admission process.

Remaining Service (RS) remnant The RS remnant is a 2-byte remnant to store the number of remaining time slots of the current service cycle. The SS-SM uses this remnant for the service scheduling.

Service Cycle (SC) remnant The SC remnant is a 2-byte remnant to store the length of the service cycle in the number of time slots. The SS-SM uses this remnant for the service scheduling.

Buffer Subscribing Connection Counter (BSCC) remnant The BSCC remnant is a 2-byte remnant to store the number of VCs which are currently subscribing the buffer space. Thus, it also indicates the number of entries filled from the top in the BSCT. The BA-SM increments the value of this remnant; and the SS-SM and BA-SM decreases it.

Buffer Over-Subscribing Connection Counter (BOSCC) remnant The BOSCC remnant is a 2-byte remnant to store the number of VCs which are currently over-subscribing the buffer space. Thus, it also indicates the number of entries filled from the top in the BOSCT. The BA-SM increments the value of this remnant; and the SS-SM and BA-SM decreases it.

Number of Virtual Circuit (NVC) remnant The NVC remnant is a 2-byte remnant to store the number of VCs currently supported by the output port. The SS-SM uses this value for the service scheduling. Since call admission is assumed to be given, we assume that a value is already set to this remnant.

Current Virtual Circuit Pointer (CVCP) remnant The CVCP remnant is a 4-byte remnant to store the address of the VCTEntry_k which receives the service in the current time slot. This remnant is used by the SS-SM.

Transmission Buffer Pointer (XBP) remnant The XBP remnant is a 4-byte remnant to store the address of the XBEntry_i. The SS-SM transfers a cell in the BS to the location indicated by XBP_i field when the scheduling decision is made.

C.3 Specifications of State Machines

The algorithm of the Work-conserving WRR scheduling scheme is performed by the SS-SM, and that of the CSVP buffer allocation scheme by the BA-SM. The flowchart of these SMs are transferred from the validated computer program of the emulator. The algorithms of the other SMs are straightforward.

C.3.1 Assumed Standard Procedures

The FIFO/LIFO operations of the VC queues and the Free Queue are assumed to be given. The operations assumed are:

- **Enqueue:** add an BSEntry to the tail of the queue.

- **Dequeue:** remove an BSEntry from the head of the queue.
- **Truncate:** remove an BSEntry from the tail of the queue.

The procedure to obtain an integer randomly from the range of 1 to a specified integer n, *random*(), is also assumed to be available by a pseudo-random number generator. The function, $m = random(n)$ returns an integer $m \in \{1, 2, \cdots, n\}$ at random, i.e., according to the uniform distribution.

The sorting procedures of tables by matching the value of the field(s) are trivial and are also assumed to be given.

The iteration operations are expressed using abstract parameters such as i, j, and k.

C.3.2 Pre-Buffering SM (PB-SM)

The flowchart of the PB-SM is shown in Figure C.8. The PB-SM transfers the cells from the switching fabric to the PBC_i fields of the PB. The PB-SM is activated at the beginning of each time slot. The PB-SM stores the cells from the top entry of the PB and advances to the subsequent entries. When the transfer of a cell is completed, the PB-SM changes the PBS_i field from '00' to '01'. When all cells arriving in the current time slot are stored in the BS, the PB-SM changes the PS from '0' to '1'. Then, it halts until the next time slot.

C.3.3 Cell Buffering SM (CB-SM)

The flowchart of the CB-SM is shown in Figure C.9. The CB-SM transfers the

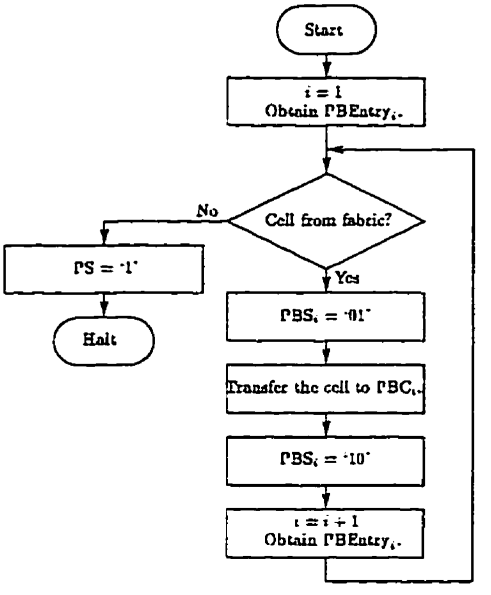

Figure C.8: The Pre-Buffering SM (PB-SM)

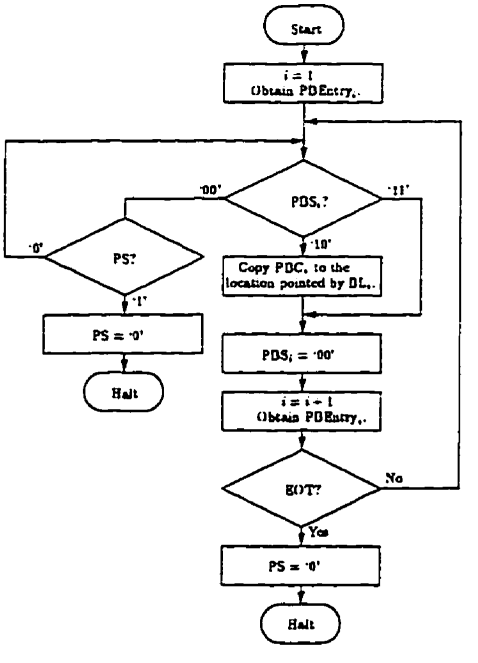

Figure C.9: The Cell Buffering SM (CB-SM)

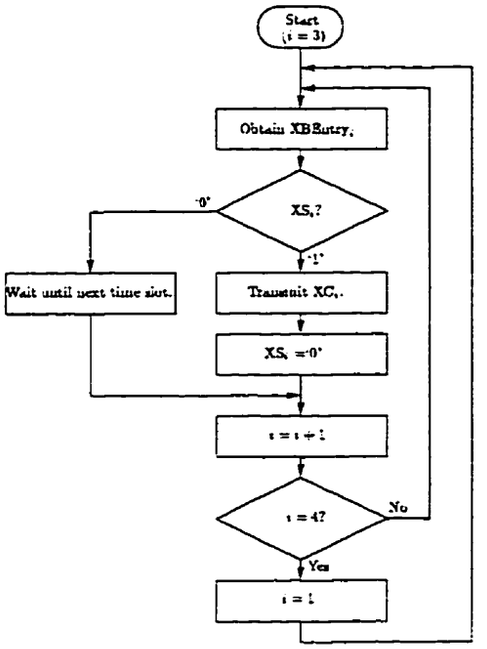

Figure C.10: The Cell Transmission SM (CX-SM)

cells in the PBC_i of the PB to the BC_j of the BS. The CB-SM is activated at the beginning of each time slot. The CB-SM deals with the PB from the top entry and advances to the subsequent entries. When the PBS_i turns to '10', the CB-SM transfers the cell to the BS at the location in BL_i, and turns PBS_i to '00'. When the PBS_i turns to '11', the CB-SM turns PBS_i to '00'. If the PBS_i remains as '00' and the PS turns to '1', the CB-SM changes the value of the PS from '1' to '0' and halts until the next time slot. The CB-SM also changes the value of the PS from '1' to '0' and halts when it reaches the end of the PB.

C.3.4 Cell Transmission SM (CX-SM)

The flowchart of the CX-SM is shown in Figure C.10. The CX-SM transmits a cell

in the XB to the output link. Unlike the other SMs, the CX-SM runs continuously and does not halt after a cell transmission since a cell transmission takes a cell time (one time slot). When the CX-SM sees the XS_i to be '1', it transmits the cell in XC_i to the output link. Then, when the cell transmission is completed, the CX-SM changes the value of the XS_i from '1' to '0'. When the CX-SM sees the XS_i to be '0', it waits until the next time slot. Then, the CX-SM advances to the next entry of the XB. Since the XB is a three-entry ring-buffer, the subsequent entry from $XBEntry_3$ is $XBEntry_1$. At the initiation of the system, the CX-SM starts its operation from the $XBEntry_3$ while the SS-SM from $XBEntry_1$. By having one entry between the two, the faster SS-SM never overwrites on the entry that the CX-SM is transmitting.

C.3.5 Service Scheduling SM (SS-SM)

The general flowchart of the SS-SM is shown in Figure C.11. The SS-SM is activated at the beginning of each time slot. The SS-SM identifies a VC to receive cell transmission according to the Work-conserving WRR scheduling scheme. Then, the SS-SM obtains the cell at the head of the VC queue and transfers the cell in the BC_j of the BS to the XC_i indicated by XBP. Then, it changes the XS_i from '0' to '1' and advances the XBP to the subsequent entry in the XB. At the initiation of the system, the SS-SM starts its operation from the $XBEntry_1$. After these operations, the SS-SM activates the BA-SM and halts until the next time slot. The actual transmission will take place in the next time slot by the CX-SM.

Figure C.12 shows the algorithm of the WRR scheduling scheme. The Exhaus-

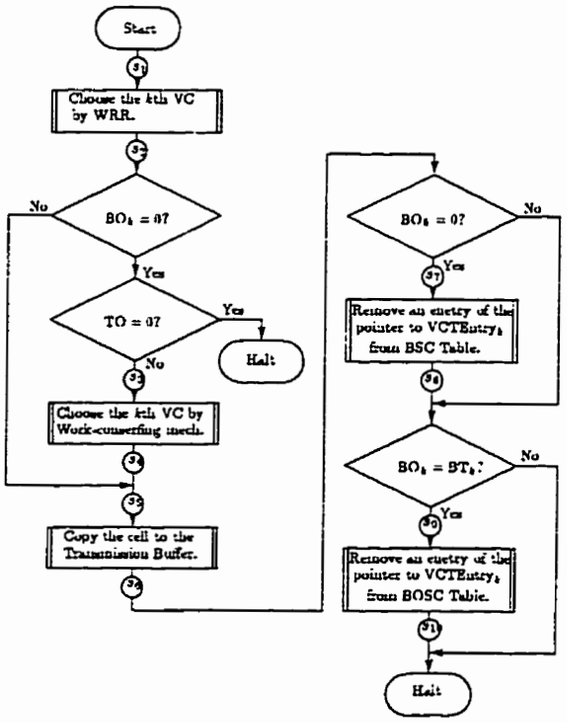

Figure C.11: The Service Scheduling SM (SS-SM)

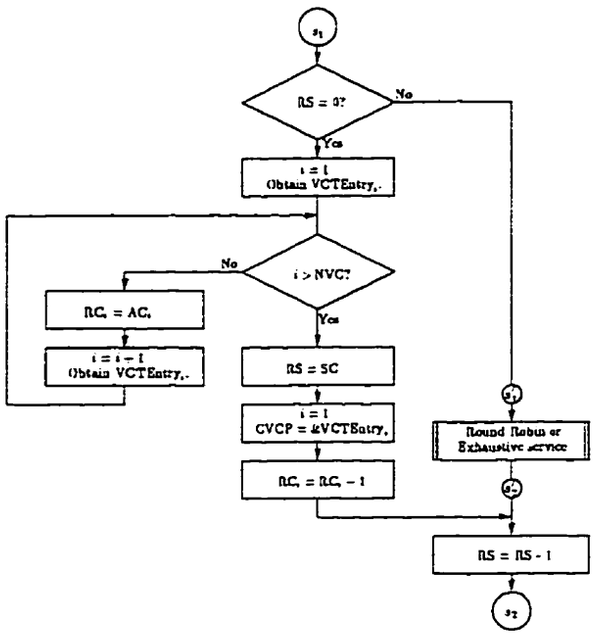

Figure C.12: The WRR scheduling scheme

tive and the Round Robin services are further specified in Figures C.13 and C.14, respectively. The extra complexity of the Round Robin service is the loop in the algorithm.

The time slot reassignment operation is shown in Figure C.15.

The selected cell is transferred to the XB by the algorithm in Figure C.16.

C.3.6 Buffer Allocation SM (BA-SM)

The general flowchart of the BA-SM is shown in Figure C.17. The BA-SM performs the cell admission according to the CSVP buffer allocation scheme. In each time slot, the BA-SM is activated by the SS-SM. The BA-SM examines the cell admission from the cell in the top entry of the PB and advances to the subsequent entries.

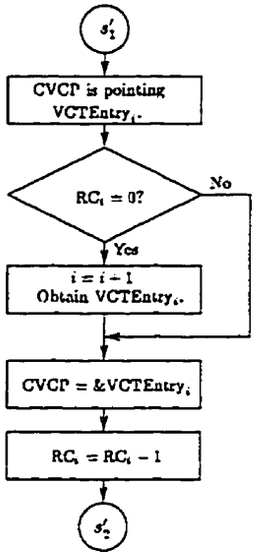

Figure C.13: The Exhaustive service

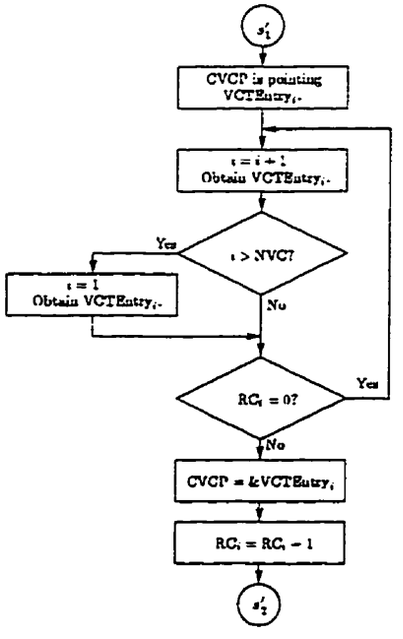

Figure C.14: The Round Robin service

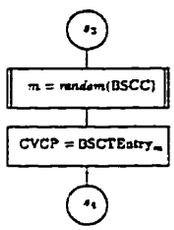

Figure C.15: The time slot reassingment operation

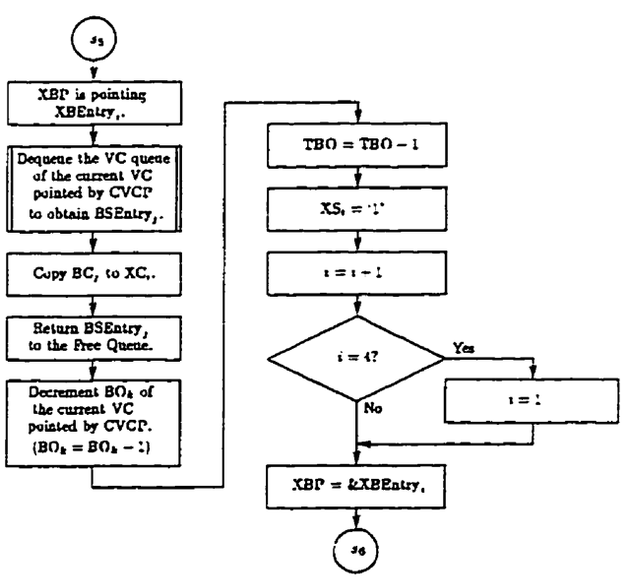

Figure C.16: The copy of a cell in the BS to the XB

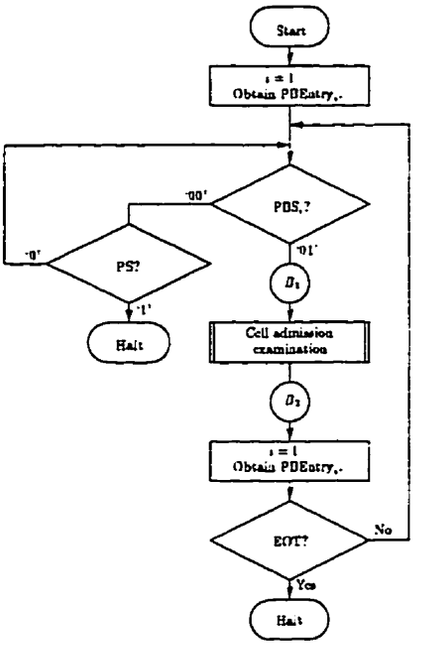

Figure C.17: The Buffer Allocation SM (BA-SM)

When the PBS_i field of the PB turns to '01', the BA-SM performs the cell admission examination of the cell in the PBC_i. The cell admission examination is shown in Figures C.18. Note that the first procedure in the flowchart, where matching of VPI-VCI pair is performed, requires at most K comparisons.

If the cell is admitted to an empty space, the BA-SM locates an available entry in the BS and stores the address of the $BSEntry_j$ to the BL_i field in the PB. Then, the BA-SM changes the PBS_i field from '01' to '10'. It also enqueues the $BSEntry_j$ in the VC queue. The cell admission to non-empty buffer is shown in Figure C.19. The cell may be admitted to a full buffer using the pushout operation shown in Figure C.20.

If the cell is blocked, the BA-SM changes the PBS_i from '01' to '11'. Then the

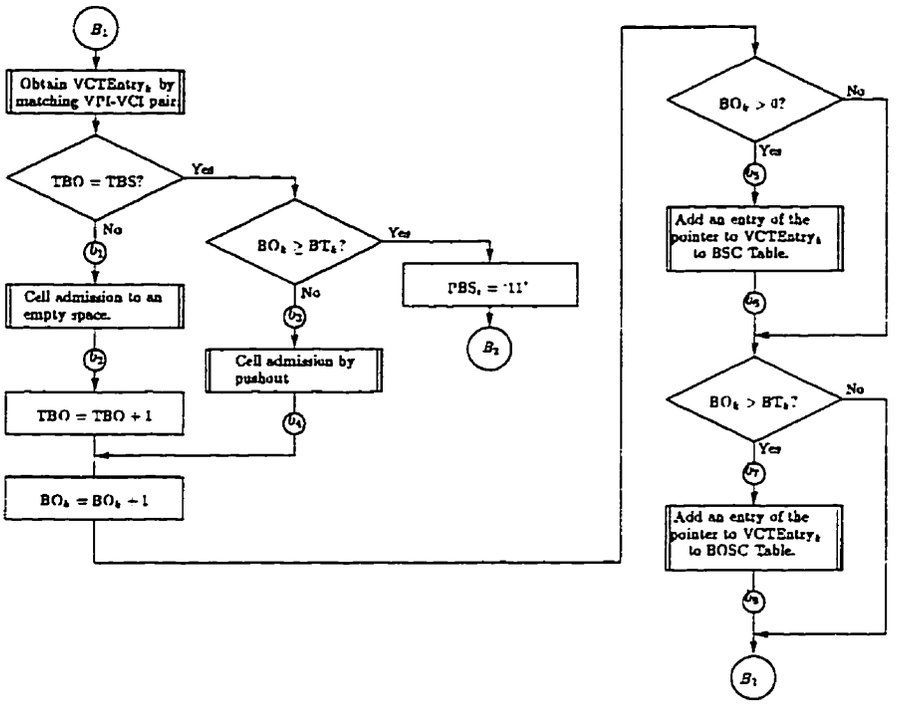

Figure C.18: The cell admission examination

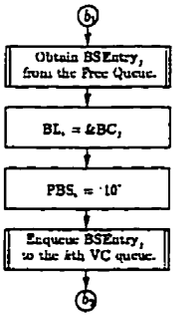

Figure C.19: The cell admission to an empty space

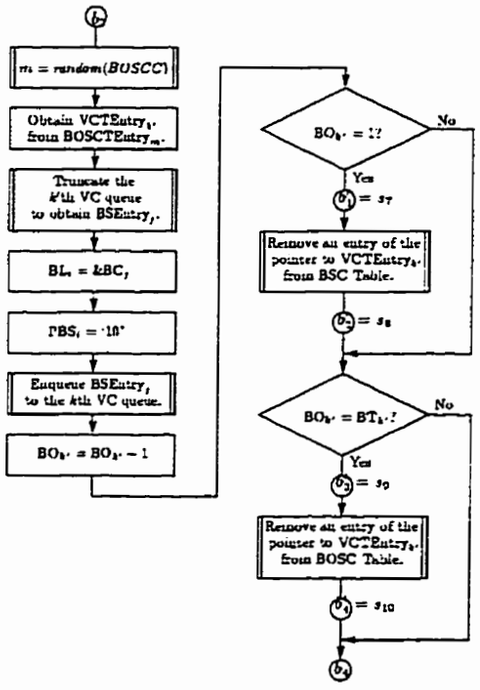

Figure C.20: The cell admission by the pushout operation

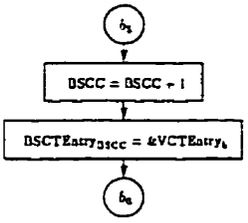

Figure C.21: Addition of an entry to the BSCT

BA-SM advances to the subsequent entry in the PB. If the PBS_i remains as '00' and the PS turns to '1', the BA-SM halts. The BA-SM also halts when it reaches the end of the PB.

C.3.7 Operations of the BSCT and the BOSCT

The operations to add an entry to the BSCT is shown in Figure C.21. And the operations to remove an entry from the BSCT is shown in Figure C.22. Since the structure and the size of an entry of the BOSCT segment are the same as those of the BSCT segment, the operations of the BSCT and the BOSCT are basically identical. The operations can be obtained by replacing "BSC" with "BOSC" in Figures C.21 and C.22. Note that the procedure to find "$\text{BSCTEntry}_m = \&\text{VCTEntry}_k$" or "$\text{BOSCTEntry}_m = \&\text{VCTEntry}_k$" requires at most K comparisons for the BSCT. and $K-1$ for the BOSCT.

C.4 Discussion on Memory Contention Problems

As already mentioned in the specification, the memory contention problem between the SS-SM and the CX-SM for the XB segment is solved by having three entries.

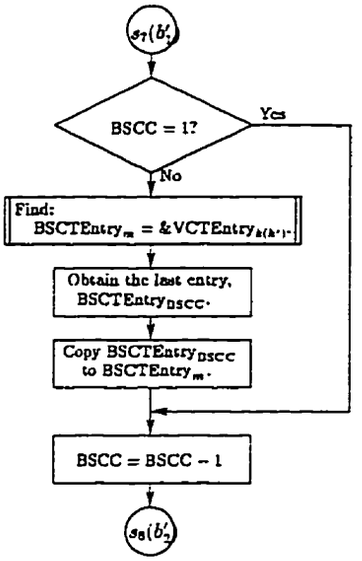

Figure C.22: Removal of an entry from the BSCT

Since the SS-SM and the BA-SM operate in a serial order, there is no memory contention problem between them. It is obvious that there is no memory contention problem between the BA-SM and the CB-SM at the BS because the CB-SM cannot copy a cell in the PB to the BS until the BA-SM finds a space for the cell, sets the address to the BL_i field, and changes the PBS_i field from '01' to '10'.

The only other potential memory contention problem is in the sharing of the PB segment among the PB-SM, the BA-SM, and the CB-SM. The starting of operations on an entry by these SMs will not cause any problem due to the state transition mechanism of the PBS_i field. The BA-SM cannot start the admission examination of a cell in the PB until the PB-SM changes the PBS_i field from '00' to '01'; and the CB-SM cannot start its operation on the cell until the BA-SM changes the PBS_i field from '01' to '10' or '11'. However, a problem may occur at the recognition

of the end of operation for the time slot. How can the BA-SM and the CB-SM detect that there is no more cells to be operated in the time slot and halt? For this purpose, the PS remnant is designed to notify that the PB-SM completed its operation for the time slot to the BA-SM and the CB-SM. Only when the PBS_i field of the next entry of the PB remains as '00' and the PS turns to '1', i.e., the PB-SM is halting, the BA-SM and the CB-SM terminate their operations in the time slot. It should be noted, however, that when all entries of the PB segment is filled by the PB-SM, there is no subsequent entry, but the BA-SM and the CB-SM may seek the subsequent entry and be unable to read the PBS_i field of the entry. Therefore, a mechanism for the BA-SM and the CB-SM to recognize the end of the PB is necessary.

Bibliography

[ACD+95] R. Arvind, G. L. Cash, D. L. Duttweiler, H-M. Hang, B. G. Haskell, and A. Puri. Image and video coding standards. In T. S. Rzeszewski, editor, *Digital Video*, pages 444–465. IEEE Press, New York, 1995.

[AD88] H. Ahmandi and W. E. Denzel. A servey of modern high-performance switching techniques. *IEEE Journal on Selected Areas in Communications*, 6(9):1528–1537, 1988.

[BDGP94] S. Bassi, M. Dècina, P. Giacomazzi, and A. Pattavina. Multistage Shuffle networks with shortest path and deflection routing for high performance ATM switching: the open-loop shuffleout. *IEEE Transaction on Communications*, 42(10):2881–2889, 1994.

[BZ96] J. C. R. Bennett and H. Zhang. WF^2Q: worst-case fair weighted fair queueing. In *Proceedings of IEEE INFOCOM '96*, pages 120–128, 1996.

[CCI89] CCITT. *CCITT Recommendation I.121, "Broadband Aspects of ISDN"*. Geneva, Switzwerland, 1989.

[CCI91a] CCITT. *CCITT Recommendation I.321, "B-ISDN Protocol Reference Model and Its Application"*. Geneva, Switzwerland, 1991.

[CCI91b] CCITT. *CCITT Recommendation I.361, "B-ISDN ATM Layer Specification"*. Geneva, Switzwerland, 1991.

[CCI91c] CCITT. *CCITT Recommendation I.362, "B-ISDN ATM Adaptaion Layer (AAL) Functional Description"*. Geneva, Switzwerland, 1991.

[CCI91d] CCITT. *CCITT Recommendation I.363, "B-ISDN ATM Adaptaion Layer (AAL) Specification"*. Geneva, Switzwerland, 1991.

[CGGK95] I. Cidon, L. Georgiadis, R. Guérin, and A. Khamisy. Oprimal buffer sharing. *IEEE Journal on Selected Areas in Comminucations*, 13(7):1229–1240, 1995.

[CGK94] I. Cidon, R. Guérin, and A. Khamisy. On protective buffer policies. *IEEE/ACM Transactions on Networking*, 2(3):240–246, 1994.

[Cha91a] H. J. Chao. A novel architecuture for queue management in the ATM network. *IEEE Journal on Selected Areas in Communications*, 9(7):1110–1118, 1991.

[Cha91b] H. J. Chao. Recursive modular terabit/second ATM switch. *IEEE Journal on Selected Areas in Communications*, 9(8):1161–1172, 1991.

[Che96] M. Cheung. Personal communication, 1996.

[CM93] D. X. Chen and J. W. Mark. SCOQ: a fast packet switch with shared concentration and output queueing. *IEEE/ACM Transactions on Networking*, 1(1):142–151, 1993.

[CM94] D. X. Chen and J. W. Mark. A buffer management scheme for the SCOQ switch under nonuniform traffic loading. *IEEE Transaction on Communications*, 42(10):2899–2907, 1994.

[CR87] L. P. Clare and I. Rubin. Performance sharing boundaries for prioritized multiplexing systems. *IEEE Tansactions on Information Theory*. IT-33(3):329–340, 1987.

[Cru91a] R. Cruz. A calculus for network delay, part I: Network elements is isolation. *IEEE Transaction on Information Theory*, 37(1):114–131. 1991.

[Cru91b] R. Cruz. A calculus for network delay, part II: Network analysis. *IEEE Transaction on Information Theory*, 37(1):132–141, 1991.

[CT94] C. G. Chang and H. H. Tan. Queueing analysis of explicit policy assignment push-out buffer sharing schemes for ATM networks. In *Proceedings of IEEE INFOCOM '94*, pages 500–509, 1994.

[CU95] H. J. Chao and N. Uzun. An ATM queue manager handling multiple delay and loss priorities. *IEEE/ACM Transactions on Networking*, 3(6):652–659, 1995.

[DCS88] M. Devault, J-Y. Cohennec, and M. Servel. The Prelude ATD experiment: assessment and future prospects. *IEEE Journal on Selected Areas in Communications*, 6(9):1528–1537, 1988.

[DGP94] M. Dècina, P. Giacomazzi, and A. Pattavina. Multistage Shuffle networks with shortest path and deflection routing for high performance ATM switching: the closed-loop shuffleout. *IEEE Transaction on Communications*, 42(11):3034–3044, 1994.

[DKS90] A. Demers, S. Keshav, and S. Shenker. Analysis and simulation of a fair queueing algorithm. *J. Internetworking Res. and Experience*, 1:3–26, 1990.

[DL86] J. N. Daigle and J. D. Langford. Models for analysis of packet voice communications systems. *IEEE Journal of Selected Areas of Communications*, SAC-4:847–855, 1986.

[DM95] M. D'Ambrosio and R. Melen. Evaluating the limit behavior of the ATM traffic within a network. *IEEE/ACM Transactions on Networking*, 3(6):832–841, 1995.

[EKO+93] N. Endo, T. Kozaki, T. Ohuchi, H. kuwahara, and S. Gohara. Shared buffer memory switch for an ATM exchange. *IEEE Transaction on Communications*, 41(1):237–245, 1993.

[EYH87] K. Y. Eng, Y. S. Yeh, and M. G. Hluchyj. A Knockout switch for variable length packets. *IEEE Journal on Selected Areas in Communications*, 5(9):1426–1435, 1987.

[Fis91] W. Fischer. A tandem queueing system with deterministic service. In *TELETRAFFIC AND DATATRAFFIC, ITC-13*, pages 563–569, 1991.

[For95] The ATM Forum. *ATM User-Network Interface Specification*. Version 3.1. Prentice-Hall, 1995.

[Fri91] V. J. Friesen. "A performance study of broadband networks". Master's thesis, University of Waterloo, Waterloo, Ontario, Canada, 1991.

[FV90] D. Ferrari and D. C. Verma. A scheme for real-time channel establishment in wide-area networks. *IEEE Journal on Selected Areas in Communications*, 8(3):368–379, 1990.

[Gal91] D. Le Gall. MPEG: A video compression standard for multimedia applications. *Communications of the ACM*, 34(4):47–58, 1991.

[GGPS96] L. Georgiadis, R. Guérin, V. Peris, and K. N. Sivarajan. Efficient network QoS provisioning based on per node traffic shaping. *IEEE Tansactions on Networking*, 4(4):482–501, 1996.

[GH93] M. Ghanbari and C. J. Hughes. Packet coded video signals into ATM cells. *IEEE/ACM Transactions on Networking*, 1(5):505–509, 1993.

[Gol90] S. J. Golestani. A stop-and-go queueing framework for congestion management. In *Proceedings of ACM SIGCOMM '90*, pages 8–18, 1990.

[Gol94] S. J. Golestani. A self-clocked fair queueing scheme for broadband applications. In *Proceedings of IEEE INFOCOM '94*, pages 636–646, 1994.

[Gol95] S. J. Golestani. Network delay analysis of a class of fair queueing algorithms. *IEEE Journal on Selected Areas in Communications*, 13(6):1057–1070, 1995.

[Gru91] R. Grunenfelder. A correlation based end-to-end cell queueing delay characterization in an ATM network. In *Queueing, Performance, and Control in ATM, ITC-13*, pages 59–64, 1991.

[GSL94] J. N. Giacopelli, W. D. Sincoskie, and M. Littlewood. Sunshine: a high-performance self-routing broadband packet switch architecture. In *Proceedings of ISS '90*, pages 123–129, 1994.

[Han89] R. Handel. Evolution of ISDN towards Broadband ISDN. *IEEE Network*, pages 7–13, 1989.

[Hay84] J. F. Hayes. *Modeling and Analysis of Computer Communications Networks*. Plenum Press, New York, 1984.

[HK88] M. G. Hluchyj and M. J. Karol. Queueing in high-performance packet switching. *IEEE Journal of Selected Areas of Communications*, 6(9):1587–1597, 1988.

[HL86] H. Heffes and D. M. Lucantoni. A Markov modulated characterization of packetized voice and data traffic related statistical multiplexer per-

formance. *IEEE Journal of Selected Areas of Communications*, SAC-4:856–868, 1986.

[Hui91] J. Y. Hui. Switching integrated broadband services by sort-banyan networks. *Proceedings of IEEE*, 79(2):145–154, 1991.

[ISO94] ISO. *ISO 1117-2: Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mbit/s*, 1994.

[ISO95] ISO. *ISO 13818-2: Generic Coding of Moving Pictures and Associated Audio Information: Video*, 1995.

[JU95] Y. C. Jung and C-K. Un. Banyan multipath self-routing ATM switches with shared buffer type switch elements. *IEEE Transaction on Communications*, 43(11):2847–2857, 1995.

[KHBG91] H. Kroner, G. Hébuterne, P. Boyer, and A. Gravey. Priority management in ATM switching nodes. *IEEE Journal on Selected Areas in Communications*, 9(3):418–427, 1991.

[KHM87] M .J. Karol, M. G. Hluchyj, and S. P. Morgan. Input versus output queueing on a space-division packet switch. *IEEE Transaction on Communications*, COM-35(12):1347–1356, 1987.

[Kim94] H. S. Kim. Design and performance of multinet switch: a multistage ATM switch architecture with partially shared buffers. *IEEE/ACM Transactions on Networking*, 2(6):571–580, 1994.

[KK77] H. Kobayashi and A. G. Konheim. Queueing models for computer communications system analysis. *IEEE Tansactions on Communications*. COM-25(1):2–29, 1977.

[KK80] L. Kamoun and L. Kleinrock. Analysis of shared finite storage in a computer network node environment under general traffic conditions. *IEEE Tansactions on Communications*, COM-28(7):992–1003, 1980.

[KKK90] C. R. Kalmanek, H. Kanakia, and S. Keshav. Rate controlled servers for very high-speed networks. In *Proceedings of IEEE GLOBECOM '90*, pages 12–20, 1990.

[KKM+97] C. R. Kalmanek, S. Keshav, W. T. Marshall, S. P. Morgan, and R. C. Restrick, III. Xunet 2: lessons from an early wide-area ATM testbed. *IEEE/ACM Transactions on Networking*, 5(1):40–55, 1997.

[KL95] Y. M. Kim and K. Y. Lee. KSMIN's: Knockout switch-based multistage interconnection networks for high-speed packet switching. *IEEE Transaction on Communications*, 43(8):2391–2398, 1995.

[Kle76] L. Kleinrock. *Queueing Systems, Volume II: Computer Applications*. Wiley-Interscience, New York, 1976.

[Kle91] L. Kleinrock. The path to broadband networks. *Proceedings of the IEEE*, 79(2):112–117, 1991.

[KLG90a] H. S. Kim and A. Leon-Garcia. A self-routing multistage switching network for broadband ISDN. *IEEE Journal on Selected Areas in Communications*, 8(3):459–466, 1990.

[KLG90b] H. S. Kim and A. Leon-Garcia. Performance of buffered banyan networks under nonuniform traffic patterns. *IEEE Transaction on Communications*, 38(5):648–658, 1990.

[KSC91] M. G. H. Katevenis, S. S. Sidiropoulos, and C. Courcoubetis. Weighted round-robin cell multiplexing in a general-purpose ATM switch chip. *IEEE Journal on Selected Areas in Communications*, 9(8):1265–1279, 1991.

[KTT+89] Y. Kida, S. Tsuzuki, K. Takayama, A. Natsume, N. Gotoh, I. Yoshida, and K. Kitajima. FDDI based 100 Mbps fiber-optic LAN. *Sumitomo Denki*, No.134:41–48, 1989. Sumitomo Electric Industries, Ltd., Osaka. Japan (in Japanese).

[Kur92] J. Kurose. On computing per-session performance bounds in high-speed multi-hop computer networks. *Performance Evaluation Review*. 20(1):128–139, 1992.

[KW94] G. Kesidis and J. Walrand. Conservation relations for fully shared ATM buffers. *Probability in theh Engineering and Information Sciences*. 8:147–151, 1994.

[LLG96] K. L. E. Law and A. Leon-Garcia. ATM multiplexers and output port controllers with distributed control and flexible queueing disciplines. In *Proceedings of IEEE INFOCOM '96*, pages 663–669, 1996.

[LS91] A. Y-M. Lin and J. A. Silvester. Priority queueing strategies and buffer allocation protocols for traffic control at an ATM integrated broadband switching system. *IEEE Journal on Selected Areas in Comminucations*, 9(9):1524–1536, 1991.

[LS93a] D-S. Lee and B. Sengupta. Queueing analysis of a threshold based priority scheme for ATM networks. *IEEE/ACM Transactions on Networking*, 1(6):709–717, 1993.

[LS93b] Y-M. Lin and J. A. Silvester. On the performance of an ATM switch with multichannel tranmission groups. *IEEE Transaction on Communications*, 41(5):760–770, 1993.

[LS96a] Y-S. Lin and C. B. Shung. Queue management for shared buffer and shared multi-buffer ATM switches. In *Proceedings of IEEE INFOCOM '96*, pages 688–695, 1996.

[LS96b] T-L. Ling and N. Shroff. Scheduling real-time traffic in ATM networks. In *Proceedings of IEEE INFOCOM '96*, pages 198–205, 1996.

[Min95] D. Minoli. *Video Dialtone Technology.* MacGraw-Hill, Inc., 1995.

[MLLK95] H. Y. Ming, T. Liu, K. Y. Lee, and Y. M. Kim. The Knockout switch under nonuniform traffic. *IEEE Transaction on Communications*, 43(6):2149–2156, 1995.

[MOSM90] M. Murata, Y. Oie, T. Suda, and H. Miyahara. Analysis of a discrete-time single-server queue with bursty inputs for traffic control in ATM networks. *IEEE Journal on Selected Areas in Communications*, 8(3):447–458, 1990.

[MSB97] W. Matragi, K. Sohraby, and C. Bisdikian. Jitter calculus in ATM networks: multiple nodes. *IEEE/ACM Transaction on Networking*, 5(1):122–133, 1997.

[MSH95] P. S. Min, H. Saidi, and M. V. Hegde. A nonblocking architecture for broadband multichannel switching. *IEEE/ACM Transactions on Networking*, 3(2):181–198, 1995.

[NGT+95] J. H. Naegle, S. A. Gossage, N. Testi, M. O. Vahle, and J. H. Maestas. Building networks for the wide and local areas using Asynchronous Transfer Mode switches and synchronous optical network technology. *IEEE Journal on Selected Areas in Communications*, 13(4):662–672, 1995.

[NLG96] H. Naser and A. Leon-Garcia. Emulation study of delay and delay variation in ATM networks, part 1: CBR traffic. In *Proceedings of IEEE INFOCOM '96*, pages 393–400, 1996.

[NTFH87] S. Nojima, E. Tsutsui, H. Fukuda, and M. Hashimoto. Iintegrated services packet network using Bus Matrix switch. *IEEE Journal on Selected Areas in Communicavions*, 5(8):1284–1291, 1987.

[OMM91] Y. Ohba, M. Murata, and H. Miyahara. Analysis of interdeparture processes for bursty traffic in ATM networks. *IEEE Journal of Selected Areas in Communications*, 9(3):468–476, 1991.

[PF91] D. W. Petr and V. S. Frost. Nested threshold cell discarding for ATM overload control: optimization under cell loss constraints. In *Proceedings of IEEE INFOCOM '91*, pages 1403–1412, 1991.

[PG93] 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 Tansactions on Networking*, 1(3):344–357, 1993.

[PG94] A. K. Parekh and R. G. Gallager. A generalized processor sharing approach to flow control in Integrated Services Networks: the multiple-node case. *IEEE/ACM Tansactions on Networking*, 2(2):137–150, 1994.

[Pre97] B. R. Preiss. Personal communication, 1997.

[PZ94] P. Pancha and M. El Zarki. MPEG coding for variable bit rate video transmission. *IEEE Communications Magazine*, 32(5):54–66, 1994.

[RMW94] J. F. Ren, J. W. Mark, and J. W. Wong. End-to-end performance in ATM networks. In *Proceedings of IEEE ICC '94*, pages 996–1002, 1994.

[RRA95] S. Rampal, D. S. Reeves, and D. P. Agrawal. An approach towards end-to-end QoS with statistical multiplexing in ATM networks. Technical Report TR 95/2, Center for Communications and Signal Processing, North Carolina State University, Raleigh, 1995.

[SBPD93] B. Steyaert, H. Bruneel, G. H. Petit, and E. Desmet. End-to-end delays in multistage ATM switching networks: approximate analytic derivation of tail probabilities. *Computer Networks and ISDN Systems*, 25:1227–1241, 1993.

[SMT96] D. Saha, S. Mukherjee, and S. K. Tripathi. Carry-over round robin: a simple cell scheduling mechanism for ATM networks. In *Proceedings of IEEE INFOCOM '96*, pages 630–637, 1996.

[Sta91] I. Stavrakakis. Efficient modeling of merging and splitting processes in large networking structures. *IEEE Journal on Selected Areas in Communications*, 9(8):1336–1347, 1991.

[SV96] D. Stiliadis and A. Varma. Latency-rate servers: a general model for analysis of traffic scheduling algorithms. In *Proceedings of IEEE INFOCOM '96*, pages 111–119, 1996.

[SW86] K. Sriram and W. Whitt. Characterizing superposition arrival process in packet multiplexer for voice and data. *IEEE Journal of Selected Areas in Communications*, SAC-4(6):833–846, 1986.

[THP94] L. Tassiulas, Y. C. Hung, and S. P. Panwar. Optimal buffer control during congestion in an ATM network node. *IEEE/ACM Transactions on Networking*, 2(4):374–386, 1994.

[Tob90] F. Tobagi. Fast packet switch architectures for Broadband Integrated Services Digital Network. *Proceedings of the IEEE*, 78(1):133–167, 1990.

[Tur86] J. S. Turner. Dsign of an integrated services packet network. *IEEE Journal on Selected Areas in Communications*, 4(8):1373–1379, 1986.

[Tur90] A. Turner. A multistage high-performance packet switch for broadband networks. *IEEE Transaction on Communications*, 38(9):1607–1615, 1990.

[TYN+87] T. Takeuchi, T. Yamaguchi, H. Niwa, H. Suzuki, and S. Hayano. Synchronous composite packet switching—a switching architecture for broadband ISDN. *IEEE Journal on Selected Areas in Communications*, 5(8):1365–1376, 1987.

[UAH+94] T. Urabe, H. Afzal, G. Ho, P Pancha, and M. El. Zarki. MPEGTool: An X Window based MPEG endoder and statistics tool. *Multimedia Systems*, 1, 1994.

[Vis96] M. Vishnu. *Design of a Versatile ATM Switching Node with a Per-VC Architecture*. PhD thesis, University of Waterloo, Waterloo, Ontario, Canada, 1996.

[Vit86] A. M. Viterbi. Approximate analysis of time-synchronous packet netwoks. *IEEE Journal of Selected Areas in Communications*, SAC-4(6):879–890, 1986.

[VM96] M. Vishnu and J. W. Mark. HOL-EDD: a flexible service scheduling scheme for ATM networks. In *Proceedings of IEEE INFOCOM '96*. pages 647–654, 1996.

[VZF91] D. Verma, H. Zhang, and D. Ferrari. Guaranteeing delay jitter bounds in packet switching netwroks. In *Proceedings of Tricomm '91*, pages 35–46, 1991.

[WLG94] I. Widjaja and A. Leon-Garcia. The Helical switch: a multipath ATM switch which preserves cell sequence. *IEEE Transaction on Communications*, 42(8):2618–2629, 1994.

[WM95] G-L. Wu and J. W. Mark. A buffer allocation scheme for ATM networks: Complete Sharing based on Virtual Partition. *IEEE/ACM Transactions on Networking*, 3(6):660–670, 1995.

[WY95] P. C. Wong and M. S. Yeung. Design and analysis of a novel fast packet switch—Pipeline banyan. *IEEE/ACM Transactions on Networking*, 3(1):63–69, 1995.

[YHA87] Y. S. Yeh, M. G. Hluchyj, and A. S. Acampora. The Knockout switch: a simple, modular architecture for high-performance packet switching.

IEEE Journal on Selected Areas in Communications, 5(8):1274–1282, 1987.

[YKF91] T. Yokoi, N Kishimoto, and Y. Fujii. ATM network performance evaluation using parallel and distributed simulation technique. In *TELETRAFFIC AND DATATRAFFIC, ITC-13*, pages 815–820, 1991.

[YM96] A. Yamada and J. W. Mark. Emulation study of a resource allocation scheme at an ATM switch node. In *Proceedings of the 1996 Conference on Information Science And Systems*, pages 457–462, 1996.

[YS93] O. Yaron and M. Sidi. Calculation performance bounds in communication networks. In *Proceedings of IEEE INFOCOM '93*, pages 539–546, 1993.

[ZF93] H. Zhang and D. Ferrari. Rate-controlled static priority queueing. In *Proceedings of IEEE INFOCOM '93*, pages 227–236, 1993.

[ZF94] H. Zhang and D. Ferrari. Rate-controlled service disciplines. *Journal of High Speed Networks*, 3(4):389–412, 1994.

[Zha91] L. Zhang. VirtualClock: a new traffic control algorithm for packet-switched networks. *ACM Transactions on Computer Systems*, 9(2):101–124, 1991.

[Zha95] H. Zhang. Service disciplines for guaranteed performance service in packet-switching networks. *Proceedings of the IEEE*, 83(10):1374–1396, 1995.

[ZS94] Q. Zheng and K. Shin. On the ability of establishing real-time channels in point-to-point packet-switching networks. *IEEE Transactions on Communications*, 42:1096–1105, 1994.