CONTROL STRUCTURES AND TECHNIQUES FOR BROADBAND-ISDN COMMUNICATION SYSTEMS

Andreas Pitsillides

School of Electrical Engineering

Swinburne University of Technology,
Melbourne, Australia.

Thesis Submitted for

the Degree of Doctor of Philosophy

JUNE 1993

ABSTRACT

A structured organisation of tasks, possibly hierarchical, is necessary in a BISDN network due to the complexity of the system, its large dimension and its physical distribution in space. Feedback (possibly supplemented by feedfonvard) control has an essential role in the effective and efficient control of BISDN. Additionally, due to the nonstationarity of the network and its complexity, a number of different (dynamic) modelling techniques are required at each level of the hierarchy. Also, to increase the efficiency of the network and allow flexibility in the control actions (by extending the control horizon) the (dynamic) tradeoff between service-rate, buffer-space, cell-delay and cell-loss must be exploited. In this thesis we take account of the above and solve three essential control problems, required for the effective control of BISDN. These solutions are suitable for both stationary and nonstationary conditions. Also, they are suitable for implementation in a decentralised coordinated form, that can form a part of a hierarchical organisation of control tasks. Thus, the control schemes aim for global solutions, yet they are not limited by the propagation delay, which can be high in comparison to the dynamics of the controlled events.

Specifically, novel control approaches to the problems of Connection Admission Control (CAC), flow control and service-rate control are developed. We make use of adaptive feedback and adaptive feedfonvard control methodologies to solve the combined CAC and flow control problem. Using a novel control concept, based on only two groups of traffic (the controllable and uncontrollable group) we formulate a problem aimed at high (unity) utilisation of resources while maintaining quality of service at prescribed levels. Using certain assumptions we have proven that in the long term the regulator is stable and that it converges to zero regulation error. Bounds on operating conditions are also derived, and using simulation we show that high utilisation can be achieved as suggested by the theory, together with robustness for unforeseen traffic connections and disconnections. Even with such a high efficiency and strong properties on the quality of service provided, the only traffic descriptor required from the user is that of the peak rate of the uncontrollable traffic.

A novel scheme for the dynamic control of service-rate is formulated, using feedback from the network queues. We use a unified dynamic fluid flow equation to describe the virtual path (VP) and hence formulate two illustrative examples for the control of service-rate (at the VP level). One is a nonlinear optimal multilevel implementation, that features a coordinated decentralised solution. The other is a single level implementation

that turns out to be computationally complex. Therefore, for the single level implementation the costate equilibrium solution is also derived. For the optimal policies derived, we discuss their implementation complexity and provide implementable solutions. Their performance is evaluated using simulation. Additionally, using an ad hoc approach we have extended previous published works on the decentralised coordinated control of large scale nonlinear systems to also deal with time-delayed systems.

Using a hierarchical structure we demonstrate the derivation of a particular solution for the control of service-rate. In particular we decompose the system, both vertically and horizontally. We provide local coordinated decentralised solutions at the lower levels of the hierarchy and more global solutions operating at slower time scales at the higher levels. At the highest level of the hierarchy, we extend published results on the control of service-rate to the case of a multiobjective formulation. At the lowest level of the hierarchy, we propose a novel link server protocol derived from heuristic arguments. At the intermediate level we use the dynamic service-rate control scheme, described above. Also, to demonstrate the flexibility and adaptability of the hierarchical structure an additional level has been formulated.

In this thesis, adaptive control theory and multilevel optimal control theory are applied to a variety of BISDN problem formulations, which demonstrate the suitability and power of these techniques in the BISDN context.

In summary, we offer an integrated structured approach to the effective control of BISDN that has the essential features of implementability, efficiency, effectiveness and robustness.

ACKNOWLEDGEMENT

First of all I would like to thank my wife Barbara and children Stacey and Michael without whose whole-hearted support this research would not have been completed.

Next, I would like to thank my thesis advisers Jim Lambert and Cheng Siong Lee of Swinburne University of Technology for their valuable support and encouragement. In particular many thanks go to Dr Lambert for his insightful discussions and invaluable criticism. Similarly, I would like to thank Dr Lee for many useful suggestions for improvement of this thesis presentation as well as Dr Zurawski for helpful discussions during the early phase of this research study.

My thanks also go to Bob Warfield, Meir Herzberg, Leith Campbell, Sammy Chan, Moshe Zukerman, John Burgin, Stephen Bye and others at the Telecom Australia Research Laboratories. I would like to especially thank Dr Herzberg and Dr Warfield for the joint work. In particular Dr Warfield has been a supportive and incisive critic, drawing attention to many matters which I would otherwise have overlooked.

I would also like to thank the Telecom Australia Research Laboratories. in particular Dr Campbell, for the six months spent there as an academic visitor.

Also my thanks go to 30 Steiner and Nian Li from the Mathematics department of Swinburne University of Technology. In particular I wish to thank Nian Li for several useful discussions relating to some of the proofs.

Finally I would like to thank Swinburne University of Technology, for the sabbatical leave of one year duration, and my colleagues at the School of Electrical Engineering.

PUBLICATIONS RELATED TO THE WORM PRESENTED IN THIS THESIS

Reports

i) **A.** Pitsillides, "A multilevel/multilayer control theoretic approach to the control of Broadband-ISDN", Internal report LTRO192, Laboratory for Telecommunication Research, Swinburne University of Technology, January 1992.

ii) A. Pitsillides, "Connection admission control (CAC)", Internal Report LTR0492, Laboratory for Telecommunication Research, Swinburne University of Technology, July 1992.

iii) A. Pitsillides, "Report on the six month visit to Telecom Australia Research Laboratories, 6th March-August 31st 1992", Report LTR0692, Laboratory for Telecommunication Research, Swinburne University of Technology, September 1992.

vi) A. Pitsillides, "Adaptive (feedback and feedforward) combined CAC and flow control for BISDN featuring high utilisation and bounded QoS performance", internal working paper, Laboratory for Telecommunication Research, Swinburne University of Technology, October 1992.

Papers presented (or accepted for presentation)

v) A. Pitsillides, J. Lambert, N. Li, J. Steiner, "Dynamic Bandwidth Allocation in Communication Systems: An Optimal Control Approach", IEEE International Conference on Systems Engineering, Kobe, Japan, September 1992.

vi) **A.** Pitsillides, J. Lambert, "**An** Optimal Control Approach to Dynamic Bandwidth Allocation in B-ISDN", Communications'92, Sydney, October 1992.

vii) A. Pitsillides, J. Lambert, "**A** Multilevel Control Theoretic Approach to Dynamic Bandwidth Allocation in Broadband-ISDN", TENCON'92, Melbourne, November, 1992.

viii) **A.** Pitsillides, J. Lambert, B. Warfield, "A structure for the control of BISDN", ATRS'92, Adelaide, November, 1992.

ix) M. Herzberg, A. Pitsillides, "**A** hierarchical approach for the bandwidth allocation, management and control in B-ISDN", ICC '93, Geneva, May 1993.

x) A. Pitsillides, J. Lambert and B. Warfield, "A hierarchical control approach for Broadband-ISDN communication networks", 12th Triennial IFAC World Congress, Sydney, July 1993.

xi) A. Pitsillides, J. Lambert, "High throughput with controlled quality of service in Broadband-ISDN", ABSSS'93, Wollongong, Australia, July 1993.

Papers submitted (pending acceptance)

xii) A. Pitsillides, J. Lambert, "Integrated adaptive connection admission and flow control (ACFC): controlled QoS yet high utilisation", submitted to GLOBECOM'93, Houston, Texas, November 1993.

xii) A. Pitsillides, J. Lambert, "Adaptive congestion control in Broadband-ISDN: High throughput with sustained quality of service", submitted to International Conference on Telecommunications, ICT'94, Dubai, January 1994.

xiv) A. Pitsillides, J. Lambert, "Adaptive control techniques for guaranteed quality of service and high throughput in Broadband-ISDN", submitted to ITC 14, Antibes Juan-les-Pins, France, June 1994.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

ACFC	Adaptive Connection admission and Flow Control
AR	Auto Regressive
ARMA	Auto Regressive Moving Average
ARMAX	Autoregressive Moving Average with exogenous input
ATM	Asynchronous Transfer Mode
BISDN	Broadband-ISDN
CAC	Connection Admission Control
CARMA	Controlled Autoregressive Moving Average (also known as ARMAX)
CARIMA	Controlled Autoregressive Integrated Moving Average
CCITT	International Telegraph and Telephone Consultative Committee
CQ	Cyclic Queue
CSTS	Continuous System Type Simulation
DE	Discrete Event
DES	Discrete Event System
DENS	Discrete Event Nonstationary Simulation
DSCT	Discrete-State Continuous-Time
FIFO	First In First Out
FIR	Finite Impulse Response
GMV	Generalised Minimum Variance
GPC	Generalised Predictive Control
ISDN	Integrated Services Digital Network
LRPC	Long Range Predictive Control
LSP	Link Server Protocol
LQ	Linear Quadratic
LQG	Linear Quadratic Gaussian
LU	Local Unit
MA	Moving Average
MIMO	Multiple Input Multiple Output
MV	Minimum Variance
NPC	Network Parameter Control
OD	Origin Destination
OS	Overall Supremal unit
QoS	Quality of Service
RLS	Recursive Least Squares

SISO	Single Input Single Output
SIMO	Single Input Multiple Output
SU	Supremal Unit
TPBV	Two Point Boundary Value
UPC	Usage Parameter Control
VC	Virtual Channel
VP	Virtual Path
VPAM	Virtual Path Allocation and Management
VPC	Virtual Path Control
VPOSU	Virtual Path Overall Supremal Unit

Kendall notation for Queues

In 1952, D. G. Kendall described a shorthand notation that captures the essential characteristics of queuing systems, as follows:

Inter amval distribution	/ Service time distribution	/ Number of servers	/ {queue places, queue discipline)

where (commonly) the following letters are used

M-Markov D-Deterministic G-General N-Number

Conferences and journal abbreviations:

ABSSS	Australian Broadband Switching and Services Symposium
ACM	Association for Computing Machinery
ATRS	Australian Teletraffic Research Seminar
CDC	Conference on Decision and Control
GLOBECOM	Global Telecommunications Conference
ICC	International Conference on Communications
ICT	International Conference on Telecommunications
IEE	Institution of the Electrical Engineers
IEEE	Institute of the Electrical and Electronic Engineering
IFAC	International Federation of Automatic Control
IFIP	International Federation of Information Processing
INFOCOM	The Conference on Computer Communications
ITC	International Teletraffic Congress
JSAC	Journal on Selected Areas in Communications
NOMS	Network Operations and Management Symposium
TENCON	Computers, Communication and Automation toward the 21st Century

GLOSSARY

In this glossary, we list some mathematical symbols and the common meaning of the main variables used in the thesis. A detailed explanation of their meaning, can be found within the text. An example of the variable's usage appears in the page listed under the page heading. Note that commonly capital letters are used for matrices, and small letters for vectors.

symbol	description	units	page
q^{-1}	shift or delay operator		38
$\alpha(q^{-1})$	polynomial in the shift operator q^{-1}		69
z	z-transform operator		48
Δ	differencing operator		38
Au	incremental form of variable		38
$\hat{x}$	estimate of x		70
$\tilde{x}$	error between actual and estimated value of x		83
$E\{x\}$	mathematical expectation of x		79
$\{x\}$	sequence of x		70
$\lceil x \rceil$	is the next integer toward $+\infty$ (round up).		21
$\lfloor x \rfloor$	is the next integer toward $-\infty$ (round down).		33
$erf(x)$	error function of x		80
$erfc(x)$	complementary error function of x		80
A^T	transpose of matrix (or vector) **A**		69
$\otimes$	Kronecker product		128
0^n	a n x 1 vector of zeros		128
$0^{n\times m}$	a n x m matrix of zeros		128
e^n	a n x 1 vector of ones		128
$e^{n\times m}$	a n x m matrix of ones		128
∞^n	a n x 1 vector of infinities		128
α, β, γ	Lagrange Multipliers		170

ε	noise		69
θ	parameter vector		69
λ	arrival or flow of cells (cell-flow)	cells/time unit	22
λ_o	control penalty factor		74
λ_f	forgetting factor		75
ν	costate vector		170
ξ	coordination variable		200
π	Lagrange Multiplier		170
ρ	throughput		99
σ^2	variance of variable		79
τ	time delay	time units	128
φ	past data vector		69
B, b	buffer size	cells	22
C, c	service-rate, bandwidth, capacity	cellsltime unit	128
e	white noise		70
$f(x)$	general function of x		21
$G(x)$	general function of x		128
h	peak cell-rate	cells/time unit	44
$\mathcal{H}$	Harniltonian function		170
i, j	general indexing variables		38
J	cost function, objective function, etc...		148
k	discrete time delay (in integer multiples of the sampling time)		71
$\mathcal{L}$	Lagrangian function		199
m	mean value of variable		99
n	order of system		38
P_{loss}	probability of loss		22
$p\%$	pth percentile of the buffer distribution		79
p	pole location		49
t	time (continuous or discrete)	time unit	37
T	sampling rate	per time unit	53
u	control input		37

v	feedforward signal		37
v_1	feedforward signal 1; the average (over one control interval) total uncontrollable traffic flow	cells/control interval	52
v_2	feedforward signal 2; the average (over one control interval) of the cell queue length	cell places/ control interval	52
w	tradeoff weights		148
x	system state		128
y	system output		37
z	interaction variable		199

LIST OF ILLUSTRATIONS

CHAPTER 1

INTRODUCTION

The increasing needs of an information based society has imposed tremendous pressures on our telecommunication system to carry, in addition to voice, a diverse mix of information types. This has led to the development of a set of proposals for a telecommunication system, to carry these diverse information units in an economic fashion. The proposed telecommunication network is known as Broadband-ISDN (BISDN).

The development of this type of network has induced significant research effort. The principal subject of this thesis, is the development of control structures and techniques to improve the performance of BISDN networks. At the outset it should be noted that although this work focuses on BISDN, many of the conclusions are relevant to other high speed network technologies such as high-speed packet switched computer networks (e.g. IBM's plaNET) and Asynchronous Time Sharing (ATS) based broadband networks.

1.1 BISDN in perspective

BISDN [1] is the high bandwidth multimedia telecommunication network, proposed by CCITT[#1], to incorporate broadband features into the Integrated Services Digital Network (ISDN). It is tailored to become the universal future network, scheduled for implementation by telecommunication authorities worldwide within this decade. This network will need to handle a variety of types of service, with diverse demands on the network in terms of the bit rate and burstiness required. Continuous as well as variable bit rates will be serviced, e.g. data, voice, still and moving pictures, and multimedia applications (see for example tables 2.1 to 2.5 in [2] adapted from [3]). Asynchronous Transfer Mode (ATM)–a fusion between packet switching techniques and synchronous time division multiplexing [4]–will be the transfer mode for implementing BISDN, as already agreed by CCITT [1]. Information streams are divided into fixed length, self routing packets, commonly referred to as cells[#2], which are directed through the network by fast hardware switches. The bandwidth allocated to a connection may vary over the lifetime of the connection, hence ATM offers multiplexing and buffering within

#1 CCITT–International Telegraph and Telephone Consultative Committee–is the international body responsible for setting standards for public telecommunication.

#2 Cells are 53 bytes long, with 5 bytes assigned for the cell header and the remaining 48 bytes assigned for the information field of the cell.

the network to allow more effective use of the resources. However buffering and multiplexing leads to cell-delay and (possibly) cell-loss. At the same time, network users require guaranteed levels of performance.

These requirements, coupled with the wide range of traffic characteristics and quality of service constraints, as well as the geographic distribution and large dimension of the network, lead to some substantial problems in the control of BISDN networks.

BISDN has been exrensively researched (especially during the past 2-3 years), as evidenced by the large body of published papers, see for example: the proceedings of INFOCOM, GLOBECOM, ICC, ITC, ATRS, ABSSS, to name but a few; the journals devoting whole issues to BISDN [5], [6], [7], [8]; the books published [2], [9], [10], [11]; and the large number of CCITT recommendations–a listing of CCITT recommendations for BISDN appears in [2]. Even with this large body of published works there are still substantial unresolved problems of control in BISDN networks. See, for example: the guest editorial comments in [12] (they state that: "The international telecommunications community fully appreciates the complexity of the issue and, to cope with this problem, proposed a large variety of congestion control techniques. Many researchers believe that there is no silver bullet and that control of high-speed packet networks can be obtained by executing several concurrent mechanisms..."); [13] the CCITT recommendations on traffic control and congestion control; Jain [14], and Boyer [15], for an enlightening discussion; and [16] the guest editorial comments of the JSAC special issue on congestion issues in BISDN for a brief discussion of some of the control difficulties (they state that: "the dynamic, heterogeneous, time-varying network environment, with different service requirements is a significant factor in the design of controls"; and that "the design of the entire system and the interaction of the various components is often more important than the optimisation of individual components").

1.2 Motivation for our approach

In this thesis, we focus on a structured approach for the solution of the complex control problem of BISDN discussed above. Initially we motivate an appropriate structure that can formally address complexity, and then offer specific solutions in the key generic functions areas of CCITT [13] – flow control, call admission control and service-rate[#1] control[#2]. We have been guided by the general objectives of traffic control and congestion control in BISDN as defined in [13]: "to protect the network and the user in order to achieve network performance objectives, with an additional role of network

[#1] Cell-service-rate, bandwidth and capacity are equivalent terms in this context; for consistency we will use service-rate throughout this thesis.
[#2] Control is used as the generic word for management, control, and allocation (see footnote on page 25 for a loose definition of these terms (as used by telecommunication theoreticians).

resource optimisation". However, we demonstrate that not only can high resource utilisation be obtained, but also very tight control of the network performance can be achieved by using appropriate problem formulations and solutions.

Additionally, as it is increasingly noted, communication networks must have satisfactory dynamic as well as steady-state performance (e.g. Van As [17], Tipper et al [18], Lovegrove et al [19], Bolot at al [20], the session on dynamic phenomena at GLOBECOM'92). Nonstationary conditions occur in communication networks when the statistics of the traffic arrival processes or queue service processes vary with time (for example due to nonstationary input loads, topological changes to the network or failures of network resources). This nonstationary behaviour is particularly significant in the context of BISDN networks because of the mix of traffic types and the nature of resource sharing. Therefore, the dynamic aspects of network behaviour cannot be ignored.

Moreover, we deviate from the norm and assume that the user is <u>not</u> able to declare all traffic characteristics of the offered traffic in advance, i.e. at the call setup. Therefore we do not base our control decisions on any user declared parameters[#1] (apart from the peak rate of a special group of traffic, which we define as controllable[#2]). Our assumption stems from the fact that in the majority of cases not all the characteristics of real-time traffic (as for example the mean value, the burstiness factor, etc) are known in advance. It is stated [15] that in most cases the peak rate is the only traffic parameter that the users are able to declare at the call set-up. Notwithstanding the above argument, even if the statistical parameters are reasonably well known, there are still substantial problems in using these traffic descriptors for network control in an open loop fashion. (For example enforcement issues[#3]; also see [21] in which, using a real video sequence of 30 minute duration, they demonstrate that different sources with identical statistical characteristics can experience cell-loss rates that differ from each other by several orders of magnitude.) Also, on a more philosophical note, the flexibility of transmitting information on demand, in real time, is taken away from the user [15].

So, our control philosophy is principally guided by the following key assertions (elaborated at the relevant parts of the thesis):

- a structured approach is necessary due to the complexity of the system (see discussion in chapter 2).
- the dynamic aspects of network behaviour cannot be ignored.
- the use of formal control theory is worthy of investigation. We also allege that feedback, possibly supplemented with feedfonvard control, is an essential

[#1] A standardised traffic descriptor is often discussed in the literature, in which a set of standard traffic parameters is available that <u>completely</u> characterises the behaviour of a traffic source.
[#2] The definition of controllable and uncontrollable traffic appears in section 3.2.1, page 31
[#3] References abound in section 3.1.1 page 26.

component for an effective and efficient control system (e.g. see discussion in section 2.2.3).

- there is no need for user declared parameters, except for the peak rate which is easily enforceable at the ATM layer Service Access Point (e.g. see discussion in section 3.1).

1.3 Overview of the thesis

The details of the organisation of the thesis are as follows:
The thesis is divided into six main chapters, and one appendix. The structuring of this work is slightly non-conventional. We have split the introduction between this chapter and the next, in order to highlight (and justify) our control philosophy. Additionally, we present the literature reviews and local appendices within the chapters that they best relate to. Therefore we present the table below as a guide mapping the "traditional" chapter classifications with those used here.

	Introduction	Literature survey	Theory	Results and their discussion	Conclusion	Other
Chapter 1	✓	✓				
Chapter 2	✓	✓	✓			
Chapter 3		✓	✓	✓		
Chapter 4		✓	✓	✓		
Chapter 5		✓	✓	✓		
Chapter 6					✓	
Appendix A						✓

Table 1.1. Thesis guide

An overview of the rest of the chapters in the thesis follows.

In chapter 2 we show that a hierarchical organisation of tasks is necessary in a BISDN network due to the complexity of the system, that is, because of its large dimension and physical distribution in space with different event time scales ranging over several orders of magnitude. We propose a novel hierarchical structure based on the system behaviour in both time and space. We also argue that, for the effective and efficient control of BISDN: feedback, supplemented by feedforward control, must be considered; various dynamic modelling techniques are essential; and that the tradeoff between service-rate, buffer-space, cell-loss and cell-delay cannot be ignored.

In chapter 3 we solve the flow control and call admission control problems together, thus taking into account the interaction between them. We aim to use a feedback control system to maintain the Quality of Service (QoS) close to a target value, irrespective of variations in traffic (the disturbance). We define two distinct groups of traffic (controllable and uncontrollable), which allows us to introduce the concept of network controllability. Controllability is achieved by simply bounding the uncontrollable traffic. Bounding of the uncontrollable traffic becomes the sole role of CAC. The feedback signal is derived from a network performance monitor (which predicts the pth percentile of the buffer distribution). The controller regulates QoS by manipulating the flow of controllable traffic into the network. Controllability guarantees that the network can be operated efficiently (theoretically at 100% utilisation) and still provide the user with tightly regulated QoS (set at any desired target value). We employ the general methodology of adaptive control (featuring both adaptive feedback and adaptive feedforward) to solve the difficult problems of CAC and flow control together. By using this approach we are able to do away with the restrictive requirements of existing schemes (for example the need to declare a complex traffic descriptor at the call connection request). The performance of the derived algorithm is illustrated via analysis and simulation. The presented solution is in a form suitable for incorporation in a hierarchically organised control structure.

In Chapter 4 we focus on service-rate control at one level only—that of the virtual path (VP) level. We present a novel scheme that dynamically allocates service-rate by using the state of the buffers in the network as a feedback signal. In particular, we use a dynamic fluid flow type equation to model the VP and formulate precise problems for the control of service-rate. The interactions within the nodes spanned by a VP, as well as the interactions between the VPs sharing a link, are addressed in the problem formulation. We investigate the use of optimal single level and multilevel control theory concepts (featuring decomposition and coordination) in the solution of the service-rate control problem. The performance of the scheme is investigated using simulation. The form of the solution is suitable for incorporation in a broader, hierarchically organised, control structure. We also use an ad hoc approach to extend published results on the coordinated decentralised control of large scale nonlinear systems to deal with time-delayed systems.

Chapter 5 presents an illustrative example of a hierarchically organised control structure (that can form a part of the overall solution) for the control of service-rate. At the highest level, an extension to an existing service-rate allocation algorithm is presented and its integration with lower levels discussed. At the lowest level, a novel link service protocol (based on heuristic arguments) is described. For the intermediate levels we make use of the solution of the service-rate at the VP level described in Chapter 4. To illustrate the flexibility of the hierarchically organised control structure, we formulate another intermediate level and discuss its integration with the overall structure.

Finally, the conclusions appear in Chapter *6* together with a listing of the specific contributions of the thesis and the suggestions for further work in this area.

CHAPTER 2

TOWARDS A CONTROL FRAMEWORK FOR BISDN

2.1 Introduction

This chapter discusses the philosophy behind the control framework employed in the later parts of the thesis. In particular, for the control of BISDN we propose: the use of feedback, possibly supplemented by feedforward, control; a hierarchically organised control structure which features both a vertical as well as a horizontal decomposition; that a number of different (dynamic) modelling techniques should be employed; and that the (dynamic) tradeoff between service-rate, buffer-space, cell-delay and cell-loss should be exploited.

In section 2.2 we focus on feedback control and define a control horizon, within which feedback controls are effective. We discuss the time and space behaviour of the system, and show that due to the varying, wide ranging "time constant" of the network a structured approach to handle its complexity is required.

In section 2.3 we present such a structured approach: that of a hierarchically organised system. The control fbnctions in the hierarchically organised control system can be allocated to a level related to an appropriate time scale and location. The controls can be calculated in a decentralised fashion with the upper layers coordinating the distributed local units for the overall network benefit. Each control function is associated with a certain control horizon, which sets a physical constraint on the location of the control processing relative to the site of the particular physical control event. At lower levels of the hierarchy, control fbnction processing is constrained to be physically close to the control events, so must be distributed throughout the network. At higher levels, the control horizons become so large that physical location of control function processing is arbitrary.

In section 2.4 we suggest that different modelling techniques may be usefully employed to describe different levels of the decomposed (possibly hierarchically organised) control system. Furthermore, as it is increasingly noted, communication networks must have satisfactory dynamic as well as steady-state performance, thus the dynamic aspects of the behaviour cannot be ignored.

The (dynamic) tradeoff between service-rate, buffer-space, cell-delay and cell-loss, discussed in section 2.5, allows more flexibility in the effective and efficient control of BISDN.

2.2 System behaviour in time and space

There is a relationship between temporal and spatial distribution of events in a BISDN. A number of connections will generate traffic that is expressed as a distribution of cells over time. The time distribution of the cell stream, as seen at a single point in the network, will also be influenced by the spatial distribution of the source nodes of the connections. The influence of spatial distribution on temporal distribution is due to the finite speed of propagation of cells. Of course, delays arise from other causes as well as propagation, but all delays may be likened to propagation times and hence to equivalent distances in space (see section 2.5, figures 2.5 and 2.6). In the following discussion, the time scales for the feedback control mechanism and for the network are considered separately. By considering the whole system in this way, we will develop arguments for the distribution in space of certain essential control functions.

2.2.1 Traffic Sources and Services

A BISDN network provides services of various types between users distributed in physical space. The traffic generated by users forms one component of the time and space characteristics of the network. The service types and user demands will be extremely diverse. In general, we need to consider traffic types including Connection Oriented and Connectionless, Continuous Bit-rate and Variable Bit-rate, and with or without a Timing Relation between origin and destination. The Quality of Service (QoS), as perceived by a network user, will depend on the cell-loss, the cell-transfer-delay and the cell-delay-variation (a brief discussion of QoS appears in section 2.5).

2.2.2 Network Physical Resources

The resource infrastructure, comprising the physical links and processing nodes, is finite and also distributed in space. The resources that we will consider are of two types: cell waiting places in buffers, and cell-slots on links (the server resource). Note that these are substitute resources (see section 2.5, [22], [23])

2.2.3 Feedback Control System

It is well known that feedforward and feedback controls both have essential roles in effective control. Feedforward control is immediate and effective, but only when measurements and behavioural models are accurate. Feedback can be effective even in the presence of model inaccuracies, but reacts only after a disturbance has begun to take effect on the system, and its speed of response is limited by the total delay around the feedback loop. Here we concentrate on the feedback controls, to gain insight into the influence of the spatial distribution on the performance of feedback controls.

2.2.3.1 Control Horizon

The *control **horizon*** will be defined as the shortest time scale possible for a particular control action, and can be usefully visualised in either time or distance units. For example, a control horizon of 100 celltimes at 155 Mbit/sec is equivalent to the propagation delay of a signal on a approximately 60 kms link. The diagram below shows the relationship between the physical link dimensions of a section of metropolitan network and a 100 cell control horizon circle[#1].

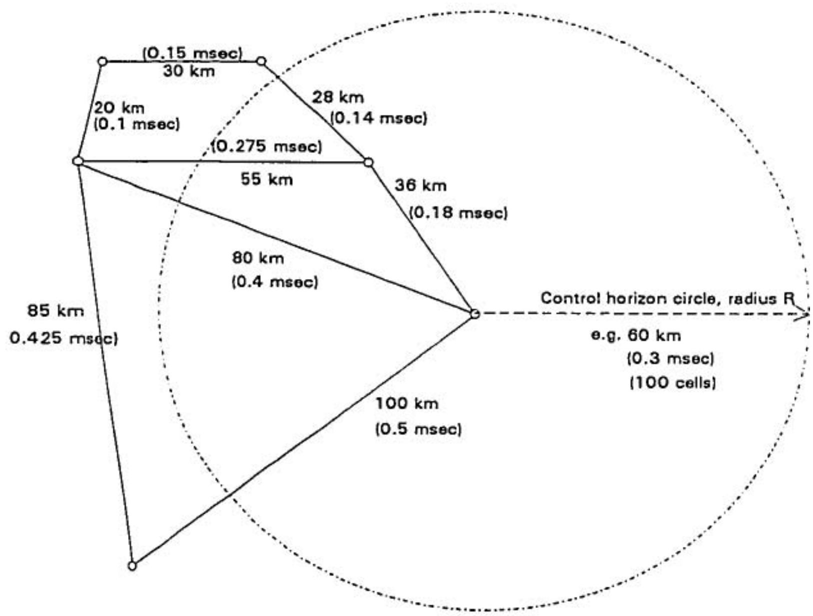


Figure 2.1. Physical and temporal relationship in a network.

The control horizon for a feedback control loop is determined by many factors–these are complex and diverse, but for discussion purposes these will be replaced by the loosely defined idea of "time constant", expressed as the sum of "feedback time constant" and "network time constant" (see figure 2.2). Note the use of-"time constant" here is different from the classical, rigorously defined, time constant.

[#1] Note that for averaging periods of say 600 cells (a reasonable averaging period for flow measurements), corresponding to 360 kms (a control horizon that covers most metropolitan networks), propagation delay may be an insignificant constraint on feedback control.

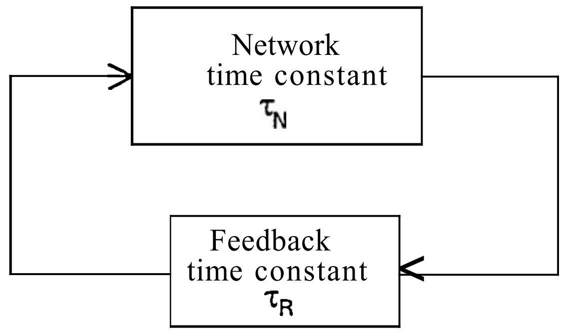


Figure 2.2. Factors influencing the control horizon.

The "feedback time constant" is a function of the measurement interval, the propagation delay from point of measurement to point of control action, and the processing time required to calculate the new control actions.

$$\tau_R = f(\tau_m, \tau_p, \tau_{pr}) \tag{2.1}$$

where:

τ_m is the measurement interval, i.e. the time involved in making a measurement,

τ_p is the propagation delay, and

τ_{pr} is the processing time (which includes the time required to make the measurements and calculate the feedback signals, e.g. unfinished work distribution in a buffer).

The "network time constant" depends on the duration of the transients following a control action. This is more difficult to represent, since it depends on a number of network and user attributes and requirements. Among others it depends on:

$$\tau_N = f(\lambda_{call}, \lambda^d_{call}, \tau_h, \frac{\partial \lambda_{call}}{\partial t}, D_{\max}, P^{\max}_{loss}, \lambda_{cell}, \frac{\partial \lambda_{cell}}{\partial t}, C_{link}, B_{link}) \tag{2.2}$$

where:

λ_{call} is an S x 1 vector of the call arrival rate

λ^d_{call} is an S x 1 vector of the call death rate

τ_h is an S x 1 vector of the call holding time

$\frac{\partial \lambda_{call}}{\partial t}$ is an S x 1 vector of the rate of change of the arrival or death rate of the call rate

$D_{\max}$ is an S x 1 vector of the maximum tolerable delay

P_{loss}^{max} is an S x 1 vector of the maximum tolerable cell-loss

λ_{cell} is an S x 1 vector of the cell flow rate

$\frac{\partial \lambda_{cell}}{\partial t}$ is an S x 1 vector of the rate of change of the cell arrival rate

C_{link} is the link service-rate,

B_{link} is the link buffer-space (if logically divided among the S type calls, then its individual components must be taken into account).

S number of calls (or the number of different types of calls; assuming that the properties of a call can be adequately represented by a general class i, where $i = 1,..,S$).

It is commonly assumed that network behaviour can be separated into different components, each of which has its own "network time constant". These components can then be grouped into broad categories by time constant. At least five broad time constants (levels) have been proposed in the literature [24], [25], mainly using intuition: cell level (tens of microseconds); burst level (tens of milliseconds to seconds); call level (tens of milliseconds to tens of minutes); virtual path level (several seconds to tens of minutes); and the network level (tens of minutes, hours, days). The recognition of the existence of different time scales for different controlled events has been an important first step in providing effective controls. What is required, in addition to the above decomposition of the time constant, is an association of the control functions for each broad level with their spatial distribution—that is the above time decomposition provides only the vertical level decomposition. We will now consider the other important point of view—the horizontal decomposition—by examining some of the fbnctions that need to be performed in the network.

2.2.3.2 Control Functions

Some examples of network control functions are presented in table 2.1. For each control function the table shows: the probable time scale and location; the controlled event and its location; and the main factors affecting the time response. Note that the probable time scale and the probable physical location of the control function are deduced from the type of information the control functions require and also the frequency of updating of the control output (as proposed in the literature, e.g. see [26]).

Control function (examples)	Controlled event	Controlled event location	Probable time scale level of the control function	Probable physical location of the control function	Main factors affecting the time response
- reconfiguration of the network (e.g. creation and deletion of VPs) - resource reservation (e.g cell-rate for VPs)	- network topology, VP topology - service-rate	spans all nodes of the network	network level	main control centre	$\lambda'_{call}, \lambda^{d,i}_{call}, \tau'_h,$ $\frac{\partial \lambda'_{call}}{\partial t}$
- routing (e.g. selection of alternative VPs) - dynamic service-rate control - flow control -rate based -window based	- VP connection, Virtual call connection - service-rate - cell-flow	spans several nodes from origin to destination	VP level virtual connection level	main control centre or Origin node or nodes along the connection path	$\lambda'_{call}, \lambda^{d,i}_{call}, \tau'_h,$ $\frac{\partial \lambda'_{call}}{\partial t}$ $D'_{\max}, P^{I,\max}_{loss}, \lambda'_{cell},$ $\frac{\partial \lambda'_{cell}}{\partial t}$ C_{link}, B_{link}
- Call Admission Control (CAC) - dynamic service-rate control - flow control - fast reservation protocol -explicit congestion notification	- call, connection - service-rate - cell-flow - service-rate - cell-flow	spans several nodes from origin to destination	call level	main control or Origin node or nodes along the connection path	$\lambda'_{call}, \lambda^{d,i}_{call}, \tau'_h,$ $\frac{\partial \lambda'_{call}}{\partial t}$ $D'_{\max}, P^{I,\max}_{loss}, \lambda'_{cell},$ $\frac{\partial \lambda'_{cell}}{\partial t}, C_{link}, B_{link}$
- Cell discarding - priority control - usage parameter control - traffic shaping - link server discipline - link buffer	- cell - cell - cell-flow - cell-flow - service-rate - cell-space	node	cell level	nodes along the connection path	$D'_{\max}, P^{I,\max}_{loss}, \lambda'_{cell}$ $\frac{\partial \lambda'_{cell}}{\partial t}, C_{link}, B_{link}$

Table 2.1. Examples of control functions showing: the controlled events and their location; probable time scale levels and location of functions; and factors affecting the time response.

From table 2.1 we get a glimpse of the variation of the time constant as well as the geographic separation of the events. The time constant can vary over several orders of magnitude. Even within the vertical levels identified in the literature–for example the call holding time for a video connection may be tens of minutes whereas for a file transfer it can be tens of milliseconds. The geographic separation can span all the nodes along a connection–from a few kilometres to thousands of kilometres.

For simplicity of exposition we focus on one controlled event, that of the service-rate. From table 2.1 we observe that it is associated with all four levels, indicating that elements of its time constant extend over all the identified levels. Especially, we note that the time constant can vary widely with the instantaneous connection mix, and can be time varying, even within an identified level. However in the literature the vast majority of the schemes discussing service-rate control focus on one level in isolation (possibly using the assumption that the time scales associated with the various levels are widely separated and hence no interference is experienced between the levels) and propose solutions that do not take into consideration the interactions between the different levels and the wide variations of the time constant values (even within the identified levels). The dynamic behaviour of the event is largely ignored.

Additionally, as can be seen from table 2.1, the controlled events (e.g. service-rate) are geographically distributed, and due to the limit imposed by the control horizon, cannot be effectively controlled unless the control processing is also distributed in space.

Based on the above discussion, we are prompted to find an approach that will allow us to tackle the diverse range of the time constants as well as the geographic distribution of controlled events. Only by decomposing events both vertically as well as horizontally (in time and space), and organising these control tasks in an orderly fashion will we be able to design coherent and effective controls.

2.3 Hierarchical (multilevel/multilayer) control

A hierarchical multi-level/multi-layer control structure (Mesarovic et al [27], Singh and Titli [28], Siljak and Sundareshan [29], Cruz [30], Findeisen et al [31], Mahmoud et al [32], Haimes et al [33]) is investigated here for the effective'and efficient control of this large scale complex network system.

There is an inherent constraint on a control system, irrespective of the control structure it is placed in. All elements of a (feedback) control loop for a given function must be within the physical constraint of the control horizon circle for that function. This sets a constraint on situations in which feedback control is feasible. Hierarchical control normally is used to tackle the problem of complexity. Decomposition allows the complexity of each block to be contained. **An** inherent consequence is usually to progressively relax the time scales at progressively higher (vertical) levels of the hierarchy. Hierarchical control is also relevant to the problem of physical distribution of the network and the finite propagation time between elements. Decomposition (horizontally) may allow lower level blocks to be limited to sets of physically close elements. The inherent relaxation of time scale at high levels of a hierarchical system can be directly associated with a relaxation of physical proximity constraints on the control elements of that functional level of the hierarchy. Therefore, within a hierarchically organised system, control functions can be allocated to a level related to an appropriate time scale and location, and controls can be calculated in a decentralised fashion with the upper layers coordinating the distributed local units for the overall network benefit.

Our proposed structure is intended to provide adaptability and robustness. Siljak [34] has shown that a complex system, when synthesised of interconnected stable subsystems, has highly reliable stability properties. **A** locally stabilised system is robust and has a high tolerance to nonlinearities in the interactions among the subsystems. The modularisation into simple well defined tasks also allows flexibility as well as a natural evolution path as technology changes.

The **key features** of hierarchical control are:

- Conceptual simplification is achieved by the decomposition of the system into several subproblems organised at different levels.
- Decomposition addresses complexity by yielding small subsystems of low dimensionality, and as a result, a reduction of the dimensionality of the overall system.
- The subsystems can be spatially distributed, and have limited communication with one another.
- Subsystems make decisions autonomously using private information. This allows fast acting local controllers to be implemented, coordinated by higher levels toward achieving global goals. Organisation of the control problem in decision hierarchies can be based on aggregated models, with different hierarchies solving different decision problems (the lower levels solve the more detailed problems whereas the highest level solves the global problem based on a more aggregated model of the overall system).
- Each subsystem can be described by a different class model. These models can be linear or nonlinear, static or dynamic or less traditional, such as Discrete Event

System (DES) models [35]. Therefore appropriate models can be used in order to adequately describe the behaviour of each component of the decomposed system.

- Each subsystem can be optirnised independently using optirnisation techniques appropriate for the particular subsystem. Note that this independence is provided by the fixing of certain variables in the optimisation and control problems for the subsystem. These values are updated by a higher level using coordination variables (for a discussion on decomposition and coordination see appendix 4.1, page 159).

Note though that there is a cost, delay, or distortion in transmitting information between the levels in the hierarchy.

Historical note

A number of papers have appeared in the open literature discussing hierarchical structures for the control of communication networks, some examples are: a general hierarchical structure for the control of BISDN (Pitsillides et al [36], [37]); a multidimensional framework centered around a rate-based access control scheme (Ramamurthy et al [38]); a general multistrata framework (the M-architecture) for resource allocation in which a stratum is defined both by the layer of traffic flow and the controlled level, and the resources can be allocated to cells, bursts, calls and flows (Filipiak [25]); a stratified network management scheme (Warfield et al [39]); a general layered approach for network management and control with the details of particular functions (task and location) left to the users (Campbell and Everitt [40]); a layered and distributed congestion-control framework (Eckberg et al [41]); a two level hierarchical structure for routing and flow control (Muralidhar and Sundareshan [42]); a two level hierarchical structure for call admission and bandwidth allocation at a single ATM channel (Bola et al [43]); a two layer congestion control scheme (Ren and Meditch [44]); a three level resource management architecture based on a preventative congestion control strategy (Anido, Bradlow, and co-workers [45]); bandwidth allocation in three levels, the packet, burst and cell level (Hui [24]); and Jain [14] who argued on strong intuitive grounds the assertion that no single congestion control strategy is adequate implying the need for more than one level. With a few exceptions, e.g. Warfield et al, Eckberg et al, Pitsillides et al, the proposed schemes have been motivated by the implementation of one solution to the problem, organised in a multilevel structure.

Our work differs from published work of others in the sense that we motivate a general hierarchical organisational and control structure from an analysis of the network behaviour both in **time and space.**

~~2.3.1 The~~ proposed hierarchical control structure

We consider three fundamental classes of control unit (see figure **2.3**):

Local Unit, LU:

At the lowest level a Local Unit (LU) uses local measurements, takes direct action, and may be coordinated by a higher level unit. Because of the physical separation of the local units and their limited control horizon , local units must have the ability to act autonomously.

Supremal Unit, SU:

At intermediate levels Supremal Units (SU) coordinate groups of lower level units. The intermediate SU level may need to communicate with higher levels as well as lower levels so that achievement of local objectives contributes to the global network benefit.

Overall Supremal Unit, OS:

At the highest level the Overall Supremal unit (OS) provides global coordination for overall network benefit. A number of conflicting and possibly noncommensurable objectives can be formulated at the overall supremal (OS) level, for example minimisation of total network delay, minimisation of total network loss, maximisation of bandwidth utilisation and maximisation of total network revenue. The overall supremal unit is responsible for resolving this conflict and directing the behaviour of lower level units.

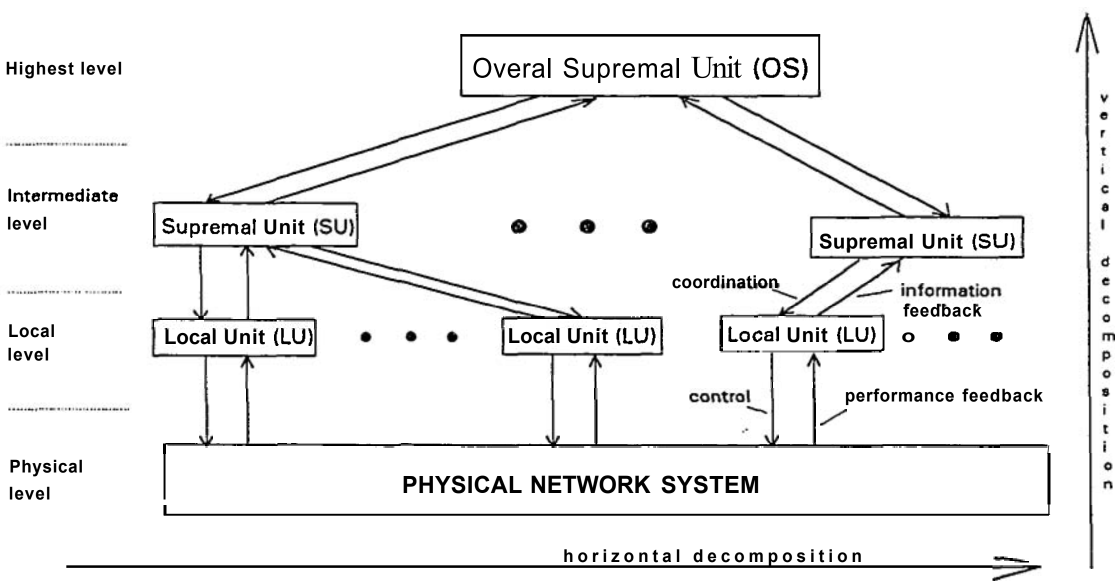


Figure **2.3.** A hierarchically (multi-level/multi-layer) organised control structure.

Note that there may be more than one level (with possibly their own local units) within each broad level of the hierarchy.

As an example we show a particular implementation of a BISDN control structure, using these general concepts, in figure 2.4. This example is mainly concerned with service-rate control and it makes use of the Virtual Path (VP) concept (defined as a pre-established route through the network into which Virtual Channels (VCs) can be grouped [46], [47], [48], [49], [50]). Note that the VP is a convenient basis for dynamic service-rate control since it represents flows of traffic (rather than individual connections). At the local level, the link LUs use local feedback, under some higher level direction, to allocate the time of the server resource (the link) amongst the VPs which share the resource. At higher levels of the hierarchy SUs are responsible for VP traffic control and resource management tasks. The function of the SUs is to allocate appropriate resources among the competing VPs, select alternative paths, shape traffic at the source, and accept/reject new call attempts. These functions can be organised into several levels, three of which are shown in this example: SUL3, responsible for coordinating the actions of all Origin-Destination (OD) pairs originating at a node; SUL2, responsible for one OD pair with (possibly) multiple VP paths through the network; and SUL1, responsible for one VP path of the OD pair. The likely location of the SUs is at the origin node of each OD pair. At the highest level of the hierarchy the OS is responsible for directing the behaviour of lower level units toward achieving the global objective(s), for example the maximisation of the total bandwidth utilisation. The likely location of the OS is at the main control centre.

2.3.2 Time scales and decompositions

As already discussed, there is an inherent relationship between vertical and horizontal decompositions . It is this relationship that will determine the necessary extent of the decompositions. Intuitively, one can state that: if the control horizon is not long enough to cover the span of the controlled event then further decompositions are necessary. Note that the investigation of this relationship is beyond the scope of this thesis: its formal study is recommended.

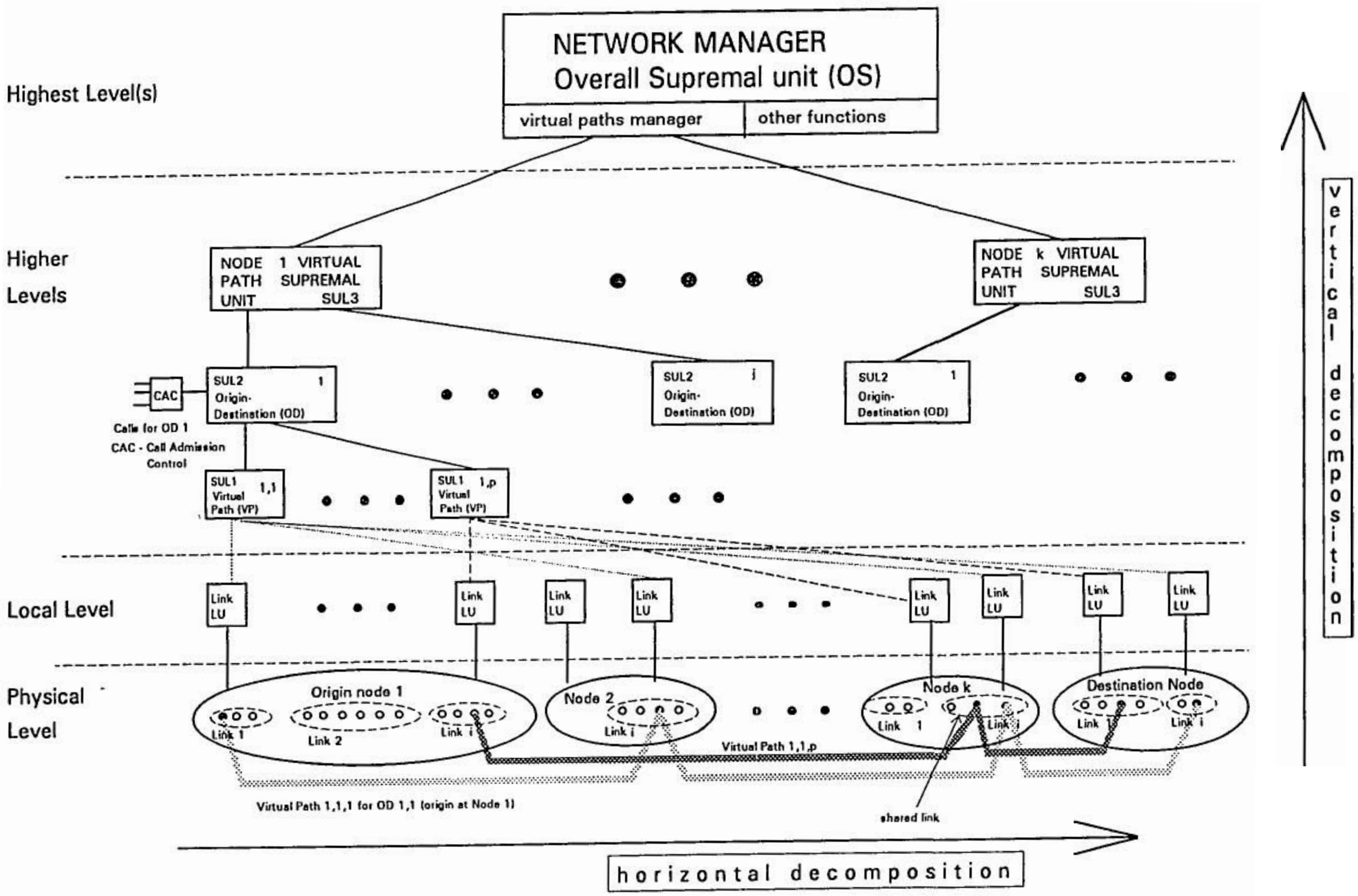


Figure 2.4. Example of a hierarchically (multi-level/multi-layer) organised control structure.

2.4 Modelling techniques for the dynamic control of BISDN

In this section, we look at modelling techniques suitable for the dynamic control of BISDN. The problem of modelling is inherently difficult due to the complexity of the system. In order to make analysis more tractable, the system must be modelled as an interconnection of simpler sub-systems, each giving adequate "lumped average" description of the total behaviour of the system at that level. The fundamental problem of modelling is that accurate characterisation of this system requires an excessively large state space. For example, assuming it can be modelled by a Discrete-State Continuous-Time (DSCT) Markov process, a 3 node network system with finite (100 cell places) buffers requires at least 10^{12} discrete states (see section 4.4, page 129). This precludes any hope of deriving a simple on-line controller by this method which is computationally feasible only for the simplest of cases. The key simplifying insight is that instead of dealing with "true" states of the complex system, we can work with aggregated quantities such as cells in a queue, flow of cells into a queue, and so on. The advantage of this approach is that

- expected values are real numbers as opposed to states which are integers, so we are dealing with continuously variable quantities instead of discrete-state variables,
- the model has lower state space dimensions, and therefore leads to algorithms which have reasonable computation time.

Something is lost of course: the innumerable transitions among the discrete states are lumped together as additive "noise".

key assertions:

- the most detailed model of DSCT Markov process model is not tractable, hence
- the use of expected values of quantities, which can be measured. Also
- the dynamic aspects of BISDN cannot be ignored, and that
- the system may be modelled as an interconnection of simpler sub-systems (vertical and horizontal decomposition), giving adequate "lumped average" description of the total system behaviour at each level.

Different modelling approaches are therefore required at different levels. In this thesis we make use of a range of models (e.g. in Chapter 3 we use a stochastic difference equation, in Chapter 4 a dynamic fluid flow model and in Chapter 5 a probabilistic model).

2.5 The cell-delay, cell-loss, buffer-space and service-rate performance and resource tradeoffs

The Quality of Service (QoS), as perceived by a network user, will depend on cell-loss, cell-transfer-delay and cell-delay-variation. The QoS target depends on the requirements of the individual connection (or connection mix). The achievement of the target QoS is determined by the service-rate and buffer-space resources allocated to the connection (or connection mix).

2.5.1 Cell-transfer-delay and Cell-delay-variation (CDV)

Queuing and propagation delays may be put into a users perspective as follows: A typical cell-transfer-delay encountered by a cell passing from an origin to destination can be seen in figure 2.5 for a 100 kilometre link, 1000 cell queue, and a service rate of 155 Mbit/sec. For voice, the packetisation delay dominates, and total delay is equivalent to less than 5 metres propagation at the speed of sound. For video telephony, 5 Mbit/sec video (assuming no frame delay) and 10 Mbit/sec data, transfer delay is dominated by queueing delay, but this is still negligibly small from a user perspective. [Note: packetisation delay, the delay experienced in "filling" up a cell, is taken as the quotient of the length of the information field of a cell (in bits) and the connection bit rate; and depacketisation delay (dejittering and reassembly) is made equal to the maximum queuing delay–this allows for the worst cell delay variation in the arrival of cells].

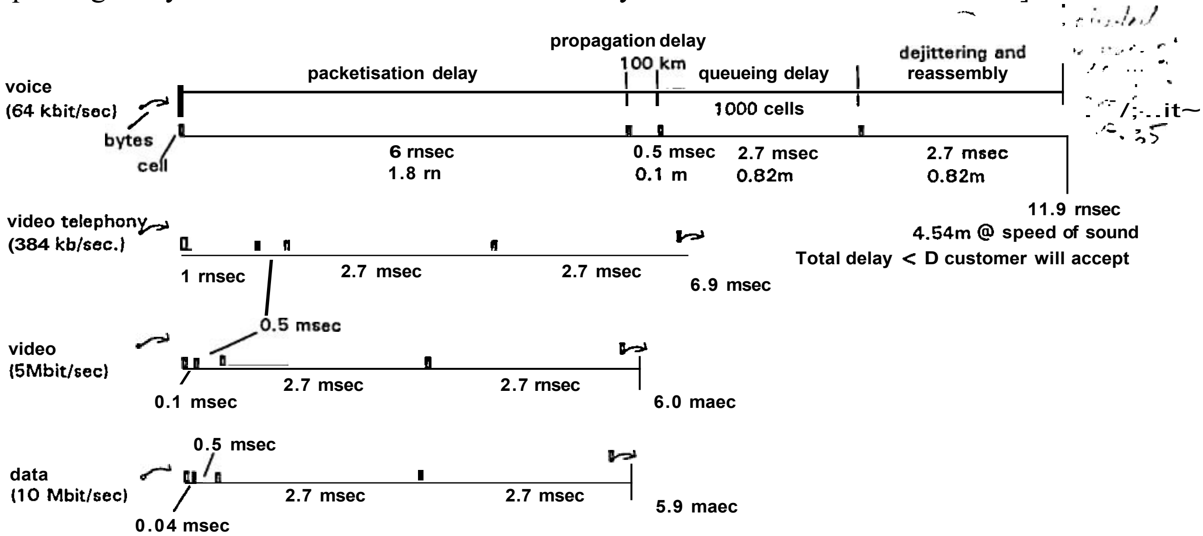


Figure 2.5. Time delays for various traffic types over 100 kms link and 1000 cell queue.

In figure 2.6 the voice example is extended to a 1000 kms link, with queueing delays of 1000 and 10000 cells. Voice is considered since it has a low cell-rate and therefore it features high packetisation and depacketisation delay.

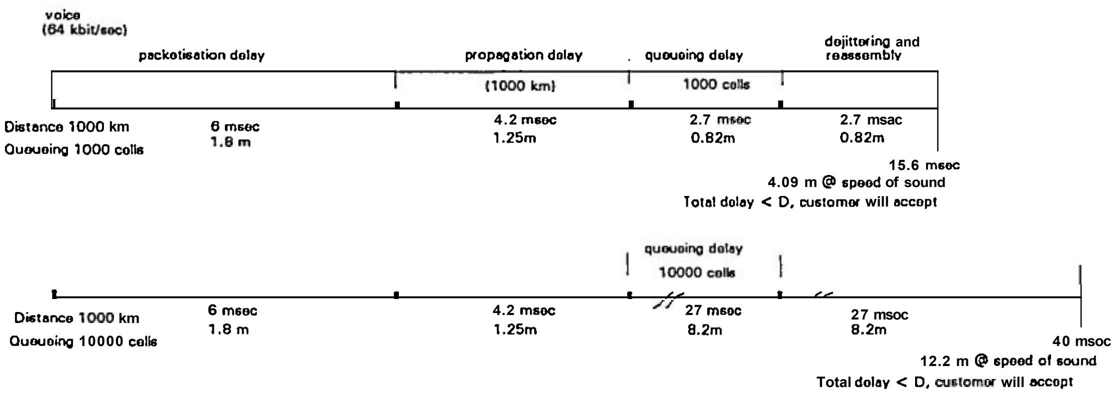


Figure 2.6. Time delays for voice over 1000 kms link and 1000 and 10000 cell queues.

It can be seen that from a customer point of view, fairly large distances (for a 1000 kms link the propagation delay is 4.2 msec) and/or large queues (the queueing delay for a 1000 cell queue is 2.7 msec) can be tolerated. For example, according to CCITT recommendation G.114 [51] the maximum permissible one-way delay for telephone connections is 400 msec#1 (although echo cancellation must be employed when a one-way delay of about 25 msec is exceeded). The dejittering and reassembly process imposes only a very small demand in terms of the storage memory requirements. For example, dejittering and reassembly for a voice connection requires a cell buffer of

$$b_d = \left\lceil \frac{\tau_q \times c^{dest}}{C^{link}} \right\rceil \qquad (2.3)$$

where

b_d is the size of the dejittering and reassembly cell buffer,

τ_q is the queueing delay in number of cells,

c^{dest} is the service-rate of the destination terminal (in most cases the same as the transmitting terminal),

$\lceil x \rceil$ is the next highest integer toward $+\infty$ (round up).

Example: for a 10000 cell-queueing-delay experienced by a voice connection, transmitted at a rate of 64 Kbit/sec, we require a 5 cell-buffer (i.e. 240 bytes) for dejittering and reassembly.

#1 This delay figure takes into account connections involving satellite links.

Note that certain applications, for example some data file transfers, are delay tolerant; considerably more than what has been shown for voice.

2.5.2 Cell-loss

Acceptable cell-loss performance is dependant on the tolerance of a connection to information loss. Quantitative assessments of the tolerance to loss for different service types are hard to find in the literature. A typical figure quoted in the literature as an acceptable target for BISDN is a probability of loss of $P_{loss} = 10^{-9}$ but this seems a much lower rate than necessary for many services. Voice is reported to be more tolerant to losses than the above figure, and other traffic types may also have a higher tolerance to cell-loss. Different targets for different traffic types are more relevant to customer requirements, but the network controls may then become more complex. Its worth mentioning that CCITT suggests two classes of cell loss rate (on a queue by queue basis): 10^{-6}, for example for voice connections; and 10^{-8} or 10^{-10}, for example for data. In Chapter 3, it is shown that the probability of loss can be regulated, by using a control theoretic approach.

~~2.5.3 The~~ tradeoffs

Cell buffer capacity and service-rate have been shown to be substitute resources in a queueing network [22], [23]. We consider their effect on cell-delay and probability of cell-loss.

If we allow the buffer size to tend to infinity, then the probability of loss, for a finite input rate will tend to zero, i.e. if

$$b \to \infty \Rightarrow P_{loss} \to 0, \quad \lambda < \infty \tag{2.4}$$

where

b is the number of cell-waiting places in the cell buffer,

λ is the input rate into the queueing system, and

P_{loss} is the probability of cell-loss (from the cell buffer).

Note that for system stability, we also require the input rate into the system to be less than or equal to the link server service-rate i.e. $\lambda \le C^{link}$.

Since maximum delay is directly proportional to the maximum number of cells in the queueing system (state of the cell buffer **x**), for a maximum tolerable delay and a finite cell-buffer size the above postulate must be modified to

$$x \to Min(x^{\max}, m) \Rightarrow P_{loss} \to Max(P_{loss}^{\max}, 0), \qquad x^{\max} \in X, \qquad \lambda \le C^{link} \tag{2.5}$$

where

x is the cell-buffer-state (and hence delay),

$x^{\max}$ is the maximum cell-buffer-state(and hence maximum delay),

$D_{\max}$ is the maximum tolerable delay,

$P_{loss}^{\max}$ is the maximum tolerable probability of cell-loss (from the cell-buffer),

X is the set of maximum cell-buffer-places (mapped from the set of maximum tolerable delays $\{D_{\max}\}$), and

C^{link} is the link server service-rate.

A demonstration of this tradeoff can be seen in table III of Heyman et al [21]. A real video sequence of half an hour duration is used as the input to a multiplexer with a finite cell-buffer and a fixed service-rate. The experiment is repeated for 5 different cell-buffer sizes determined by the constraint on the maximum allowed delay (for example, for a link rate of 45 Mbit/second, a cell size of 64 octets and a maximum allowed delay of 5 msec the buffer size is equal to 439 cells). The observed Probability of cell-loss ($\times 10^{-6}$) versus delay is shown in the figure 2.7 below:

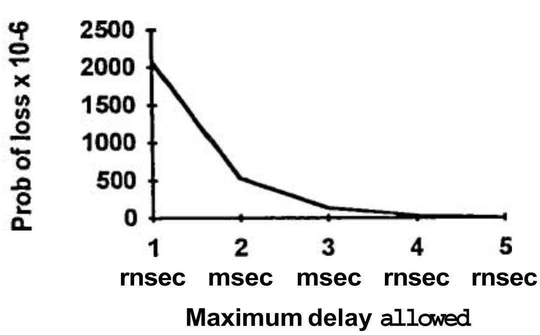


Figure 2.7. Tradeoff between cell-delay and cell-loss.

Observe the improvement in the probability of cell-loss by nearly 3 orders of magnitude (from 2.070×10^{-3} to 2.88×10^{-6}) for what appears to be a modest increase in the maximum tolerable cell-delay (fiom 1 to 5 msecs).

The discussion so far assumes that the link service-rate is constant, but if it is allowed to also vary then we have a means of influencing both the cell-loss and cell-delay, i.e.

$$C^{link} \to \infty \Rightarrow \{P_{loss} \to 0,\ x \to 0\}, \qquad \lambda \le C^{link}. \tag{2.6}$$

Thus, from the above discussion, we can observe the tradeoff between service-rate, buffer-space, cell-delay and cell-loss. Also observe that cell waiting places in a buffer and service-rate are substitute resources. As long as the QoS constraints on cell-delay and cell-loss are not violated, there is scope for the service-rate and buffer-space tradeoff. Note that more substantial gains can be made if the tradeoff is also dynamic.

Therefore the (dynamic) tradeoff between cell-delay, cell-loss, buffer-space and service-rate must be exploited in order to gain network efficiencies for both the customer and the network operator. For example, in Chapter 3 we offer a control scheme with a regulated QoS measure, within which these tradeoffs can be exploited by appropriate setting of the reference value.

Additionally, note that the tradeoff between buffer-space and control horizon range cannot be ignored (since by providing for more buffering, within the delay tolerance constraint, the control horizon can be extended).

2.6 Conclusions

The main conclusions are:

- the use of feedback for the effective control of BISDN is feasible despite the propagation delays, as long as it is constrained to lie within the control horizon (as for example by appropriate vertical and horizontal decompositions);
- a hierarchically organised control structure (featuring both a vertical as well as a horizontal decomposition) is necessary for the effective control of BISDN;
- different (dynamic) modelling techniques must be employed for the control of BISDN;
- the (dynamic) tradeoff between service-rate, buffer-space, cell-delay and cell-loss cannot be ignored.

The results in this chapter motivate our general approach to control in BISDN. For example, in Chapter 3 we integrate the problem formulation of CAC and flow control (we use a stochastic difference equation to describe the system, thus taking into account its dynamic behaviour), with the essential coordination handles provided in the problem formulation. In Chapter 4 we solve the dynamic service-rate control problem at the VP level (making use of a dynamic fluid flow type equation), and finally in Chapter 5 we provide an example of a hierarchically organised solution to the service-rate control problem (at the higher level a constrained optimisation problem using a steady state performance measure—derived from probabilistic principles—is used).

CHAPTER 3

ADAPTIVE CONNECTION ADMISSION AND FLOW CONTROL (ACFC)

3.0 Introduction

In this chapter, we aim to achieve both high utilisation of resources as well as maintain the network performance at acceptable levels, in user terms. We propose to achieve this by integrating Connection Admission Control (CAC) and flow control (generic functions for managing[#1] and controlling traffic and congestion in ATM networks [13]), and by making use of a novel control concept and adaptive control techniques. We aim to use a feedback control system to maintain the QoS close to a target value, irrespective of variations in traffic (the disturbance).

After a review of some of the existing CAC and flow control schemes we summarise their limitations.

We then define two distinct groups of traffic (controllable and uncontrollable), which allows us to introduce the concept of network controllability. Controllability is achieved by simply bounding the uncontrollable traffic. Bounding of the uncontrollable traffic becomes the sole role of CAC. The feedback signal is derived from a network performance monitor (which predicts the pth percentile of the buffer distribution, and hence the QoS). The controller regulates QoS by manipulating the flow of controllable trffic into the network. Controllability guarantees that the network can be operated efficiently (theoretically at 100% utilisation) and still provide the user with tightly regulated QoS (set at any desired target value).

We employ the general methodology of adaptive control (featuring both adaptive feedback and adaptive feedfonvard) to solve the difficult problems of CAC and flow control together. By using this approach we are able to do away with the restrictive requirements of existing schemes (for example the need to declare a complex traffic descriptor at the call connection request).

The proposed control scheme is formulated so that it can be integrated within a hierarchically organised control structure (essential for the overall control of BISDN–

[#1] The terms management and control have the following interpretation among Telecommunication engineers (loosely defined): management–it is a higher level task, implemented at a slow time scale; control–is a lower level task implemented at a much faster time scale *than* the management level. Since they are both control tasks, at different levels, we shall make no distinction between the two terms and use control for either.

see discussion in Chapter 2).

The performance of the derived algorithm is illustrated via analysis and simulation.

3.1 Review of CAC, flow control, and adaptive control schemes

As mentioned in the introduction, traditionally the design of CAC and flow control has been carried out separately without explicitly taking into account their interacting nature. We follow the same separation here to briefly review some of the existing approaches and highlight some of their limitations. We refer the interested reader to the vast literature on these two topics, and the report [52] on CAC. Additionally we point out that mainly preventive open loop strategies have been proposed in preference to reactive controls. These open loop strategies aim to prevent congestion by controlling traffic at the network edge, but without taking into account the actual loading conditions in the network (based on assumed, often worst case, traffic models). This results in low network utilisation.

We also review adaptive control, with particular emphasis on the class of Long Range Predictive Control (LRPC) adaptive algorithms.

3.1.1 Connection Admission Control (CAC)

Connection Admission Control is defined as "the set of actions taken by the network at the connection set-up phase (or during connection renegotiation phase) in order to establish whether a Virtual Channel/Virtual Path (VC/VP) connection can be accepted" [46].

Traditionally, for BISDN preventive open loop based strategies have been proposed in the majority of works. It is not uncommon to see statements such as "Call admission control is, of course, a form of preventive control, which is by its nature more effective than reactive control for use in high-speed networks" Murase et al [53], which they reference back to Woodruff [54]. The weakness of this idea is the assumption of the existence of an accurate model. Feedback on the other hand can be effective in spite of model inaccuracies and can also compensate for disturbances [55] (also see discussion in Chapter 2). However the large propagation delays (more precisely the bandwidth-delay product) have been extensively cited ([54], [45]) as the prohibitive factor for an effective feedback based CAC strategy. But, as discussed in Chapter 2 the problem of the large propagation delays can be avoided by appropriate formulations and control structures (e.g. adaptive control techniques can tolerate fairly long control intervals, and a coordinated decentralised control structure permits local feedback thus limiting the control horizon).

Despite being sensitive to model inaccuracies, CAC has been traditionally performed in an open loop fashion. Based on a set of user declared traffic parameters (a traffic descriptor[#1]) a model of the traffic behaviour and its influence on existing connections is inferred. If the call is admitted to the network then its traffic is monitored and controlled to ensure that the traffic contract parameters are not exceeded [13]. For effective CAC control these open loop (preventive) based schemes must be supplemented by an excessively conservative policing unit (a brief discussion of policing units appears in the next section) hence the network efficiency drops. The sensitivity of these CAC schemes to either overallocation of connections, or actual excess flow above the agreed parameters, is well known in the literature. This sensitivity is shown, for the case of one admission scheme based on the bufferless approximation, in the appendix 3.I (an excerpt from report [52]), where it is concluded: that linearisations of the connection acceptance surface should not be used to simplify the CAC algorithm; and that a very conservative CAC and policing scheme has to be used since only a few percent overallocation, or policing ineptness, will result in a very high probability of loss, at least as predicted by this model. Hence to address the sensitivity of buffer occupancy to uncertainty in traffic modelling we propose using adaptive feedback and feedforward control, which can resolve the problem of uncertainty [56] (examples of uncertainty include: model inaccuracies, traffic disturbances, and noisy feedback).

The vast majority of existing CAC schemes use (in an open loop fashion) either the average (long term) cell-loss probability[#2] ([57], [58], [53], [59]) or the effective bandwidth[#3] ([60], [61], [62], [63], [64]) as the parameter on which to base their decision to accept or reject a connection request. The average cell-loss probability method assumes simplified worst case connection models to decide whether to accept a new request. Hence it is inefficient. Furthermore different traffic types, as well as connections within the same type, may experience different QoS. The effective bandwidth is a complex function of the allocated link capacity, cell-loss probability, delay, source traffic characteristics, as well as the rest of the connections and their properties. Additionally the effective bandwidth of each connection will change as connections are added or removed. So the effective bandwidth is a highly complex, nonlinear, time varying variable, and attempts to find a usefbl, efficient, robust approximation of it have not been successful; one wonders if there is such an

[#1] A standardised traffic descriptor is often discussed in the literature, in which a set of standard traffic parameters is available that completely characterises the behaviour of a traffic source.

[#2] In most schemes the cell-loss probability is calculated for the aggregate of all service classes/connections within a service class.

[#3] The "effective bandwidth" is an estimate of the service-rate required by a connection (usually less than the peak rate) that takes into account the gains from multiplexing whilst satisfying network QoS constraints.

approximation (note that a robust approximation is to use the peak rate of a connection, however this is not efficient as it cannot benefit from multiplexing gains).

One of the very few exceptions to the open loop based approaches has been described by Saito et al in [65]. This uses real time measurements from the network and aims to improve an estimate of the upper bound on the cell-loss probability, based on user declared parameters. Note though that in a later paper [66] Saito discusses call admission using an upper bound of cell-loss probability calculated without monitoring the network load, but rather relying on user declared values explicitly (an open-loop approach, requiring at least the mean and the peak values of the number of cells amving during a fixed interval to be declared by the user).

Summarising some of the limitations, as seen by us, of existing schemes are:

- the need to make assumptions about the statistical properties of the connection. Note that in order to simplify the computations, some existing CAC schemes use an extreme case model (such as an on-off model) for all connections. Therefore an inefficient CAC policy is the result.
- reliance on user declared parameters, which are both difficult to estimate, (e.g. the average rate, average call duration, average burst rate, average burst duration) and in some cases difficult to police (e.g. the average bit rate).
- the need for an overly conservative policing unit, due to its inability to effectively police essential contract parameters, (e.g. the average bit rate).
- the need for open loop implementation, and as a result the need for a conservative admission policy to allow for model-reality mismatch (e.g. schemes based on the bufferless approximation assume a worst case on-off model for all sources, in addition to the assumption that there are no buffers in the network).
- schemes that rely on the average (long term) cell-loss probability have to rely on alternative schemes in order to deal with the short term losses. Additionally, schemes that calculate the cell-loss in the aggregate of all connections cannot guarantee the cell-losses of the individual connections.
- schemes that rely on the effective bandwidth assume that it remains constant over time, and independent of other service classes. Most of these schemes do not even allow multiplexing between service classes. Because they have to allow for a worst case effective bandwidth they are inefficient.

- these schemes are designed independently of "active" (see next section for a definition of "active") flow control strategies, even though they are highly interacting.
- most schemes rely on off-line computations, due to their computational complexity (see appendix 3.1 for one such scheme).

In this thesis we take into account all the above limitations to formulate an effective and efficient CAC policy.

~~3.1.2 Flow~~ control

Once a connection is admitted to the network, the vast majority of existing techniques propose Usage/Network Parameter Control (UPC/NPC)) as a means of monitoring and controlling the traffic on the connections. These controls are defined as follows [13]: "Their main purpose is to protect network resources from malicious as well as unintentional misbehaviour which can affect the QoS of other already established connections by detecting violations of negotiated parameters and taking appropriate actions". A discussion of some UPC/NPC or policing mechanisms can be found in [9]. Their effectiveness to police is still questioned [67], [68], [69]; even for what is seemingly a straightforward task of controlling, or policing, the peak rate [70]. For this reason, they have to be forgiving to variations from the contract parameters, otherwise they may penalise a "law abiding" connection unnecessarily. A conservative policing unit must be employed; hence the network efficiency drops due to the inability of the CAC to police effectively. These schemes can be viewed as "passive", in the sense that no action or regulation is taken unless a violation of the declared parameters is sensed.

In addition to UPC\NPC a variety of "active" flow control techniques, such as window (feedback) based mechanisms, rate based (mainly open loop) mechanisms, input buffer limits, isarithmic flow control, etc... have been discussed in the literature (see for example the survey in [71], the extensive discussion in [72], and the overview of rate and feedback controls in [73]). Various forms have been proposed (e.g. end-to-end , edge-to-edge, node-to-node) in traditional packet switched networks and some have been adapted for BISDN, as for example: window based flow control [74], [75], [76], [77] mainly considered for end-to-end; rate based flow control [38] implemented in an open loop fashion at the network edge; and [78] for a combined open/closed loop approach. A dynamic rate control scheme which adjusts the source video coding rate has been presented by Yin et al [79]. However their proposed control scheme is designed in an ad-hoc manner the source coder switches from low to high and visa-versa based on congestion information from the network. It also suffers from the high end-to-end propagation delays. A control theoretic approach to flow control has been adopted by a large number of researchers (see for example [SO], [18], [81]). Our approach is substantially different from those in the literature, in both methodology and problem formulation.

Summarising, edge-to-edge window based methods rely on feedback and are thus limited by the round-trip propagation delay, whereas the vast majority of rate based flow control schemes are open loop and hence have all the disadvantages of any open loop implementation. Later we show how the limitations discussed here can be overcome (e.g. by limiting the control horizon to node-to-node and customer-premises-to-node rather than end-to-end). We only implement an "active" form of flow control. However, a "passive" flow control unit may be incorporated to monitor the declared peak rate for only the members of a particular group of traffic—the uncontrollable group—discussed in the next section.

3.1.3 Adaptive control

Most control problems involve uncertainty. Sources of uncertainty are: noisy feedback and feedforward signals, caused for example by approximate calculations and noisy "sensors"; unmeasured disturbances which affect the output, caused for example by uncontrollable connections, against which the control law has to regulate; system nonlinearities modelled linearly; and above all inaccuracies in the system dynamic model. The use of adaptive control to resolve the problem of uncertainty is common practice in the control engineering community [82]. The basic idea of adaptive control is to adapt the control law in response to changes in the system dynamics or environment. This adaptation mechanism is based on input/output data and uses on line system identification techniques to infer a model of the system. Adaptive control theory has been around for over three decades, see for example the early contributions of Kalman in 1958 [83] and Mishkin and Brown in 1961 [84]. Much of the early interest in the application of adaptive control can be traced back to the papers by Peterka in 1970 [85], Astrom and Wittenmark in 1973 [86], and Clarke and Gawthrop in 1975 [87]. These papers communicated the adaptive control theory in a manner that made it readily available to practicing engineers with little training and background in adaptive control theory. A brief discussion of adaptive control theory is in the appendix 3.II.

3.2 Preliminaries

3.2.1 Network controllability and the controllable and uncontrollable groups of traffic

In this section we introduce the concept of controllability to determine whether our aim is achievable. We define **(strict) controllability** as follows:

> ***Definition I:*** A system is termed **controllable** if its control signal can influence the system behaviour, so that the desired output state of the system can be reached in finite time. Furthermore **strict controllability** is obtained if all desired output states are reachable in finite time.

Note: Our definition is less formal than **controllability** in the literature on control systems theory, and possibly closer to **reachability** [88], page **473.**

Many of the proposed approaches to the control of BISDN assume that all accepted connections are uncontrollable. This leads to a number of major performance limitations. Admission policies must be conservative and little flexibility is available to operate the network efficiently. Under normal conditions the admitted connections cannot be disconnected, or downgraded[#1], in response to the state of the network. Once a connection has been established the network must adhere to the negotiated parameters and performance guarantees. Control then relies mainly on open loop preventive measures, possibly supplemented by a number of closed loop policies operating at longer time scales. The feasibility of this control approach assumes the existence of an accurate traffic model. There is concern that even if a statistically correct model exists, the network will be operated inefficientlyand will not meet the needs of either the users or the network operators [15]. Where control is supplemented by edge to edge feedback methods (e.g. window based control) then performance is limited by round-trip propagation delays. The conventional approach to BISDN control is unlike other complex control methods where controllability is a basic design requirement. Ydstie et al [89] state (bold type introduced by the current author)

> ***"Most chemical processes have complex nonlinear dynamics and interactions between the different operating variables... Although the majority of chemical processes are designed to be self stabilising and can be controlled via the use of simple heuristic approaches like the one described above it is not at all clear that this is the best alternative. It is likely that energy, raw material and operating costs can be reduced***

[#1] A negotiated stepwise variable bit rate traffic source has been proposed in [90] to describe aggregation of sporadic sources. possibly generated by a LAN interconnect. However we see this as a much slower task, involving negotation between the user and ihe network, and hence not able to provide for an effective control tool to influence the network behaviour in the short term. Similarly, IBM's plaNET/Paris architecture provides downgrading or disconnection for violating users only in conditions of extreme congestion.

significantly i more advanced control and identification theory is used."

Even though the above quote discusses chemical processes, the same is true of other application areas, as for example complex mechanical structures, such as robots etc.

Thus before any effective control can be exercised, controllability of the network must be enforced. We define two groups of **traffic**; the **controllable** and the **uncontrollable group.**

> ***Definition 2:*** **Controllable traffic** is the group of traffic, that has agreed to have its cell-transmission-rate controlled by a network control system, in accordance with the network state. The performance on cell-loss can be guaranteed. However no guarantees can be offered on the end-to-end cell-transfer-delay, or cell-delay-variation. Note that the network controls do not need the user to declare any parameters.

Ideal candidates for this group of traffic are any delay tolerant connections. It is envisaged that appropriate price discounts will be offered by the network in order to provide the necessary incentives for a connection to accept classification as controllable. **A** dynamic pricing policy, see for example [91], driven by the network state, could even be designed and implemented.

> ***Definition 3:*** **Uncontrollable traffic** is the group of traffic, that has **not** agreed to have its cell-transmission-ratecontrolled. It has agreed to limit its peak cell-rate only, as negotiated during the connection admission phase. No other contractual requirements from the connection are required. The network will guarantee the **QoS** performance (**i.e.** the cell-loss, cell-delay and cell-delay-variation).

Connections that have strict timing requirements should join this group. Note that the only user declared parameter required for uncontrollable traffic is the peak rate.

The Quality of Service **(QoS)**, as perceived by a network user, will depend on the cell-loss, the cell-transfer-delay and the cell-delay-variation (see section 2.5). The **QoS** target depends on the requirements of the individual connection (or connection mix). The achievement of the **QoS** target is determined by the service-rate and buffer-space resources allocated to the connection (or connection mix).

> ***Definition 4:*** The **Quality of Service target (QoS target)** is a set of upper bounds for the ***cell loss, cell delay*** and ***cell delay variation.*** The network

operator will guarantee that the offered quality of service will be no worse than that specified in the QoS target.

A given QoS target is fully specified by the set

QoS target $=\{P^{\max}_{cell-loss}, \tau^{\max}_{cell-delay}, \tau^{\max}_{CDV}\}$.

where

$P^{\max}_{cell-loss}$ is the maximum tolerable probability of cell-loss (at the cell-buffer),

$\tau^{\max}_{cell-delay}$ is the maximum tolerable cell-queueing-delay,

$\tau^{\max}_{CDV}$ is the maximum tolerable cell-queuing-delay-variation.

Based on a QoS target the network is responsible for the allocation of adequate resources in order to ensure (with a given probability) that the required level of service is maintained at all times. We propose to achieve this by mapping the QoS parameters onto $E_{overflow}$ and y^{ref} as follows.

$E_{overflow} = P^{\max}_{cell-loss}$, and

$$y^{ref} = \left\lfloor \frac{\min(\tau^{\max}_{cell-delay}, \tau^{\max}_{CDV})}{\tau_{celltime}} \right\rfloor$$

where

$E_{overflow}$ is defined later (page 37) as the constant specifying the desired value of the expected cell-overflow probability,

y^{ref} is defined later (page 37) as the desired reference value for the QoS target feedback,

$\tau_{celltime}$ is the time taken to service a cell, i.e. it is equal to one celltime (e.g. for a 155 Mbit/sec link $\tau_{celltime}$ is about 2.6 psec), and

$\lfloor x \rfloor$ is the next integer toward $-\infty$ (round down).

Example: Assume that for a specific OD connection mix the QoS target is specified by the set {10^{-9}, 75 psec, 50µsec} and that the OD path spans a single node served at a rate of 155 Mbit/sec. The value of

$E_{overflow} = 10^{-9}$, and

y^{ref} = 19 cell-buffer-places (expected to accommodate $(1-E_{overflow})\times 100\%$ of the served cells).

∇∇∇

Introducing the concept of bounded uncontrollable traffic makes the network controllable. Therefore an effective control policy is now feasible.

3.2.2 Control concept

Our control concept is shown in figure **3.1.** By manipulating the flow of the controllable traffic, 'the control system aims to maintain the buffer utilisation such that the **QoS** is kept close to a target value, irrespective of the variations in the bounded uncontrollable traffic (the disturbance). This is feasible since the network is controllable, therefore the desired output state can be reached. Thus the network can be operated efficiently (theoretically 100% utilisation is achievable) and still provide the user with a tightly controlled **QoS** as determined by the target value. The formal role of **CAC** is to preserve the bound on uncontrollable **traffic** and hence enforce controllability. In practice the **CAC** policy may be more aggressive (see **CAC** algorithm 2, page 45).

Note that the transmission-rate of the uncontrollable **traffic** cannot be controlled once a connection is accepted into the network. The uncontrollable connections can transmit at any rate (in **a bursty** or otherwise fashion) limited only by their peak rate constraint.

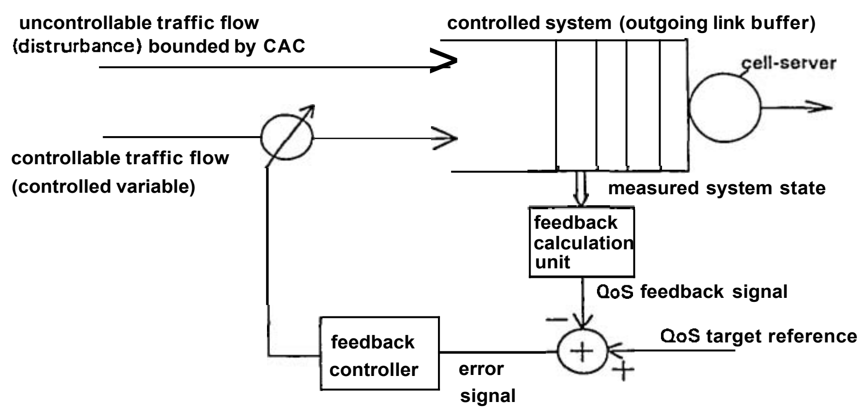


Figure **3.1.** Block diagram of the flow control concept.

Using the above concept, as shown in figure **3.1,** we are now in a position to solve the combined **CAC** and flow control problem.

Note, as discussed earlier, that a given **QoS** target (cell-loss, cell-delay and cell-delay-variation) is maintained by appropriate choice of the values of the cell-overflow constant ($E_{overflow}$) and the reference value (y^{ref}). The control system maintains the specified cell-overflow $E_{overflow}$ within the specified fraction y^{ref} of the physical buffer-space. As a consequence the cell-loss, cell-delay and cell-delay-variation(**QoS**) are indirectly controlled (or maintained) at prescribed levels. Of course, for controllable traffic the experienced cell-delay and cell-delay-variation must include the contribution of Q, the queue of the controlled sources.

As discussed in Chapter **2** the control structure must be decentralised, based on local feedback and also offer the necessary knobs to allow for its coordination from higher levels. Additionally the dynamic behaviour must be modelled. **An** overview of the proposed control concept for connection admission and flow control is presented in figure **3.2. A** single adaptive regulator is shown, located at a node with direct responsibility of one outgoing link. We envisage one controller per outgoing link (for ATM switches with completely partitioned buffers among the output links).

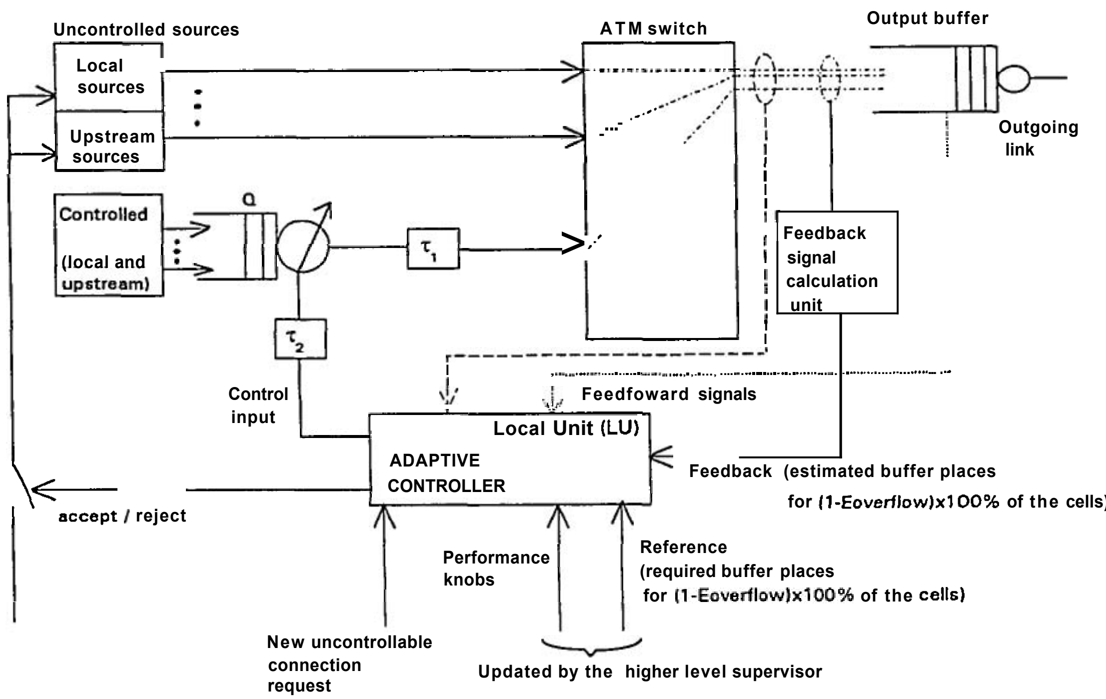


Figure **3.2.** The adaptive feedback and adaptive feedforward control concept for CAC and flow control.

At the Local Units we have various control options available, as for example:

1) direct control of all the local controllable sources. The controllable buffers can be located at the customers premises. The controllable part of upstream traffic can be controlled either by treating each upstream node (controllable part only) as a controlled source or by directly controlling each upstream source.

2) direct control of the queue Q outgoing rate. The controllable traffic from both local connections and upstream nodes is fed into Q. Note that in order to avoid cell-loss at Q, sources feeding into it must limit their flow, say for example by using the queue length as the feedback signal.

Option 1 trades off controller complexity for ATM switch simplicity (however at the expense of extra communication overhead and possibly the need to deal with large propagation delays–though substantially less than for schemes that rely on edge-to-edge strategies–for upstream controllable traffic). Option 2 trades off ATM switch complexity for controller simplicity. Note that the theory of adaptive control can handle large, and possibly varying, time delays, as well as multiple-input multiple-output formulations. It can therefore, in principle, be applied to control of traffic upstream. However, this is not good engineering practice because it is well known that large time delays in the control path will limit the effectiveness of any control scheme. This is one of the reasons for advocating action locally, supplemented by slower coordination (see discussion in chapter 2). **A** thorough feasibility analysis that takes into account these tradeoffs currently remains an open issue.

In this thesis we use option 2 in order to simplify the controller formulation and bring out the relevant issues. **A** conceptual ATM switch design for option 2 is described in the appendix 3.111. Further, for simplicity, we assume that the Q buffer size is infinite, and that the source is saturated.

3.2.3 Dynamic system model

A dynamic model based on the ARMAX representation is a popular choice among control theoreticians for adaptive control purposes. This model is well tested–a discussion of the model choice for recursive on line implementation can be found in section 5.3 of [92].

Mathematical formulation

Let

Y feedback based on QoS target (it predicts thepth percentile of the buffer distribution; see feedback sensor discussion in the next section). It is given in terms of the number of queue places, that are expected to accommodate $p\% = (1 - E_{overflow}) \times 100\%$ of all the cells passing through the system, e.g. if $E_{overflow} = 10^{-5}$ then 99.999% of the cell passing through are,expected to occupy cell places below the value ofy.

y^{ref} the desired reference value for y. Note that, as already discussed, this value influences the offered QoS.

$E_{overflow}$ a constant specifying the desired value of the expected cell-overflow probability, for example if $E_{overflow} = 10^{-9}$ then 1 cell out of 10^9 is expected to overflow above the reference value of y^{ref}. (Note that this does not necessarily imply that the overflowed cells will be lost. It simply means that cell places above the reference value are expected to be occupied). As discussed earlier, also note the influence of this constant on the offered QoS.

u^i the control effort; regulated flow of controllable sources; several controlled sources are allowed in the model ($i = 1,\ldots,N_u$).

v^j the feedforward signal; several feedforward signals are allowed in the model ($j = 1,\ldots,N_{ff}$), e.g. the uncontrollable measurable flow.

N_u number of control inputs.

N_j number of disturbance inputs.

ε unmeasurable disturbances and equation errors of unspecified character.

A general ARMAX model with multiple feedforward signals, and multiple control signals, is formulated

$$
\begin{aligned}
y(t)+a_1y(t-1)+\ldots+a_{na}y(t-na) = \\
\sum_{i=1}^{i=N_u} b_0^i u^i(t-k_i)+b_1^i u^i(t-k_i-1)+\ldots+b_{nb'}^i u^i(t-k_i-\text{nb}') \qquad (3.1)\\
+\sum_{j=1}^{j=N_J} \gamma_0^j v^j(t-k_j^f)+\gamma_1^j v^j(t-k_j^f-1)+\ldots+\gamma_{nc^j}^j v^j(t-k_j^f-n\gamma^j)+\varepsilon(t)
\end{aligned}
$$

where

t is the discrete sample time.

k_i is the delay experienced at the output due to the control input i (that is on applying the control input u_i at time t the output y is only affected after a time delay of k_i, measured in multiples of the discrete time t).

k_j^f is the delay experienced at the output due to the disturbance v_j.

$a_{ra}, b^i_{rb_i}, \gamma^j_{r\gamma_j}$ are the **coefficients** of the parameters that characterise the system

$(ra = 1,\dots,na, \quad rb_i = 0,\dots,nb^i, \quad r\gamma_j = 0,\dots,n\gamma^j)$.

$na, nb^i, n\gamma^j$ are the orders of the output, input and **feedforward** variables of the system.

In shorthand notation above equation can be described as

$$A(q^{-1})y(t) = \sum B_i(q^{-1})u_i(t-k_i) + \sum_j \Gamma_j(q^{-1})v_j(t-k_j^f) + C(q^{-1})e(t) \tag{3.2}$$

where

q^{-1} is the backward shift (or delay) operator defined as $q^{-1}y(t) \equiv y(t-1)$, and

$C(q^{-1})e(t)$ is a linear filter, $C(q^{-1})$, driven by white noise $e(t)$ [a sequence of equally distributed independent normal (zero mean, unity variance) random variables]. It is common practice to represent the general disturbance $\varepsilon(t)$ in this form to represent coloured noise.

Note that in order to characterise offsets, i.e. the situation for which a zero control signal is accompanied by a nonzero output mean, as is the case here, an appropriate form is the CARIMA model

$$A(q^{-1})\Delta y(t) = \sum_i B_i(q^{-1})\Delta u_i(t-k_i) + \sum_j \Gamma_j(q^{-1})\Delta v_j(t-k_j^f) + C(q^{-1})e(t) \tag{3.3}$$

where

$\Delta = 1 - q^{-1}$ is the differencing operator, i.e. $\Delta y(t) = y(t) - y(t-1)$

$A(q^{-1})$, $B(q^{-1})$, $\Gamma_j(q^{-1})$, $C(q^{-1})$ are polynomials in the backward shift operator of appropriate orders.

This model leads to inherent integral control action in a natural way as a consequence of the internal model principle [93].

3.2.4 Feedback of the network performance

The on-line feedback "sensor" provides a measure of performance using the methodology of Addie and Zukerman, see for example [94], [95], [96], [97]. (The works of Warfield and Chan [98], Yamada and Sumita [99], and Ahmadi and Kermani [100] are worthy of perusal for an alternative on-line feedback "sensor".)

This feedback sensor offers several features, required for on-line control use:

- it provides a measure of the network QoS;
- it is accurate enough for the purpose intended (and it proves to be robust for deviations from the assumed model);
- it has predictive abilities;
- computationally it is not overly complex, and thus can be incorporated in the on-line computations;
- it allows arbitrary choice of interval length, which allows matching to the system dynamics;
- it allows arbitrary, and time varying, choice of cell-server rate, which can be important if bandwidth is dynamically allocated, as for example on a per VP basis (see discussion on page 46 and Chapter 5).

Addie and Zukerman have developed a Gaussian traffic model for a BISDN statistical multiplexer, and derived closed form approximations for the mean and the distribution of the unfinished work. This model is in terms of three parameters: the process mean, the variance and the autocovariance sum. These three parameters are estimated on line and then used in generating the feedback. We refer the interested reader to the above mentioned papers for the details as well as an insightful discussion of the modelling of these processes and the difficulties involved in trying to establish an accurate model, for (possibly) open loop implementation, that captures the full range of second order statistics. We only quote here the essential formula required to calculate the feedback variable. **A** very brief description of their work appears in the appendix 3.IV.

The feedback signal $y(t)$ is the expected number of queue places required to accommodate $p\%$ of the cell traffic (the pth percentile of the buffer distribution, (see page 80 equation (3.1V.6))

$$y(t) = t_p \approx \begin{cases} \frac{1}{s^*}\ln\left(\frac{1-\frac{p}{100}}{\tilde{c}}\right) & \tilde{c} > 1-\frac{p}{100} \\ 0 & \tilde{c} \leq 1-\frac{p}{100} \end{cases} \tag{3.4}$$

where $\mathbf{s}^*$ and $\tilde{c}$ are defined in appendix 3.IV, page 79.

Example: To calculate the expected number of queue places required to accommodate 99.9999999% of the cells passing through the system, we set $p = (1-E_{overflow})$ x

$100\% = (1-10^{-9}) \times 100\%$, i.e. one cell is expected to overflow above $y(t)$ every 10^9 cells passing through (note $E_{overflow} = 10^{-9}$). Also, consider a system whose net input process is a stationary Gaussian process with a utilisation ρ=0.7 and variance σ^2=50. For various degrees of correlation of the net input process, as captured by the autocorrelation sum S (see appendix 3.IV, page 79) and shown in the table below, the calculated queue places $y(t)$ are:

Autocorrelation sum, S	0	50	100	200
Calculated queue places, $y(t)$, required to accommodate p% of the traffic	54	155	258	465

This signal is calculated on line using a sampling interval length of T_{az} celltimes. In addition to this performance metric, we monitor the expected utilisation, ρ (equation 3.IV.7), the expected loss, $E\{L_\infty\}$ (equation 3.IV.8), and the probability of loss, P_{loss} (equation 3.IV.9). These are shown in appendix 3.VI, on the simulation results.

3.3 Formulation and solution of the CAC and flow control problem

We solve the CAC and flow control problem using system identification, adaptive feedback and adaptive feedforward techniques. The basic idea of adaptive control is to adapt the control law in response to changes in the system dynamics or environment. The adaptation techniques are based on input/output data and use on line system identification techniques to infer a model of the system (see figure 3.9). Hence these adaptive techniques do not necessitate any prior assumptions with respect to the model behaviour and/or statistical parameters.

3.3.1 The adaptive Connection admission and flow control (ACFC) alaorithm

The solution of the combined CAC and flow control problem follows readily once the flow control problem is solved. These are presented as phase 1 and phase 2 below.

3.3.1.1 Phase 1: Flow control

An adaptive control algorithm of the LRPC class, is appropriate for this application, as it offers: robustness and insensitivity to non minimum phase and open loop unstable systems; insensitivity to inaccurate prior knowledge of the time delay; insensitivity to the model order; computational attractiveness. **An** implicit adaptive control approach is selected. This enables the derivation of a simple (computationally) control algorithm featuring a RLS identification algorithm. The incremental form (or integrating form) is used since it can take care of offsets. As an illustration of the application of adaptive control, in this thesis we provide a solution based on a simple form of the LRPC adaptive controller type.

Using the system model equation (3.3) in its incremental form, we derive the integrating, or incremental, k-step-ahead prediction model in a robustified form by using the Diophantine equation (see appendix 3.V.A)

$$\begin{aligned} &A_o(q^{-1})A_m(q^{-1})y(t+k) - A_o(1)A_m(1)y(t) = \\ &\quad = \mathcal{A}(q^{-1})\Delta y(t) + \mathcal{B}(q^{-1})\Delta u(t) + \sum_j \mathcal{G}_j(q^{-1})\Delta v_j(t) \end{aligned} \tag{3.5}$$

where

$A_o(q^{-1})$ is the observer polynomial; it is set equal to 1 (optimal choice is for $A_o(q^{-1}) = C(q^{-1})$ but since $C(q^{-1})$ is not explicitly identified[#1] an appropriate choice is to set it equal to one).

$A_m(q^{-1})$ is the (desired) model polynomial, that shapes the output in response to the reference $y^{ref}(t)$ as $A_m(q^{-1})y(t) = A_m(1)y^{ref}(t-k)$.

$\mathcal{A}(q^{-1})$, $\mathcal{B}(q^{-1})$ and $\mathcal{G}_j(q^{-1})$ are polynomials whose coefficients are identified directly from system data (note that if an indirect adaptive control approach is selected then the coefficients of these polynomials can be calculated from the model parameters by solving the Diophantine equation).

$A_o(1)A_m(1)$ the term multiplying $y(t)$ is added to the k-step-ahead prediction equation to **robustify** it (see Astrom and Wittenmark [55], page 434).

This prediction model allows us to estimate the coefficients of the controller directly.

We then consider a simple quadratic cost functional

$$\underset{\Delta u}{Min}\, \tfrac{1}{2}\left\{\left[A_m(q^{-1})y(t+k) - A_m(1)y^{ref}(t)\right]^2 + \lambda_o\left(\Delta u(t)\right)^2\right\} \qquad \lambda_o > 0 \tag{3.6}$$

where

λ_o is the control penalty factor; which trades control effort (variance) for output precision (model following variance).

[#1] **RLS is used due to its computational simplicity and the fact that the regulation error converges to zero despite the possible bias in the estimation of the parameters.**

This functional is representative of the so called "extended minimum variance" cost functionals [87].

Note that, for computational simplicity, we have not included any hard constraints on the control effort and the output. Soft constraints have been used instead—a common approach among control theoreticians. For example in the simulation of section 3.4.1 the control signal is clipped to lie in the range 0-60 Mbit/sec.

By minimising (3.6) on the basis of (3.5), we obtain the control effort (see appendix 3.V.B for the proof)

$$\Delta u(t)=\frac{\beta_o}{\beta_o^2+\lambda_o}\{A_m(1)[y^{ref}(t)-y(t)]-\mathcal{A}(q^{-1})\Delta y(t) \\ -(\mathcal{B}(q^{-1})-\beta_o)\Delta u(t)-\sum_j \mathcal{G}_j(q^{-1})\Delta v_j(t)\} \tag{3.7}$$

where β_o is the first term of the $\mathcal{B}(q^{-1})$ polynomial.

This control law is clearly a combination of feedback from the output signal $y(t)$ and feedforward from the measured disturbances $v_j(t)$. It also allows the tradeoff between input and output model following variance, by using the control penalty factor λ_o. Observe that the output follows a setpoint via the model following polynomial $A_m(q^{-1})$. The setpoint and A, are the "knobs" that one can utilise for the coordination of these local units by a higher level coordinator (see discussion in Chapter 2).

The control law (equation 3.7) belongs to the general class of the weighted one-step-ahead-control. However its robustness is improved by the incorporation of output following (output follows a reference model) and feedforward action. Also the handling of offsets has been included by using the integrating form of the ARMAX model. Its engineering properties are discussed in the appendix 3.IV.C, page 86.

3.3.1.1 The adaptive flow control alaorithm

The algorithm can be divided into two steps.

At each sample time T_s, perform the following:

step 1: Parameter estimation.

Using the prediction model (3.5), we form:

the auxiliary filtered output

$$Y^*(t) = A_m(q^{-1})y(t) - A_m(1)y(t-k);$$

the parameter vector

$$\theta(t) = [\alpha_1 \ldots \alpha_{n_\alpha}\ \beta_0 \ldots \beta_{n_\beta}\ \gamma_1^1 \ldots \gamma_{n_{\gamma 1}}^1 \ldots \gamma_1^{N_\sigma} \ldots \gamma_{n_{\gamma N_\sigma}}^{N_\sigma}]^T;$$

and the data vector

$$\varphi(t-k) = [-\Delta y(t-k) \ldots -\Delta y(t-k-n_\alpha)\ \Delta u(t-k) \ldots \Delta u(t-k-n_\beta)$$
$$\Delta v_1(t-k_1^1) \ldots \Delta v_1(t-k_1^1-n_{\gamma 1}^1) \ldots \Delta v_{N_\sigma}(t-k_1^{N_\sigma}) \ldots \Delta v_j(t-k_{n_{\gamma N_\sigma}}^{N_\sigma}-n_{n_{\gamma N_\sigma}}^{N_\sigma})]^T$$

Note: the predicted auxiliary output $\hat{y}^*(t) = \varphi^T(t-k)\hat{\theta}(t-1)$ and the prediction error $\varepsilon(t) = y^*(t) - \hat{y}^*(t) = y^*(t) - \varphi^T(t-k)\hat{\theta}(t-1)$.

Then using standard RLS the control parameters $\hat{\theta}$ are identified directly.

step 2: Control effort calculation.

Substituting the unknown parameters of the controller with their estimates, obtained in step 1, the control effort, equation (3.7),

$$u(t) = \frac{\beta_o}{\hat{\beta}_o^2 + \lambda_o}\{A_m(1)[y^{ref}(t) - y(t)] - \hat{A}(q^{-1})\Delta y(t)$$
$$-(\hat{B}(q^{-1}) - \hat{\beta}_o)\Delta u(t) - \sum_j \hat{G}_j(q^{-1})\Delta v_j(t)\} + u(t-1)$$

is calculated.

If required, say due to computational reasons, the updating of step 1 can be performed at a much slower time-scale, as a background task. Just as any practical control algorithm operating on line, its implementation requires considerably more code than that simply for implementing the mathematical relationship. Common practice is to provide numerically stable algorithms and "diagnostic", or "jacketing" software for checking the validity of the estimation and any numerical problems

3.3.1.2 Phase 2: CAC

CAC can be viewed as a background task working together with the flow control scheme. Its task is to bound the uncontrollable traffic (i.e. by controlling the admission of the calls requesting classification as uncontrollable) in order to ensure the (possibly strict) controllability of the network. Note that there is no limit (at least conceptually) as to the amount of accepted controllable traffic, and **CAC** in its simplest form will unconditionally accept all connection requests belonging to this group. Two **CAC** algorithms are offered. The first is very conservative (in terms of the admission of uncontrollable traffic), the second is more aggressive:

3.3.12.1 The **CAC algorithm No. I**

The most conservative strategy, and the only one that can guarantee the **QoS** of existing connections (by offering strict controllability) is described next. We assume that strict policing of the declared peak rate is exercised for traffic that belongs to the local group of uncontrollable traffic. We further assume, an upper bound of the total peak rate of upstream traffic that belongs to the uncontrollable group entering the node is known[#1] or at least estimated. For uncontrollable traffic, the **CAC** scheme simply adds the peak rate of the new uncontrollable connection request (h_{new}) to the sum of the declared peak rates of all existing local uncontrollable connections ($h^{local}_{existing}$) and the upper bound of the peak rate of upstream uncontrollable traffic entering the node ($h^{upstream}_{existing}$) to calculate the new total peak rate of all uncontrollable connections h_{total}. If the new peak rate of all uncontrollable connections, h_{total}, is less than the link rate then the new connection request is accepted. For controllable traffic the connection request is accepted unconditionally. Note that upstream controllable traffic does not influence the CAC, as it behaves as local controllable traffic (since it is fed into the local controlled queue Q).

The CAC algorithm No. 1

If new connection request belongs to the controllable group,

then accept immediately

else % new connection request belongs to the uncontrollable group[#2]

-calculate the h_{total}, using

$$h_{total} = h_{existing} + h_{new} \tag{3.8.a}$$

where

h_{new} is the declared peak rate of the new uncontrollable connection and

$h_{existing}$ is the sum of the declared peak rate of existing uncontrollable (local plus upstream) connections (or estimated upper bound of the peak rate for upstream traffic) $h_{existing} = h^{local}_{existing} + h^{upstream}_{existing}$

[#1] Note that if admission of new calls (belonging to the uncontrollable group) is carried out at all nodes along the path, that is from origin node to destination node, then the peak rate of all traffic entering the node (upstream or local) is declare

[#2] The %symbol is used to indicate a comment

$$- if \begin{cases} h_{total} < \text{Link } Rate \Rightarrow Accept \\ \\ else \qquad Reject \end{cases} \tag{3.8.b}$$

3.3.3.2.2 The CAC alnorithm No. 2

An alternative policy is to accept or reject uncontrollable traffic based on the network state. A simple comparison of the threshold on the throughput $\rho^{threshold}$ with the calculated throughput $\rho^{calculated}$ is all that is required. The $\rho^{calculated}$ is calculated using the peak rate of the new connection request, and the filtered version of the calculated steady state throughput $p=\frac{E\{A_\infty\}}{E\{B_\infty\}}$ (equation 3.IV.7) of the uncontrollable connections, i.e. $\rho^{calculated}=\rho^{filtered}_{measured}+h$, where $\rho^{filtered}_{measured}$ could be obtained by simply using a Moving Average (MA or FIR) filter [101]. Note that the filter parameters can be adaptively updated, based on measured data, by using an adaptive predictor [88].

The CAC algorithm No. 2

If new connection request belongs to the controllable group,

then accept immediately

else % new connection request belongs to the uncontrollable group

-calculate the $\rho^{calculated}$, using

$$\rho^{calculated} = \rho^{filtered}_{measured} + h \tag{3.9.a}$$

where

h is the declared peak rate of the new uncontrollable connection and

$\rho^{filtered}_{measured}$ is the filtered measured throughput of the uncontrollable connections

$$- if \begin{cases} \rho^{calculated} < \rho^{threshold} \Rightarrow Accept \\ \quad else \\ \qquad\qquad Reject \end{cases} \tag{3.9.b}$$

The setting of the threshold level $\rho^{threshold}$ is a higher level task and it will reflect the aggressiveness, or otherwise of the CAC algorithm, in terms of it favouring the uncontrollable traffic.

Note that both algorithm 1 and algorithm 2 are efficient because the network can be operated at a high throughput (since the flow control scheme ensures that the network uses controllable traffic to keep server utilisation high, while the QoS for uncontrollable traffic is not compromised). However algorithm 1 can guarantee the QoS (since it offers strict controllability) whereas algorithm 2 allows increased throughput of uncontrollable traffic at the expense of throughput of controllable traffic. But algorithm 2 requires adequate modelling of the uncontrollable sources, as does any other open loop CAC scheme. It guarantees QoS only to the same limit as the equivalent open loop scheme (and hence requires traffic to be well behaved), but allows in addition a certain throughput of controllable traffic, to the limit of 100% server occupancy.

Other similar CAC algorithms could be implemented, as for example the use of a probability of loss measure. However the proposed schemes offer simplicity, and yet they aim to achieve a high throughput without QoS sacrifice.

3.3.1.3 Combining the CAC and flow control algorithms to obtain ACFC

The CAC algorithm is an integral part of the flow control algorithm. The flow control algorithm can achieve the desired QoS yet it does not sacrifice network efficiency. However it relies on the CAC algorithm to ensure that the uncontrollable traffic is bounded in order to enforce controllability (strict controllability by using algorithm 1) into the network. Thus the proposed combined CAC and flow control algorithm featuring both adaptive feedback and adaptive feedforward (ACFC).

3.3.1.4 Integration of ACFC with a hierarchically organised control system centred around the VP concept

To integrate the ACFC with a hierarchically organised control system centred around the VP concept (an example of which appears in Chapter 5), the link service-rate (B_n, as defined in the appendix 3.IV, page 79) must be replaced by the service-rate allocated to the VP by the service-rate controller (e.g. the VPC or the VPAM outputs—discussed in Chapters 4 and 5). However the CAC schemes, proposed earlier, must then take into account the time varying nature of B_n.

Note that CAC can also keep a log of call attempts, and feed this to the upper layers, for example to the VP bandwidth allocation scheme (see for example VPOSU; defined in Chapter 5).

3.3.2 Implementation aspects of the ACFC adaptive algorithm

To implement the adaptive algorithm several practical questions must be addressed. Also, several design variables have to be chosen based on prior knowledge and/or experimentation with the system.

3.3.2.1 Properties of the adaptive algorithm

A general discussion and definitions of convergence, convergence rate, stability and stability margins of adaptive algorithms can be found in appendix **3.II,** page 76. In this section we state the theorem concerning the global convergence and stability of the proposed algorithm, with its proof given in appendix 3.**V.C,** page 85.

Theorem **3.1** Subject to assumption 3.1 below, the adaptive control algorithm (3.7) when applied to the system (3.3) is globally convergent, that is the following properties are satisfied:

i) $\{y(t)\}$, $\{u(t)\}$, $\{v_1(t)\}$ and $\{v_2(t)\}$ are bounded sequences for all t.

ii) $\lim_{t\to\infty}\left[y(t)-y^{ref}(t)\right]=0$.

Remark: The boundedness of the sequences also implies global stability.

Assumption **3.1**

i) The time delay k is known.

ii) **An** upper bound for the orders of the polynomials in the system model (3.3) is known.

iii) Conditions **b.i)** and **b.ii) of the** deterministic case (described in appendix 3.V.C page 88) are satisfied, **i.e.**

b.i) all modes of the "inverse" models (relating y^{ref} to $u(t)$ and y^{ref} to $y(t)$), **i.e.** the zeros of the polynomial

$$A_m(z^{-1})B(z^{-1})-\frac{\lambda_o}{\beta_o}(1-z^{-1})A(z^{-1})$$

lie inside or on the unit circle. Additionally any modes of the "inverse" model on the unit circle have a Jordan block size of 1.

Note: z is the complex variable of the z-transform.

b.ii) all controllable modes of the "inverse" models relating y^{ref} to $u(t)$ and y^{ref} to $y(t)$, i.e. the zeros of the transfer functions

$$\frac{1}{B(z^{-1})}\left[A_m(z^{-1})B(z^{-1})-\frac{\lambda_o}{\beta_o}(1-z^{-1})A(z^{-1})\right]$$

$$\frac{1}{A(z^{-1})}\left[A_m(z^{-1})B(z^{-1})-\frac{\lambda_o}{\beta_o}(1-z^{-1})A(z^{-1})\right]$$

lie strictly inside the unit circle.

iv) The scalar $\kappa_o = \frac{\lambda_o}{P_o}$ is specified, such that in the cost function the control penalty is given by $\lambda_o = \kappa_o\beta_o$ (since λ_o is required to be positive in the cost function then κ_o must have the same sign as β_o).

v) Sequences $\{v, (t)\}$ and $\{v_2(t)\}$ are bounded.

See comments on assumptions 3.1 on page 90.

3.3.2.2 Design variables of the adaptive algorithm

Choice of the model order

The choice of the model order is a nontrivial problem, that requires a tradeoff between good description of the data and model complexity. The basic approach for on line identifiers is to compare the performance of models of different orders and test if the higher-order model is worthwhile. Alternatively several criteria, as for example the Akaike Information Theoretic Criterion, AIC or Final Prediction Error FPE, or Rissanen's Minimum Description Length, MDL (see, for example, [102] for a discussion) can be employed. However for the implicit controller case there is no need to estimate the order of the model directly.

Choice of the controller order

The controller order is decided by the controller type and it is dependant on the model order, if an explicit design is undertaken. Note that our algorithm is based on the implicit extended minimum variance cost function regulator which is robust against model order assumptions.

Choice of the model following dynamics

A first order polynomial is a common choice of model (since this is adequate for damping the response), so $A_m(q^{-1}) = 1 - pq^{-1}$, where p can be interpreted as the desired pole location.

Choice of the sampling rate, T_s

The sampling rate is related to the desired controller bandwidth (i.e. its speed of response) and the uncompensated dynamics of the uncontrollable traffic. Fast sampling may make the regulator overambitious. Slow sampling may not be able to handle fast acting disturbances (related to the bursts, their arrival rates and their duration).

Choice of the time delay

The time delay can be estimated from system knowledge, for example the time delay caused by the propagation delay is straightforward to estimate from physical data (speed of light in fibre), or else obtain experimentally. Note that the use of LRPC can overcome some of the earlier design limitations of self tuning controllers; that is the requirement of a good estimate of the upper bound of the time delay, and their inability to cope with large variations in the time delay.

Choice of the forgetting factor

The forgetting factor can be updated automatically, see for example [103]. However this is at the expense of updating an extra variable. For algorithmic simplicity we will only consider manual setting of the forgetting factor. By appropriate choice of the forgetting factor one trades off alertness to follow the time variations versus sensitivity to the noise effects. This is achieved by altering the effective memory of the controller. In normal situations the value of the forgetting factor is typically chosen as 0.98, giving an effective memory of about 50 past samples. In time varying systems, the value of the forgetting factor must be set to a lower value, for example 0.9 gives an effective memory of 9 past samples.

incorporation in silicon

As the controller part features a simple recursive filter structure, it can be easily implemented in silicon. The identification part, which can be computationally more complex can either run as a background task at a much slower time scale, or it can also be incorporated into silicon (see for example [104] for a discussion of the systolic architecture for iterative LQ optirnisation; a much more computationally intensive task than the adaptive scheme proposed here). Note that this separation is feasible since adaptive control often has the characteristic that the states of the process can be separated into two categories, which change at different rates. The slowly changing states are viewed as parameters [105].

3.4. Performance evaluation

3.4.1 Simulation test bed

The cell-level simulation test bed consists of an ATM switch with multiple input connections, completely partitioned buffers among the output links, no internal blocking, and one output cell-buffer of I00 cell places at the outgoing port.

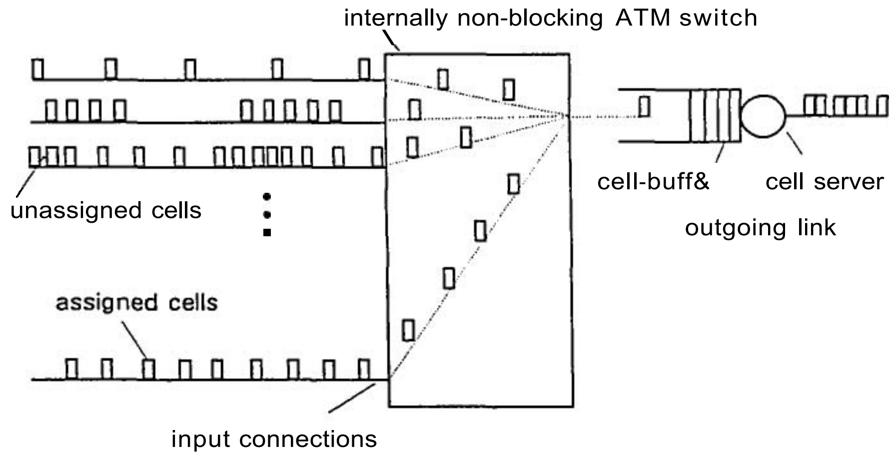


Figure 3.3. ATM switch simulator block diagram.

Only one outgoing link is simulated, since there is no interaction between any of the outgoing links. Connections at the input of the switch are multiplexed and routed to the outgoing link. If there are no cells in the outgoing link's cell-buffer an arriving cell is served immediately, otherwise it is placed in the buffer. Note that a cell has fixed duration and it requires a unit of time for transmission. One unassigned (empty) or assigned (nonempty) cell is always transmitted every unit of time. Thus we divide the time axis into celltimes. Each connection is simulated at the entry to the switch as a stream of assigned (filled or partially filled with information) cells spaced apart by an amount of unassigned cells. The spacing between assigned cells depends on the type of connection. The traffic load comprises a mix of voice, data and variable rate video, with random connections and disconnections of sources. Assuming a line rate of 155 Mbit/sec, for voice at 64 Kbit/sec: 171 cells are generated (at a constant rate) every second (equivalently one cell is assigned for every 2189 unassigned cells).

data (at say 10 Mbit/sec when on): 26667 cells are generated (at a constant rate) every second during the on-period, and zero cells during the off-period (equivalently one cell is assigned for every 14 unassigned cells)–similarly for other bit rates. The on-off periods of the data connection are randomly selected from an exponential distribution.

video (say 5-15 Mbit/sec): a variable number of cells is generated with a peak rate of 40000 cells generated every second. The number of cells during successive "frames" are correlated (in [106] four different types of correlation are identified: line, frame, scene and white noise). The first order AR model of Maglaris et al [107], (see also [108]), and a hard limiter (to enforce the upper and lower bit rates) are used to synthesise the video signals, with the cells equally spaced within one "frame". For simulative convenience the "frame" size is very small, in the region of 100 celltimes (it could represent macroblocks [109], [110] or a prebuffer), in comparison to a frame length of say 20 msec (7469 celltimes). This does not in any way limit the adaptive control approach and the interpretation of the results.

All measurements are performed at the outgoing link buffer. The sampling rate for the performance measurements is $T_{a:} = 30$ celltimes.

There are no limits, apart from computer memory limitations, as to the number of ATM input ports. Typically, to represent our view of a loaded ATM switch with a mix of voice, video and data traffic, we simulate the following two connection mixes to represent CAC algorithm 1 and 2 respectively:

<u>Connection mix 1:</u>

This ensures strict controllability by keeping the peak rate of the connection mix at or below the link rate (i.e. the peak rate of the resultant connection mix never exceeds 155 Mbit/sec). The mean rate is 92 Mbit/sec. It represents a utilisation of about 60%.

Uncontrollable traffic:

170 voice connections (multiplexed at the customer premises into 6 ATM switch input connections, with randomised starting points) at 64 Kbit/sec per connection

3 video connections at a peak rate of 25 Mbitfsec per connection (mean rate of 16 Mbitfsec per connection)

2 video connections at a peak rate of 15 Mbitfsec per connection (mean rate of 7 Mbit/sec per connection)

2 data connections at a peak rate of 10 Mbit/sec per connection (mean rate of 1 Mbitfsec for one connection and 9 Mbit/sec for the other)

3 data connections at a peak rate of 5 Mbit/sec per connection (mean rate of 0.5 Mbit/sec for the first connection, 2.5 Mbit/sec for the second and 4.5 Mbitlsec for the third)

2 data connections at a peak rate of 2 Mbitlsec per connection (mean rate of 0.1 Mbitfsec for one connection and 1.5 Mbitlsec for the other)

Controllable traffic:

1 controllable queue; maximum rate of 60 Mbitlsec

Connection mix 2:

This mix is more aggressive. Its peak rate exceeds the link server, but the mean rate does not. The sum of the peak rates of the uncontrollable traffic is equal to 192 Mbit/sec. It exceeds the link rate by about 24%. The mean rate of the uncontrollable sources is 114 Mbit/sec. It represents a utilisation of about **73%.**

The mix is as connection mix 1 but with the following additional connections:

1 video connection at a peak rate of 25 Mbitfsec (mean rate of 16 Mbitlsec)

1 data connection at a peak rate of 10 Mbit/sec (mean rate of 5 Mbit/sec)

1 data connection at a peak rate of 2 Mbit/sec (mean rate of 1 Mbitlsec)

Note: Connection mix 1 is used for simulation case **A** and Connection mix 2 for the rest.

3.4.2 The adaptive algorithm

Controller implementation

Two feedforward signals have been used for all simulation runs. One is from the average flow of the uncontrollable traffic (averaged over the control interval T_s), whilst the second is the average queue length (averaged over the control interval T_s). Note that for the first feedforward there is no need to identify the cells as to whether they are controllable or not, since from the measured total flow minus the controllable flow (control signal) we obtain the uncontrollable flow. The feedback signal, as discussed earlier, is calculated every control interval T_s (note that in order to calculate the feedback signal, the system is sampled every T_{az} time units).

Controller parameters

The system sample rate has been set at $T_s = 620$ celltimes, and the feedback signal is sampled every $T_{az} = 30$ celltimes. The controller parameters are chosen as n_β=4, n_α=3, n_{γ_1}=3, n_{γ_2}=3, and the dominant pole location p=0.5, offers a "detuned' model reference following. The forgetting factor λ=0.98, offers good tracking ability, and is able to handle fast system dynamic changes.

Controller and feedback sensor performance knobs

These are: the reference value y^{ref}, the overflow constant $E_{overflow}$ and the penalty factor λ_o. They are discussed in the simulation results section. As already stated these are the "knobs" via which a higher level supervisor can influence the local controller.

A block diagram of the adaptive controller can be seen in figure 3.4.

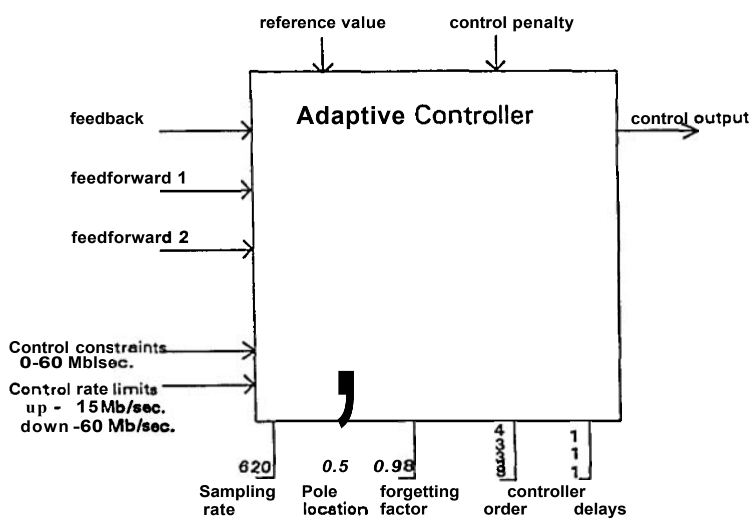


Figure 3.4. Block diagram of the adaptive controller and its standard settings.

Note that the control signal to the simulated controllable source process is clipped to lie in the range 0-60 Mbit/sec and the maximum up step is constrained to lie in the range 0-15 Mbit/sec.

3.4.3 Simulation results

Six simulation cases were considered. Connection mix 1 is used for case A (to represent CAC algorithm 1) and Connection mix 2 (to represent CAC algorithm 2) for the rest. The finite buffer size is set to 100 cell places. In case A, where $E_{overflow}$ is investigated, we set $E_{overflow} = 10^{-5}$ so that the simulation run length does not need to be very long (set at 744000 celltimes), whereas in all other cases $E_{overflow} = 10^{-9}$ in order to simulate a more demanding QoS requirement. Also, with the exception of case A, the simulation run length is kept fairly short at 37200 celltimes which is adequate for the investigations. The coefficients of the controller parameters are always set to zero at the start of a simulation run. In a real application, the coefficients can be set to the last updates of the parameters, and hence the initial transient phase can be even shorter. In addition, as stated earlier, we monitor and display the expected steady state utilisation, ρ (equation 3.IV.7), the expected steady state loss, $E\{L_\infty\}$ (equation 3.IV.8), and the expected steady state probability of loss, P_{loss} (equation 3.IV.9).

<u>case A) Simulation run for 744000 celltimes (sum of the peak bit rate of the uncontrollable sources is less than the link rate):</u>

In this run we use Connection mix 1 to ensure that the link rate is never exceeded. The reference is set at $y^{ref} = 25$ cell places. $E_{overflow}$ is set to 10^{-5} so that the simulation run does not need to be very long. The control loop then regulates the controllable traffic such that a buffer of 25 cell-places is expected to accommodate $(1-10^{-5})$ x 100% = 99.999% of the total cells served. The simulation length should be sufficient for about 7 overflow events on average.

For this simulation run of 3/4 of a million celltimes no loss occurred, the delay for the uncontrollable traffic remained bounded below 83 µsec (for a link rate of 155Mbit/sec), yet the throughput is a high 0.89 (compare to the throughput, of 0.60, obtained by setting the controllable source output to zero). Note that (for this setting of y^{ref} and $E_{overflow}$) a 48 % increase in the throughput rate has been achieved, over a scheme that admits on just the peak rate.

A summary of the results is shown in the table:

Simulation run length (celltimes)	Reference on cell places to hold 99.999% of the cells	Actual buffer places used to hold 99.999% of the cells	Throughput	Maximum instantaneous buffer places occupied	Actual cell-loss over the length of the simulation run
744000	25	30	0.89	31	0 out of 744000 cells

Observe that the actual cell-buffer places used to accommodate 99.999% of the cells passing through is equal to 30 in this run time; which can be compared to the reference value of 25. A very tight control is achieved (and hence delays and losses are bounded). **As** discussed earlier, the cell-loss, cell-delay and cell-delay-variation for the uncontrollable sources, can be kept at any desired value, by appropriate choice of y^{ref} and $E_{overflow}$ (assuming strict controllability).

See appendix **3.VI.A**for the simulation run details.

<u>Case B) Simulation runs for a video connection and disconnection:</u>

This set of simulation runs is designed to demonstrate the robustness of the scheme to unforeseen traffic connections and disconnections, without the necessity of a tradeoff in performance or efficiency. Three cases were considered: video 1 remains connected throughout the simulation run; video 1 is disconnected after 20000 celltimes; and video 1 is connected after 20000 celltimes. These could be interpreted as bursts of traffic, say due to interactive video. No noticeable degradation, in performance or efficiency, has occurred in any of the cases considered. The throughput remained constant at about 0.83 (0.85 if one excludes the initial transient phase), for all cases, even though there was a connection and disconnection of a video signal with a peak of 25 Mbit/sec. The maximum instantaneous buffer occupancy remained at a low 10 cell places for all three cases.

A summary of the results is shown in the table:

Run	Throughput	Maximum instantaneous buffer places occupied	Actual cell-loss over the length of the simulation run (out of 37200 cells)
Run 1	0.837	10	0
Run 2	0.828	10	0

Run 3	0.828	10	0

The reference is set at $y^{ref} = 25$ cell places and $E_{overflow} = 10^{-9}$. The control loop regulates the controllable traffic such that a buffer of 25 cell-places is expected to accommodate $(1-10^{-9})$ x100% of the total cells served.

See appendix 3.VI.B for the simulation run details.

Case C) Simulation runs for three different reference settings:

This set of simulation runs is designed to demonstrate that the utilisation can be increased toward unity by increasing the value of the output reference y^{ref} (note that a lower value of $E_{overflow}$ can also increase the utilisation). Also the ability (possibly from a higher level) to influence the behaviour of the system, by changing the reference value, is demonstrated. As the reference value increases the throughput increases, however with an increase in the maximum instantaneous buffer occupancy.

A summary of the results is shown in the table:

Run	Reference value, y^{ref}	Throughput	Maximum instantaneous buffer places occupied	Actual cell-loss over the length of the simulation run (out of 37200 cells)
Run 1	20	0.85	8	0
Run2	50	0.89	10	0
Run3	75	0.91	34	0

The reference is set at y^{ref} cell-places and $E_{overflow} = 10^{-9}$. The control loop regulates the controllable traffic such that a buffer of y^{ref} cell-places is expected to accommodate $(1-10^{-9})$ x 100% of the total cells served.

See appendix 3.VI.C for the simulation run details.

Case D) Simulation runs for different control penalty weights:

The tradeoff between output and control variance is demonstrated during this set of runs. In the first run, (for a low control penalty) the controller gives excessive weight to its goal to minimise the variance at the output and hence it does not prevent the loss (for the finite buffer of 100 cell places). **As** the control penalty increases in Run 2 and Run **3** we obtain a smoother control response and no losses are experienced.

A summary of the results is shown in the table:

Run	control penalty λ_o	Throughput	Maximum Instantaneous buffer places occupied	Maximum calculated cell-loss probability	Actual cell-loss over the length of the simulation run (out of 37200 cells)
Run 1	0.05	0.79	100	1	361
Run 2	15	0.835	8	10^{-18}	0
Run3	150	0.84	8	10^{-17}	0

The reference is set at $y^{ref} = 25$ cell places and $E_{overflow} = 10^{-9}$. The control loop regulates the controllable traffic such that a buffer of 25 cell-places is expected to accommodate $(1-10^{-9})$ x 100% of the total cells served.

See appendix 3.VI.D for the simulation run details.

Case E) Simulation runs without feedfonvard compensation:

Two cases (with the identical pseudorandom uncontrollable traffic sequences) were considered: with feedfonvard compensation and without. The usefulness of the case with feedforward control (that supplements the feedback) is demonstrated. Note though that our simulative experience has shown that the degree of improvement (by the case with feedforward over the case without) depends on the initial controller parameter set. The improvement in the cost functional and throughput is surnrnarised in the table below and a graphical performance comparison for the feedback signal, the control effort, the probability of loss and the throughput can be found in figure 3.26. Figure 3.27 compares the evolution of the estimated control parameters over the length of the simulation run.

Run	cost functional x10^5 (equation 3.6, page 40)	mean value of throughput
with feedfonvard compensation	0.87	0.8527
without feedforward compensation	2.46	0.8378

See appendix 3.VI.E for the simulation run details.

Case F) Simulation runs for different fixed controller output settings (open loop control):

The constant transmission-rate for the controlled source is set at fixed values starting at 15 Mbit/sec, and increasing in steps of 10 Mbit/sec for successive runs. This set of four runs uses an identical pseudorandom uncontrollable traffic sequence for each run, with a mean of approximately 100 Mbit/sec. It shows that at some critical rate there is a huge degradation in performance as the controlled flow rate is manually increased. (Note that this highlights the sensitivity of buffer occupancy to uncertainty in traffic modelling). Had an open loop CAC strategy been employed and it predicted that a source with the appropriate statistics (of say an average flow rate of 15 Mbit/sec and a peak rate of 35 Mbitlsec) is acceptable to connect to the network, then the network can experience periods of heavy losses, without the ability to minimise the loss experienced (due to the lack of any feedback from the state of the network).

A summary of the results is shown in the table:

manually fixed transmission-rate of the controlled source	15 Mbit/sec	25 Mbit/sec	35 Mbit/sec	45 Mbit/sec
losses (cells137200 celltimes)	0	47	1008	2338

See appendix 3.VI.F for the simulation run details.

3.4.3 Highlights of the performance evaluation

In the previous section, using simulation the performance of the ACFC was investigated:

- no cell-loss was experienced over a simulation run of 314 of a million celltimes (case A). The highest buffer place occupied was the 31st—well below the assumed physical buffer size of 100 cell-places [for a reference of $y^{ref} = 25$ cell places and $E_{overflow} = 10^{-5}$, the control loop regulates the controllable traffic such that a buffer of 25 cell-places is expected to accommodate $(1-10^{-5})$ x 100% = 99.999% of the total cells served].
- a high throughput can be achieved. For the simulation case **A** the throughput achieved was a high 0.89, an increase in throughput of 48% over a scheme

operating on the peak rate admission for all traffic. This could be improved even further by setting the reference higher, or $E_{overflow}$ lower (dependant on the QoS required by the user and the physical size of the buffer).

- a very tight control of cell-delav, cell-loss, and cell-delav-variation has been exhibited (as can be observed by the buffer distribution of simulation case A) whilst the throughput remained at a high 0.89.

adaptability to changed and unforeseen circumstances, and hence robustness, have been exhibited by a connection and disconnection of a 25 Mbit/sec video without any warning to the control scheme (case B). Again the throughput remained high, as the adaptive algorithm took corrective action to suit the new system conditions.

- the ability to influence local behaviour, from a higher level, was demonstrated (case *C* and D) by changing the reference value on the output and the control weight (note that $E_{overflow}$ can also be used to influence the local behaviour).
- the feedback signal is accurate, with only a slight bias observed, over the simulation run of 3/4 million celltimes. For a reference of $y^{ref} = 25$ cell places and $E_{overflow} = 10^{-5}$—so that a buffer size of 25 cell places is expected to accommodate $(1-10^{-5})\,x\,100\% = 99.999\%$ of the total cells served—the actual cell-places used for the expected number of overflow events was equal to $y = 30$.
- the feedforward signal (that supplements the feedback) is useful. Note though that our simulative experience has shown that the degree of improvement (relative to feedback acting alone) is dependant on the initial controller parameter set.
- the sensitivity of the buffer occuuancv to changes in the incoming traffic has been demonstrated.

3.5 Strong properties of ACFC

Using our control concept, based on the controllable and uncontrollable groups of traffic, combined with adaptive feedback and adaptive feedforward control methodology, we have a control solution (ACFC) that provides us with many desirable features. In this section we present the strong properties of the ACFC. Under assumptions of convergence of the adaptive regulator (proven under certain conditions, see appendix 3.V.C); network controllability (i.e. boundedness of the disturbance below the link rate); consistent and unbiased feedback measurement; and that the controlled source is saturated, the ACFC algorithm possesses the following strong properties (see discussion in appendix 3.V.D):

- The long term network utilisation is equal to unity (i.e. 100% efficiency is theoretically possible). For the proof see appendix 3.V.D. Note that our simulative experience shows that we can achieve high utilisation, and that depending on the acceptable QoS even unity utilisation is achievable.
- The long term network stability is guaranteed. For the proof see appendix **3.V.D.** Note that our simulative experience provides no counter-example of this proof, even in the short term.
- The long term QoS for uncontrollable sources is guaranteed (at least with a probability equal to $1-E_{overflow}$) to lie below certain bounds. The bound value is determined by the value of $E_{overflow}$ and the reference value. For the proof see appendix 3.V.D. The bounded components of QoS are:
    - total long term end-to-end delay.
    - the long term losses.
    - the long term CDV.

  Note i) the worst case short term loss (intersample behaviour) for a fixed size buffer can be minimised by: using a shorter sampling interval T_s; placing a limit on the maximum controllable traffic input over the sampling interval; and using a conservative regulator design, by appropriate choice of the penalty weight.

  ii) our simulation experience demonstrates the upper bound on the delay. Note that due to the long tail of the buffer distribution, see histogram in figure 3.13, the average queueing delay is usually much less than τ_i^q.
- The long term losses for the controllable group of traffic are also bounded, at the same value as for the uncontrollable sources.

Additionally, based on our simulative experience, the algorithm has demonstrated:

- Robustness against traffic uncertainties and connection and disconnections, without sacrificing the network throughput (demonstrated in case B of the simulation runs).

3.6 Conclusions

In this chapter we take into account the limitations, as seen by us, of existing schemes (see discussion in section 3.1.1) to design a robust, effective and efficient CAC and flow control policy that offers guaranteed QoS together with high utilisation of link capacity.

Specifically we take into account

1) the need to guarantee the quality of service of each and every call,
2) that there is no need to classify calls by service types, apart from the two broad groups of controllable and uncontrollable traffic,
3) that there is no need for negotiation with the network for call admission apart from the peak rate of connections requesting classification as uncontrollable,
4) that the dynamic behaviour of the system has some influence on the system performance,
5) the (possibly) long propagation delay,
6) the need for integration of the network controls,
7) the need for efficiency in use of link resources, and
8) the feasibility in the implementation of the proposed scheme.

For the designed control scheme, we have shown:

- the achievement of high network utilisation (close to unity);
- the guarantee of (worst case) bounds on the offered QoS;
- the ability to influence QoS by appropriate setting of the output reference (y^{ref}) and the overflow constant ($E_{overflow}$);
- that the controlled network's stability is guaranteed;
- that the adaptive regulator is globally convergent;
- the robustness of the control scheme to unforeseen traffic circumstances;

- that the only traffic descriptor required by the user is that of the peak rate for uncontrollable trffic;
- that the only policing required is for the peak rate of the uncontrollable traffic;
- that the only classification required is that for the uncontrollable and the controllable groups of traffic;
- that coordination of the local solution by a higher level is feasible by appropriate formulation of the control objective;
- that adaptive feedback and adaptive feedforward can be successfilly employed to solve BISDN problems.
- that a network performance measure can be usefully employed to solve network control problems;
- that the Addie-Zukerman model is robust, against model assumptions, and that it can be used for real time control.

In particular, we integrate the formulation of the CAC and flow control problems and make use of adaptive feedback and adaptive feedforward techniques. Due to our novel control formulation the network efficiency can be maintained at high levels yet the offered QoS can be regulated to defined (target) values. Since the control is implemented locally it does not suffer from any feedback delays, hence it is insensitive to the propagation delays between nodes along its path. Using analysis and simulation the ACFC performance has been investigated. Its adaptability, robustness, increased efficiency and coordinability have been demonstrated.

Additionally, the ACFC scheme is dependant on two broad traffic classifications only: the uncontrollable and controllable groups of traffic. Also the ACFC formulation is independent of the amval process model (note however that the feedback "sensor" is derived using the assumption of stationarity–our simulative experience suggests that this gives satisfactory behaviour), and does not require any user declared parameters other than the peak rate for connections identified as uncontrollable.

Finally note that adaptive control theory has been successfully employed in this section to solve the combined flow control and CAC problem. This suggests that adaptive control methodology should be considered a strong contestant for other application areas within BISDN.

Appendix 3.1: CAC, sensitivity of the cell-overflow based approaches

This is an excerpt from [52] on the cell-overflow approaches.

In this approach the cell-rate in an ATM pipe (multiplexer as seen from the output perspective) is seen as a fluid-flow. The cell-loss is calculated from the pipe overflow.

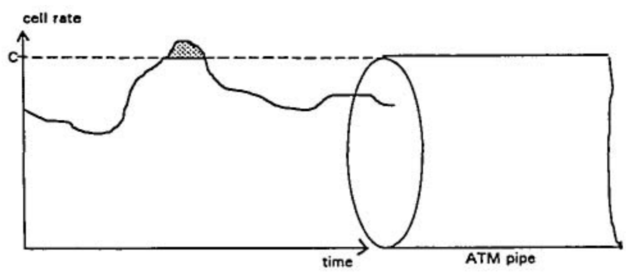


Figure 3.5. Cell rate as a fluid flow. The shaded area indicates a period of overload.

The probability of burst congestion P_b is given by the probability of the rate exceeding the link capacity C,

$$P_b = \int_C^{\infty} P(aggregate\ \text{rate} = x)dx \qquad (3.I.1)$$

which corresponds to the shaded area. The actual cell-loss rate is given by the expectation of the loss rate function, which is a ramp with gradient of one starting at C (i.e. the loss rate function is equal to $(x - C)us(x - C)$ where the symbol $us(x)$ is the unit step of x) normalised by the average arrival rate μ to yield the cell-loss probability P_l, as experienced by the users.

$$P_l = \frac{1}{\mu}\int_C^{\infty}\{P(aggregate\ \ rate = x)(x - C)\}dx \qquad (3.I.2)$$

Assuming independence between the sources, the aggregate bit rate distribution for the link is given by the convolution of the individual rate probability density function for all calls on the link. The various flavours of bufferless approximations generally differ in the means used to perform this convolution, which can sometimes be dramatically simplified. Some of these approximations will be described below.

Note that the mean probability that a cell arriving at the ATM node will overflow is usually termed the "virtual" ***cell-loss probability***. Also note that the assumption of a zero-length buffer, while providing an upper estimate on the real cell-loss probability, also means that the policing (and hence the specification) of burst periods is not required, since it is assumed that no buffer exists.

<u>On-Off envelopes</u>

Rasmussen [62], Murase [53], Appleton, Kelly Griffith and coworkers [111], [112], Esaki [113], among others have used on-off envelopes, calculated from the peak and mean rate to effectively account for worst case behaviour in terms of rate variance for a given peak and mean (note that the use of a burst model to give an upper bound on the cell-loss probability, by looking at the proportion of the instantaneous load exceeding the link capacity, is questionable [65]).

The aggregate rate distribution for N homogeneous sources is a binomial distribution, and the exact loss function, P_l, is given by Weinstein [114] as

$$P_l = \frac{1}{Np}\sum_{i=\lceil N_c \rceil}^{N}(i-N_c)\binom{N}{i}q^{N-i}p^i = \frac{1}{Np}\sum_{i=\lceil N_c \rceil}^{N}(i-N_c)bin(i,N,p) \qquad (3.I.3)$$

where

p - is the "on" probability of a source, $p=m/h$ the mean to peak bit rate ratio,

q - is the "off" probability, $q=1-p$ and

N_c - is the number of streams the link can carry at the peak rate (i.e. it must exceed the link rate for any overflow and hence any probability of loss), $N_c=C/h$ the link capacity to peak rate ratio

bin - the binomial probabilities

In a multi-class environment with K call types, each with an average bit rate m_j, peak bit rate h_j and N_j connections, the virtual cell-loss Probability formula, given by the ratio of link overflow to traffic load, becomes very complicated, as given by Suzuki et al [115].

$$P_l = \frac{\textit{Link}\text{ overflow, OF}}{\textit{traffic}\text{ load, } \rho}$$

$$P_l = \frac{1}{\sum_j^K N_j m_j} \underbrace{\sum_{n_1}^{n_1=N_1} \sum_{n_2}^{n_2=N_2} \cdots \sum_{n_K}^{n_K=N_K}}_{n_j \in \sum_{j=1}^{K} n_j h_j > C} \left(\sum_{j=1}^{K} n_j h_j - C \right) \prod_{j=1}^{K} p_j(n_j) \qquad (3.I.4)$$

where

$$p_j(n_j) = \binom{N_j}{n_j} \left(1 - \frac{m_j}{h_j}\right)^{N_j - n_j} \left(\frac{m_j}{h_j}\right)^{n_j}$$

The $p_j(n_j)$ is the probability that n_j calls of class **j** are transmitting 'bursts concurrently (a binomial distribution).

<u>Example 1</u>: Consider 2 classes of traffic, with the following possible mix of connections:

Class 1: N_1=6 h_1=20 m_1=10

Class 2: N_2=20 h_2=2 m_2=1

The line rate C=100. Then, the probability of loss surface is given by

$$P_l = \frac{1}{N_1 m_1 + N_2 m_2} \sum_{n_1} \sum_{n_2} \left(\sum_{j=1}^{2} n_j h_j - C \right) \left(\left[\binom{N_1}{n_1} \left(1 - \frac{m_1}{h_1}\right)^{N_1 - n_1} \left(\frac{m_1}{h_1}\right)^{n_1} \right] \left[\binom{n_2}{n_2} \left(1 - \frac{m_2}{h_2}\right)^{N_2 - n_2} \left(\frac{m_2}{h_2}\right)^{n_2} \right] \right)$$

Where $n_1 = 4$ and $n_2 = 11, \ldots, 20$
$n_1 = 5$ and $n_2 = 1, \ldots, 20$
$n_1 = 6$ and $n_2 = 0, 1, \ldots, 20$
since $n_j \in \sum_{j=1}^{K} n_j h_j > C$

<u>Example 2</u>: Consider 2 classes of traffic, with the following possible mix of connections:

Class 1: N_1=120 h_1=3 m_1=2

Class 2: N_2=40 h_2=10 m_2=5

The line rate C=155.

The probability of loss surface and the log of probability surface (note that log of zero has been replaced by -80 instead of -∞) are shown in figures *3.6* and **3.7.**

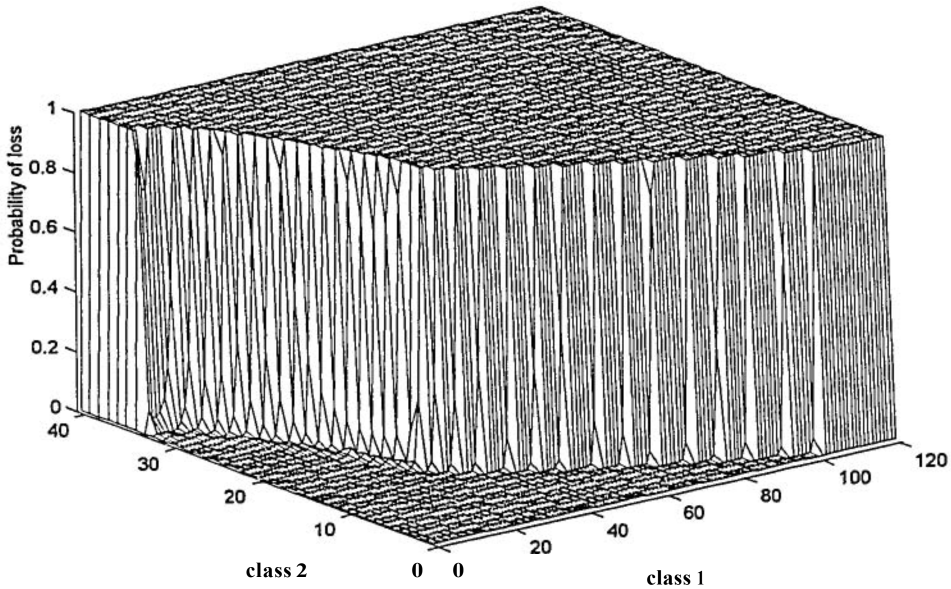


Figure *3.6.* Probability of loss surface plotted versus number of connections of class 1 and number of connections of class 2.

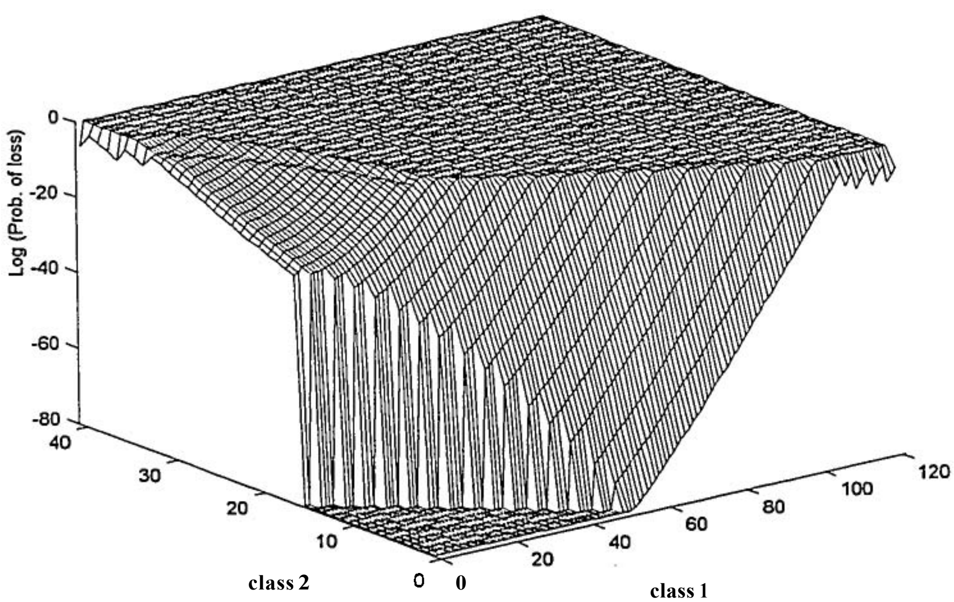


Figure **3.7.** The log of Probability of loss surface plotted versus the number of connections of class 1 and number of connections of class 2.

What is important to note is the dramatic change in the Probability of loss for even a very small change in the number of connections.

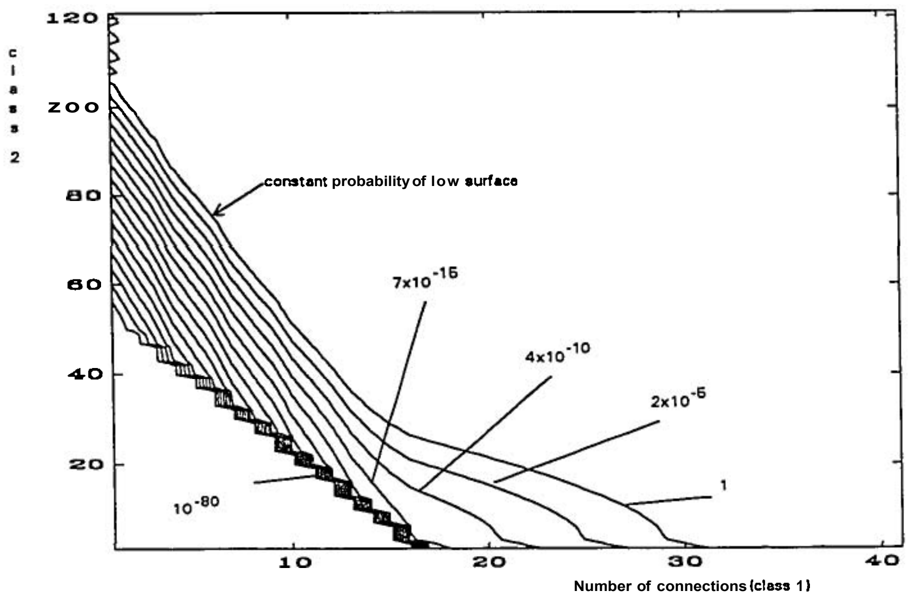


Figure **3.8.** Contour plot of the lon of the Probability of loss for different connection mix of class 1 and class **2.**

A few important observations are:

- Zero loss is achieved for the cases when $h_1 \times n_1 + h_2 \times n_2 < C^{link}$. The two extreme cases are for

  i) $n_1 < 51$; $n_2 = 0$, and

  ii) $n_2 < 15$; $n_1 = 0$.

- The nonlinearity of the acceptance surface.
- The sensitivity of the Loss surface, i.e. the very dramatic deterioration of the Probability of loss with only a very small increase in the number of connections.

The implications of these observations are

i) Linearisations of the connection acceptance surface should not be used to simplify the CAC algorithm.

ii) A very conservative CAC and policing scheme has to be used since only a few percent overallocation, or policing ineptness, will result in a very high probability of loss, at least as predicted by this model.

As can be seen from above examples considerable computational effort is required when more than one or two classes are multiplexed. To address this difficulty a number of faster approximations have been developed to evaluate the loss function. Even so, they still suffer from the same problems cited above. For further details see the report [52].

Appendix 3.11: Adaptive control using system identification techniques

The aim here is to present a brief introduction to the general concepts of adaptive control, and a concise tutorial on the approach taken in this thesis. There is vast literature on the topic, with some excellent textbooks appearing, as for example [116], [88], [117], [55], [118], [137]. Also the surveys [119] and [105] give an enlightening perspective and are worthy of perusal. See also the special issue on adaptive control that appeared in **Automatica** [120], a regular source of published papers on adaptive control.

We use topic headings of some of the terms, for easy placement.

<u>control structure</u>

The structure of a general adaptive control system is shown in figure 3.9.

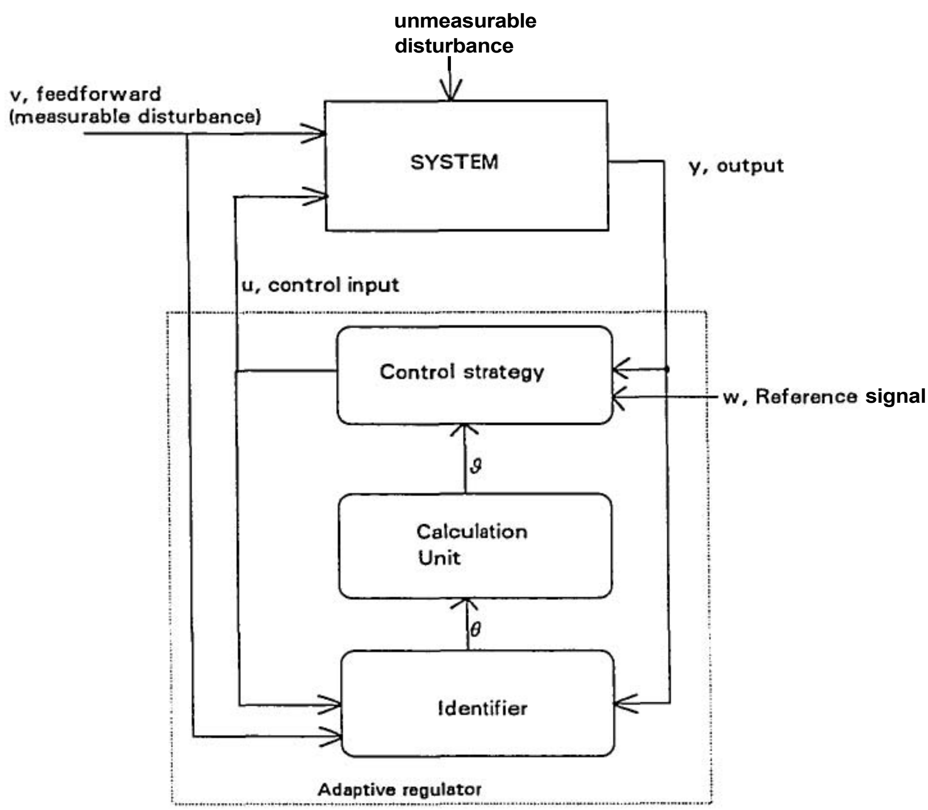


Figure 3.9. Structure of **a** general adaptive controller.

The adaptive controller consists of three conceptual blocks: the system identifier which uses measured **input/output** data to estimate the system dynamic model; the calculation unit, responsible for mapping the identified model parameters onto the controller parameters; and the actual control law. Many variants of adaptive control have been developed (see e.g. [55]). In this thesis we only consider adaptive schemes that belong to the general class of ***long-range predictive control*** [82] (LRPC). The LRPC includes the ***multi-step receding horizon*** and ***infinite horizon linear quadratic*** classes. As an aside and to avoid further confusion we point out that we do not distinguish between ***self-tuning*** and adaptive control, but rather use the more generic term of adaptive control.

system model

A model is intended to describe the system's dynamic relationship between the input and the output signals, so that good predictions of the output can be made and the effect of current and future controls can be optimised. A major decision in identification is how to parameterise the properties of the system or signal using a model of a suitable structure. We follow Ljung and Soderstrom [92].

Consider a stochastic dynamic system with input signal $\{u(t)\}$ and output signal $\{y(t)\}$. Suppose that these signals are sampled in discrete time $t=0,1,2,\ldots$ and that the sampled values can be related through the linear difference equation

$$y(t)+a_1y(t-1)+\ldots+a_{n_a}y(t-n_a)=b_1u(t-1)+\ldots+b_{n_b}u(t-n_b)+\varepsilon(t) \qquad (3.\text{II}.1)$$

where $\varepsilon(t)$ is some disturbance of unspecified character. For convenience, we make use of the backward shift (or delay) operator q^{-1} defined as $q^{-1}y(t)\equiv y(t-1)$.

Then equation (3.II.1) can be rewritten as

$$A(q^{-1})y(t)=B(q^{-1})u(t)+\varepsilon(t) \qquad (3.\text{II}.2)$$

where $A(q^{-1})$ and $B(q^{-1})$ are polynomials in the backward shift operator:

$$A(q^{-1})=1+a_1q^{-1}+\ldots+a_{n_a}q^{-n_a},$$

$$B(q^{-1})=b_1q^{-1}+\ldots+b_{n_b}q^{-n_b}$$

The model (3.II.2) can be expressed in terms of the parameter vector θ

$$\theta=[a_1\ldots a_{n_a}\ b_1\ldots b_{n_b}]^T$$

and the vector of lagged input-output data, that is the past data vector

$$\varphi(t)=[-y(t-1)\ldots-y(t-n_a)\ u(t-1)\ldots u(t-n_b)]^T$$

as follows

$$y(t) = \theta^T \varphi(t) + \varepsilon(t) \tag{3.II.3}$$

This model describes the observed variable **y(t)** as an unknown linear combination of the components of the observed vector $\varphi(t)$ plus noise. Such a model is called a *linear regression* in statistics and it is a very common type of model.

If the character of the disturbance term $\varepsilon(t)$ is not specified, then we can think of

$$\hat{y}(t) = \theta^T \varphi(t) \tag{3.II.4}$$

as a natural guess or "prediction" of what **y(t)** is going to be, having observed previous values of the input and output. The predicted output $\hat{y}(t)$ becomes a prediction in the exact statistical sense, if the sequence $\{\varepsilon(t)\}$ is a sequence of independent random variables with zero mean values, usually termed as "white noise".

Variants of this model **(3.II.1)** will give us:

- if no input is present and $\varepsilon(t)$ is white, an Autoregressive Process (AR),
- if n_a=0, a Moving Average (MA) process,
- if the noise term is not white and it is described by a MA representation (usually termed as coloured), i.e.

  $$\varepsilon(t) = C(q^{-1})e(t)$$

  then the resulting model of the well known ARMAX representation (also known as the CARMA model) is obtained as

  $$A(q^{-1})y(t) = B(q^{-1})u(t) + C(q^{-1})e(t) \tag{3.II.5}$$

  and the dynamics of the model (3.11.5) are described by the parameter vector

  $$\theta = \left[a_1 \ldots a_{n_a} \; b_1 \ldots b_{n_b} \; c_1 \ldots c_{n_c}\right]^T$$

  and the vector of past data

  $$\varphi(t) = [-y(t-1) \ldots - y(t-n_a) \; u(t-1) \ldots u(t-n_b) \; \varepsilon(t) \ldots \varepsilon(t-n_c)]^T$$

  where usually since the $\varepsilon(t)$ terms are not known they are replaced by their estimates $\hat{\varepsilon}(t)$,
- if no input signal is present, and $\varepsilon(t)$ is coloured, then an ARMA representation is obtained.

These models are the most commonly used; possibly with the addition of a state space model.

incomoration of time-delay (dead-time)

Time-delay, i.e the delay before the system input $u(t)$ can affect the output $y(t)$, can be easily incorporated into the above models. For example for a time delay of k samples, where k is the integer part of the true time delay divided by the sampling rate and rounded up, equation (3.II.1) is modified as follows

$$y(t)+a_1y(t-1)+...+a_{n_a}y(t-n_a)=b_ku(t-k)+...+b_{n_b+k}u(t-n_b-k)+\varepsilon(t) \qquad (3.II.6)$$

All other equations can be modified accordingly.

incomoration of measurable disturbances

Measurable disturbance $v(t)$ can be easily incorporated in the above models as

$$A(q^{-1})y(t)=B(q^{-1})u(t)+C(q^{-1})e(t)+\Gamma(q^{-1})v(t) \qquad (3.II.7)$$

The linear regression form is obtained by appending the parameter and past data vectors with the new parameters and past measurable disturbance respectively.

handling:of offsets

To characterise offsets, that is the situation for which a zero control signal is accompanied by a nonzero output mean an appropriate model is the CARIMA

$$A(q^{-1})y(t)=\sum_i B_i(q^{-1})u_i(t)+\sum_j \Gamma_j(q^{-1})v_j(t)+\frac{C(q^{-1})}{\Delta}e(t) \qquad (3.II.8)$$

where

$A=1-q^{-1}$ is the differencing operator, i.e. $\Delta y(t)=y(t)-y(t-1)$

This model leads to inherent integral control action in a natural way.

predictive models

Many adaptive controllers are based on predictive control laws which can be considered as generalisations of the classical Smith-predictor [121], and the minimum variance regulator of Astrom [122]. There are good reasons for this choice, such as: effective control of dead-time; easy incorporation of feedforward; easy incorporation of preprogrammed setpoints; and strong stabilisation theorems based on Linear Quadratic (LQ) ideas [82].

Early designs used a k-step-aheadprediction, arguing that $y(t+k)$ is the first output influenced by the current control choice $u(t)$. More recently **LRPC** schemes have been developed [123], [124], [125], [126], [127] in which a whole set of future outputs are predicted based on assumptions concerning current and future controls. They have been found to be robust and effective. The k-step-ahead predictor is given in Astrom [122] as

$$C(q^{-1})\hat{y}(t+k) = G(q^{-1})y(t) + B(q^{-1})F(q^{-1})u(t) \qquad (3.II.9)$$

where $G(q^{-1})$ and $F(q^{-1})$ are given by the solution of (see equation 3.26 page 168 of Astrom [122]; or the appendix 3.V.A, on how a similar equation is derived)

$$C(q^{-1}) = A(q^{-1})F(q^{-1}) + q^{-k}G(q^{-1}) \qquad (3.II.10)$$

Equation (3.II.10) is usually termed the *Diophantine* equation, and it is commonly found in the control literature. Algorithmic solutions of it are discussed in [128]. (Note that for a class of schemes, the direct or implicit schemes, it is not required to solve this equation; this is discussed in the next section of this appendix).

From (3.II.9) we can see that the prediction k-steps ahead depends on the current and past values of the input and the output. This equation is a *"positional"* predictor, in which full values of the input and output data are used to predict the output at k steps ahead. This form is usually very sensitive to offset errors. So "incremental" or "integrating" predictors, in which changes in the output are forecast, by using differences of past data, are preferred

$$\hat{y}(t+k) = y(t) + \mathcal{G}'(q^{-1})\Delta y(t) + \mathcal{F}'(q^{-1})\Delta u(t) \qquad (3.II.11)$$

where

$C(q^{-1}) = 1$ for simplicity,

$\mathcal{G}'(q^{-1})$ and $\mathcal{F}'(q^{-1})$ are appropriate polynomials obtained by straightforward manipulation of equation (3.11.9) to reflect the incremental form.

<u>predictive control laws</u>

Consider the k-step-ahead predictor of equation (3.11.9) or (3.11.11). The only unknown is the current control effort $u(t)$. Using the minimum variance (MV) [122] objective of regulating the output around the value 0, with the choice of feedback which sets the k-step-ahead prediction equal to zero, i.e. $\hat{y}(t+k) = 0$, we obtain the MV controller as

$$u(t) = -\frac{G(q^{-1})y(t)}{B(q^{-1})F(q^{-1})} \qquad (3.II.12)$$

An adaptive version of this algorithm has been proposed by Astrom and Wittenmark in their celebrated paper [86] in 1973. They defined a regression model of the same form as equation (3.11.9) but including only data up to time t

$$y(t) = G(q^{-1})y(t-k) + B(q^{-1})F(q^{-1})u(t-k) + \varepsilon(t)$$

whose unknown parameters are organised into the vector

$$\theta = [a, \ldots a\ ,\ \beta_0 \ldots \beta_{n_\beta}]^T$$

where

$$\mathcal{A}'(q^{-1}) = \alpha_0 + \ldots + \alpha_{n_\alpha} q^{-n_\alpha} = G(q^{-1}), \text{ and}$$

$$\mathcal{B}'(q^{-1}) = \beta_0 + \ldots + \beta_{n_\beta} q^{-n_\beta} = B(q^{-1})F(q^{-1})$$

and the corresponding input and output data is

$$\varphi(t) = [-y(t-\mathrm{k}) \ldots -y(t-\mathrm{k}-\mathrm{n},)\ \ u(t-\mathrm{k}) \ldots u(t-\mathrm{k}-n_b)].$$

Astrom and Wittenrnark [86] have shown that under certain conditions, a certainty equivalence law, also known as the *separation* theorem[#1] [122], which accepts the latest parameter estimates

$$u(t) = -\frac{\hat{G}(q^{-1})y(t)}{\hat{B}(q^{-1})\hat{F}(q^{-1})} \tag{3.II.13}$$

leads asymptotically to the desired MV law. Surprisingly, this is true in spite of the fact that the least squares estimate can be biased (in the presence of coloured noise) and that the noise parameters are not explicitly estimated. They called it the self-tuning regulator.

In this class of algorithms the controller parameters are identified directly (by forming an appropriate regression model; as shown above). They are called direct or implicit schemes. These can be contrasted with the class of algorithms for which the system parameters are identified firstly and then mapped into the controller parameters. They are known as indirect or explicit adaptive algorithms. The papers by Casalino et al [129] and its extension [130] attempt to unify the theory of implicit modelling and are worthy of perusal.

The MV algorithm does not provide the necessary parameters to allow its coordination, and it offers no tradeoff between control and output variances. An algorithm that offers these coordination parameters was proposed by Clarke and Gawthrop in 1975 [87] and

#1 The separation theorem *states* that the model parameters in the control law *can* be replaced by their estimates; the linear feedback is the same as would be obtained if there were no disturbances and if the state of the *system* could be measured exactly–this implies that the linear feedback design becomes a deterministiccontrol problem.

extensions in 1979 [131], the self-tuning controller. They introduced the ***generalised*** minimum variance, GMY, cost function

$$J_{GMV} = E\{[y(t+k) - w(t)]^2 + \lambda_o u^2(t)\} \tag{3.II.14}$$

where

$w(t)$ is the reference input (or desired value), and

λ_o is the control-weight which can be used to moderate the control effort and to allow for stabilising certain non-minimum-phase systems.

Design polynomials can be added to the GMV scheme, which allow its tailoring to particular applications [132]. The control law can be obtained as

$$u(t) = \frac{w(t) - \psi^*(t+k/t)}{Q(q^{-1})} \tag{3.II.15}$$

where

$\psi^*(t+k/t)$ is the predicted auxiliary output

$\psi(t) \equiv P(q^{-1})y(t)$ is the auxiliary (filtered) output, and

$Q(q^{-1})$ and $P(q^{-1})$ are the design polynomials.

These schemes are reported to be robust when applied to a broad class of systems, and effective for non-linear, time-varying, stochastic systems, commonly found in industry [133]. This is evidenced by the large number of successful applications. One disadvantage of the MV and GMV schemes is their requirement for a reasonable estimate of the time-delay k. However as mentioned earlier, this idea has been generalised to LRPC, that is claimed to be less sensitive to prior knowledge of the time delay, the model order, as well as being robust and insensitive to non minimum phase and open loop unstable systems.

<u>On line svstem identification</u>

System identification is a mature discipline in its own right and a vast array of published works and books are available, see for example the early survey by Astrom and Eykhoff [134], and the books [88], [135], [102]. In particular the book [92] focuses on recursive on line techniques. In adaptive control we make use of a recursive parameter estimator. The current input and output data are used to generate a predicted plant output. The prediction error between the predicted and the measured output can be used to refine the estimated system parameters. In this section we briefly discuss recursive *least* squares (RLS), as it and its variants are probably the most commonly used [136] on line identifiers. RLS belongs to the parametric identification group, and to the class of prediction error identification methods [92]. The basic RLS attempts to minimise a quadratic function of the prediction error.

All recursive algorithms have the same basic form

$$\hat{\theta}_{new} = \hat{\theta}_{old} + \text{correction}$$

where the correction term usually consists of a prediction error term multiplied by the adaptation gain

$$\hat{\theta}(t) = \hat{\theta}(t-1) + K(t)\varepsilon(t).$$

Different choices of adaptation gain and prediction error give us the different recursive identifier forms. For example a constant gain $K(t) = \mu$ gives us the celebrated Least Mean Squares (LMS) algorithm commonly used in adaptive signal processing. For RLS the following form is obtained

<u>RLS identifier</u>

The RLS identifier has the following form

Parameter vector update

$$\hat{\theta}(t) = \hat{\theta}(t-1) + K(t)\varepsilon(t) \tag{3.II.15a}$$

Prediction error update (see (3.11.3))

$$\varepsilon(t) = y(t) - \varphi^T(t-1)\hat{\theta}(t-1) \tag{3.II.15b}$$

Gain update

$$K(t) = \frac{P(t-2)\varphi(t-1)}{\lambda_f(t-1) + \varphi^T(t-1)P(t-2)\varphi(t-1)} \tag{3.II.15c}$$

Covariance matrix update

$$P(t-1) = \frac{1}{\lambda_f(t-1)}\left[P(t-2) - \frac{P(t-2)\varphi(t-1)\varphi^T(t-1)P(t-2)}{\lambda_f(t-1) + \varphi^T(t-1)P(t-2)\varphi(t-1)}\right]. \tag{3.II.15d}$$

Where $\lambda_f(t)$ is theforgetting factor. It is set to 1 for ordinary least squares. The forgetting factor is one of the means by which the identifier can be made to track time varying systems. The convergence properties of a number of recursive algorithms is discussed by Goodwin and Sin [88]. Implementation details are discussed in [136]. By using different choices of θ and φ, a variety of adaptive control algorithms can be implemented.

<u>convergence. convergence rate. overall stability and stabilitv margins</u>

An adaptive controller is nothing more than a special nonlinear control algorithm. It combines on-line parameter estimation with on-line control. This essential nonlinearity has been a major stumbling block in establishing global convergence and stability properties. In fact, it is surprising, given that some of these algorithms are non-linear, time-varying and operate in a stochastic environment, that anything substantial can be proven at all. Nevertheless there has been a substantial research effort, evidenced by a large number of papers, surveys and books (e.g. [88], [137], [138]) and global convergence has been proven, under certain conditions, in a few restricted classes of algorithms, e.g. [138], [55] (among which is the class that includes the one-step ahead adaptive control algorithm). The proof of ACFC, the algorithm of section 3.3.1, is presented in appendix 3.V.C.

Global convergence implies that the control output asymptotically converges to the reference value [#1]. That is, the control error of the adaptive algorithm converges to zero as time tends to infinity (i.e. $\lim_{t\to\infty}\left(y - y^{ref}\right) = 0$), and all system variables (i.e. $\{u(t)\}$ and $\{y(t)\}$) remain bounded for the given class of conditions. Stability is implied by the bounded system variables. The most restrictive condition in the proof of global convergence is the requirement that the model used to design the adaptive regulator must be at least as complex as the system to be controlled. This is certainly not practical and indeed straightforward application of the algorithm can lead to difficulties in some specific cases. Thus the current interest in adaptive (versions of) robust control [137]. There are several heuristic approaches to improve the algorithms, by for example, introducing leakage, filtering, dead zone, monitoring of excitation conditions, intentional perturbation signals, normalisation, etc. We see some of these as essential components of the "jacketing software". The convergence rate of adaptive algorithms is also proving a difficult issue. Averaging has been used [55] to give some insight into the behaviour of adaptive systems. This makes it possible to estimate the convergence rate of the parameters. However it is not possible to derive simple rules of thumb for the convergence rate. It is worth noting (as is well known) that RLS has good convergence properties.

[#1] Actually model reference output should be used. For simplicity we assume that the reference model transfer function is equal to 1.

The properties of LRPC algorithms have been discussed by a number of researchers (e.g. [125], [127]), and it is reported that they possess good robustness properties. For example the ability to handle: nonrninimum-phase systems; open-loop unstable systems; variable or unknown time-delay; and unknown system order. However on the convergence analysis side, at present, no exhaustive results are available. Note that in [127].a dynamic LRPC is obtained with a guaranteed stabilising property.

Even though the analytic global convergence proof under all conditions for all types of adaptive algorithms (if there is one) is still lacking, there is strong evidence that with a bit of care ***successful adaptive applications*** can be implemented. It is worth mentioning that there are a large number of commercially available products (e.g. the Asea Novatune, the Foxboro Exact and the Firstloop by First Control) featuring one form of adaptive control or another. Thousands of successful implementations of these products in real systems have been reported, as for example [139], some of which have been in operation for quite a few years now. Also good simulative results, as well as a large number of successful pilot applications have been reported, see for example [140], [141] over the past two decades. That is not to say that there are no unsuccessful applications; but with some engineering insight these, we believe, can be avoided.

jacketinn software

Just as with any other practical control algorithm operating on line, the adaptive algorithm's implementation requires considerably more code than that simply for implementing the mathematical relationship. Therefore, numerically stable algorithms and "diagnostic", or "jacketing" software must be provided for checking the validity of the estimation and any numerical problems [142], [143].

Conclusions

This section introduced adaptive control, for at least the areas used in this thesis. In summary, adaptive control combines recursive identification with simple, "modern" control design procedures, implemented on-line. The tuning of the control parameters is taken care of, but several design variables must still be chosen. It is an established discipline in control theoretic circles and has been extensively applied to real systems with considerable success. It has indeed reached maturity, as evidenced by the large number of successful applications and commercial products featuring these techniques.

Appendix 3.III: ATM Switch design for control option 2

A preprocessing unit can be provided to route controllable cells into the controlled buffer. It is only required to check a few bits (for example, three bits if a Payload Type code is assigned) in order to establish whether a cell is controllable or uncontrollable. If a cell is uncontrollable then it is let through immediately. However, if the cell is controllable it is routed to the controllable cells processor. At the controllable cells processor cells are sorted (time is not as critical a factor in this unit) into the controllable queues Q_i by identifying their VP identifier. Note that there is one queue Q_i for each outgoing port. See figure **3.10** for a conceptual block diagram (the controllable cells processor is omitted for clarity, and is shown in figure **3.11**).

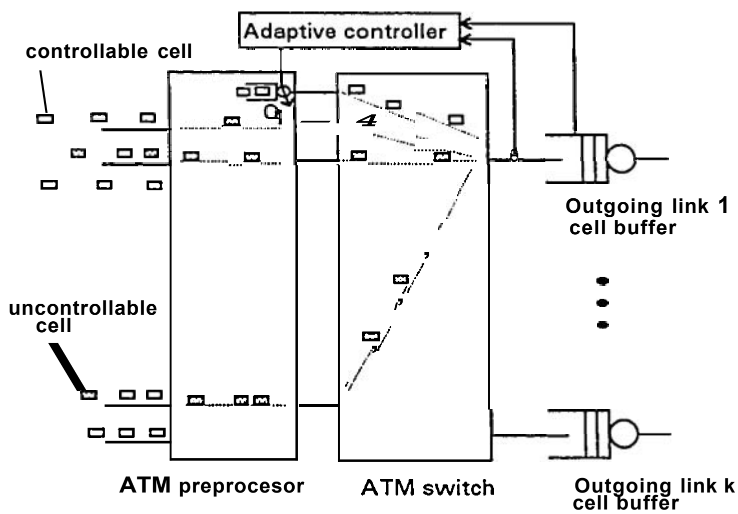


Figure 3.10. ATM switch preprocessor.

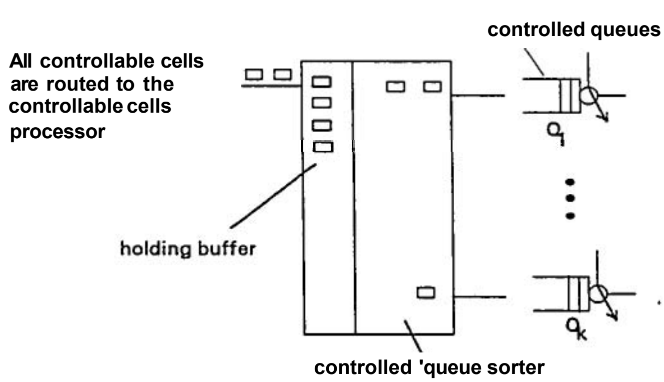


Figure **3.11.** Controllable cells processor.

Appendix 3.IV: The Addie-Zukerman model (used for our feedback signal)

Addie and Zukerman have developed a powerful set of formulae for the statistics of the unfinished work distribution in the cases of both finite [96], and infinite buffers [94], [95], [97]. Their results are based on a second order approximation and it is in terms of three parameters of the net input process: the mean, the variance and the autocorrelation sum.

Consider a FIFO single server queue. Let the time be divided into fixed length sampling intervals. The model allows arbitrary choice of interval length.

Define the following continuous random variables,

- A_n amount of work entering the system during the nth sampling interval,
- B_n amount of work that can be processed by the server during the nth sampling interval,
- Y_n net input process, given by $Y_n = A_n - B_n$, $n \geq 0$,
- V_n unfinished work at the beginning of the nth sampling interval. For the case of infinite buffer, V_n satisfies the following recurrent equation

$$V_{n+1} = (V_n + Y_n)^+, \ n \geq 0, \text{ where } V_o = 0,$$

- m mean value of Y_n i.e. $m = E\{Y_n\}$,
- σ^2 variance of Y_n, i.e. $\sigma^2 = Var\{Y_n\}$,
- U_n mutually independent Gaussian random variables with zero mean and variance equal to 1.

A Gaussian discrete-time process can be represented as

$$Y_n = m + \sum_{k=0}^{\infty} a_k U_{n-k} \tag{3.IV.1}$$

and the autocovariance ***sum*** is defined as

$$S \equiv \sum_{k=1}^{\infty} Cov(Y_n, Y_{n+k}) \tag{3.IV.2}$$

The statistics of V_∞ and the loss probability (in the case of a finite buffer) can be approximated using only the parameters m, σ^2 and S. Note that since the superposition o f j independent Gaussian processes is Gaussian, its parameters can be calculated by summing those of the superposed processes (i.e. $m = \sum_{i=1}^{J} m_i$, $\sigma^2 = \sum_{i=1}^{J} \sigma_i^2$ and $S = \sum_{i=1}^{j} S_i$), a feature which can be useful for, say, a CAC control policy based on the three measured parameters of m, σ^2 and S.

Let

$$s^* = \frac{2m}{\sigma^2 + 2S} \tag{3.IV.3}$$

and

$$\tilde{c} = \frac{erfc\left(\frac{-m}{u_1}\right) - e^{u_2 - ms^*} erfc\left(\frac{u_3 - m}{u_1}\right)}{s^* u_1\left(-u_4 erfc(u_4) + u_5 e^{-u_4^2}\right)} \tag{3.IV.4}$$

$$u_1 = \sigma\sqrt{2} \qquad u_2 = \frac{s^{*2}\sigma^2}{2} \qquad u_3 = \sigma^2 s^* \qquad u_4 = \frac{u_3}{2u_1} \qquad u_5 = \frac{1}{\sqrt{\pi}}$$

The ***stationary unfinished work distribution at steady state*** is approximately given by ([95] equation 35)

$$E\{V_\infty\} \approx -\frac{\tilde{c}}{s^*} \tag{3.IV.5}$$

and the ***probability of the stationary unfinished work distribution*** ([95] equation *34)*

$$P\{V_\infty > t\} \approx \tilde{c}e^{s^* t}. \tag{3.IV.4}$$

The p^{th} ***percentile of the stationary distribution of*** V_∞ is given by ([95] equation 36)

$$t_p \approx \begin{cases} \frac{1}{s^*}\ln\left(\frac{1-\frac{p}{100}}{\tilde{c}}\right) & \tilde{c} > 1 - \frac{p}{100} \\ 0 & \tilde{c} \le 1 - \frac{p}{100} \end{cases} \tag{3.IV.6}$$

Note that we make use of this formula for the calculation of the feedback signal. We calculate t_p the expected number of queue places, that are required to accommodate $p\%$ of the cells served by the system.

The stationary ***utilisation*** is given by

$$\rho = \frac{E\{A_\infty\}}{E\{B_\infty\}}. \tag{3.IV.7}$$

For a finite buffer the ***expected*** loss at steady state is given by ([96], equation 24)

$$E\{L_\infty\} \approx \frac{\tilde{c}e^{s^* K}\left(erf(u_2) - s^*\,\psi(-u_1)\right)}{s^*\left(1 - e^{s^* K}\right)} \tag{3.IV.8}$$

$$u_1 = \frac{\sigma^2 s^*}{2} \qquad u_2 = \frac{u_1}{\sigma\sqrt{2}} \qquad \psi(x) = \frac{\sigma}{\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}} - \frac{x}{2} erfc\left(\frac{x}{\sigma\sqrt{2}}\right)$$

$K =$ I? $- B_n$ where I? is the finite buffer size

and theprobability of loss at steady state ([96], equation 22) is

$$P_{loss} = \frac{E(L_\infty)}{E(A_\infty)}. \tag{3.IV.9}$$

Appendix 3.V: Proofs and Derivations

A) Derivation of the k-step-ahead prediction

We consider the CARIMA model (3.3). To simplify the derivation we assume that: there is a single control input; there is one feedfonvard input (measurable disturbance); the delay experienced at the output due to both inputs (control and feedfonvard) is the same; and that $C(q^{-1})$ is equal to 1, ie. $e(t)$ is white noise (a sequence of equally distributed independent, zero mean, unity variance, random variables). The assumption of $C(q^{-1})=1$ is prompted by our desire to use RLS estimators (this class of estimators does not identify the noise parameters). For convenience we reproduce the model here

$$A(q^{-1})y(t)=B(q^{-1})u(t-k)+\Gamma(q^{-1})v(t-k)+\frac{1}{\Delta}e(t) \qquad (3.V.A.1)$$

The Diophantine equation for this k-step-ahead predictor is given by

$$A_m(q^{-1})=A(q^{-1})F(q^{-1})\Delta+q^{-k}G(q^{-1}) \qquad (3.V.A.2)$$

Proof of the Diophantine eauation:

First note that since we require

$$y_m(t)=A_m(q^{-1})y(t)=A_m(1)y^{ref}(t-k) \quad \Rightarrow \quad y(t)=\frac{y_m(t)}{A_m(q^{-1})}$$

We write (3.V.A.1) in terms of $y_m(t+k)$ as follows

$$y_m(t+k)=\frac{A_m(q^{-1})\{B(q^{-1})u(t)+\Gamma(q^{-1})v(t)\}}{A(q^{-1})}+\frac{A_m(q^{-1})}{A(q^{-1})\Delta}e(t+k)$$

The first part of $y_m(t+k)$ consists of terms that depend on observed data. The last part depends on terms that can be calculated, using equation (3.V.A.1), from past data, i.e. $e(t), e(t-1),\ldots$ as well as future terms, $e(t+1),\ldots,e(t+k)$ that are independent of the data. Breaking the last part into the two terms discussed above, we can write this as

$$\frac{A_m(q^{-1})}{A(q^{-1})\Delta}e(t+k)=F(q^{-1})e(t+k)+q^{-k}\frac{G(q^{-1})}{A(q^{-1})\Delta}e(t+k)$$

where

$F(q^{-1})$ and $G(q^{-1})$ are of order $k-1$ and $n-1$ respectively.

Cancelling out $e(t+k)$ gives us (3.V.A.2), as required.

$\nabla\nabla\nabla$

Now multiply *(3.V.A1)* by $q^k F(q^{-1})\Delta$ to obtain

$$F(q^{-1})A(q^{-1})\Delta y(t+k) = F(q^{-1})B(q^{-1})\Delta u(t) + F(q^{-1})\Gamma(q^{-1})\Delta v(t) + F(q^{-1})e(t+k)$$

Replacing $F(q^{-1})A(q^{-1})A$ from the Diophantine equation gives

$$A_m(q^{-1})y(t+k) = G(q^{-1})y(t) + F(q^{-1})B(q^{-1})\Delta u(t) + F(q^{-1})\Gamma(q^{-1})\Delta v(t) + F(q^{-1})e(t+k)$$

Rewriting $G(q^{-1})y(t)$ as $(1+G'(q^{-1})\Delta)y(t)$, we obtain

$$A_m(q^{-1})y(t+k) = y(t) + G'(q^{-1})\Delta y(t) + F(q^{-1})B(q^{-1})\Delta u(t) + F(q^{-1})\Gamma(q^{-1})\Delta v(t) + F(q^{-1})e(t+k)$$

or, equivalently

$$A_m(q^{-1})y(t+k) - y(t) = \mathcal{A}(q^{-1})\Delta y(t) + \mathcal{B}(q^{-1})\Delta u(t) + \mathcal{G}(q^{-1})\Delta v(t) + F(q^{-1})e(t+k) \qquad (3.V.A.3)$$

which is in the desired incremental form. Now multiply $y(t)$ by $A_m(1)$ to robustify (see Astrom and Wittenmark [55], page *434)* we obtain the desired form.

Note that the optimal predictor of $y(t+k)$, given data up to time t, is given by

$$A_m(q^{-1})\hat{y}(t+k) = A_m(1)y(t) + \mathcal{A}(q^{-1})\Delta y(t) + \mathcal{B}(q^{-1})\Delta u(t) + \mathcal{G}(q^{-1})\Delta v(t) \qquad (3.V.A.4)$$

with error

$$\tilde{y}(t+k) = F(q^{-1})e(t+k) \qquad (3.V.A.5)$$

B) Derivation of the adaptive feedback and feedforward controller

For convenience we reproduce here the cost functional

$$\underset{\Delta u}{Min}\tfrac{1}{2}\left\{\left[A_m(q^{-1})y(t+k)-A_m(1)y^{ref}(t)\right]^2+\lambda_o(\Delta u(t))^2\right\} \qquad \lambda_o>0 \qquad (3.V.B.1)$$

and the integrating form of the k-step-ahead prediction, with $A_o(q^{-1})=1$,

$$\begin{aligned}A_m(q^{-1})y(t+k)-A_m(1)y(t)&=\\&=\mathcal{A}(q^{-1})\Delta y(t)+\mathcal{B}(q^{-1})\Delta u(t)+\sum_i\mathcal{G}_i(q^{-1})\Delta v_i(t)\end{aligned} \qquad (3.V.B.2)$$

The minimising control input of the cost functional *(3.V.B.1)* is given ***by***

$$\begin{aligned}\Delta u(t)=\frac{\beta_o}{\beta_o^2+\lambda_o}\{&A_m(1)[y^{ref}(t)-y(t)]-\mathcal{A}(q^{-1})\Delta y(t)\\&-(\mathcal{B}(q^{-1})-\beta_o)\Delta u(t)-\sum_J\mathcal{G}_j(q^{-1})\Delta v_j(t)\}\end{aligned} \qquad (3.V.B.3)$$

Proof

Rewriting *(3.V.B.2)* in terms of $A_m(q^{-1})y(t+k)$ we have

$$A_m(q^{-1})y(t+k)=\beta_o\Delta u(t)+x(t) \qquad (3.V.B.4)$$

where

$x(t)$ is all the other terms that make up $A_m(q^{-1})y(t+k)$ except $\beta_o\Delta u(t)$

Substituting *(3.*V.B.4) into *(3.*V.B.1)

$$\underset{\Delta u}{Min}\left\{\left[\beta_o\Delta u(t)+x(t)-A_m(1)y^{ref}(t)\right]^2+\lambda_o(\Delta u(t))^2\right\} \quad \lambda_o>0 \qquad (3.V.B.5)$$

and optimising with respect to $\Delta u(t)$ we obtain

$$\Delta u(t)=\frac{\beta_o}{\beta_o^2+\lambda_o}\left[A_m(1)y^{ref}(t)-x(t)\right] \qquad (3.V.B.6)$$

Upon substitution of x(t) we obtain *(3.V.B.3)* as desired. ∇∇∇

C) Proof of global convergence of the adaptive regulator

The proof of global convergence is a complex one. The basic approach we take is to exploit the properties of the parameter estimation algorithm via the Key Technical Lemma [144] (described later) together with the equations for the closed loop to prove convergence. We will therefore proceed in steps. First, for convenience, we will restate the system description and an outline of the derivation of the control algorithm. Then we will consider a deterministic system (i.e. the system parameters are known, and the system noise is zero) and derive the properties of the closed loop system. Finally we will prove the global convergence of the adaptive controller in the case of a deterministic system, given in terms of theorem 3.1. Some concluding remarks will also be offered together with a brief note on the extension of the proof to the stochastic case.

System description and the control algorithm

In this section we recall our system description, and the derivation of the control algorithm. This will set the scene for the proof of global convergence.

We considered a system described by the CARIMA model

$$A(q^{-1})\Delta y(t) = \sum_i B_i(q^{-1})\Delta u_i(t-k_i) + \sum_j \Gamma_j(q^{-1})\Delta v_j(t-k_j^f) + C(q^{-1})e(t) \qquad (3.3)$$

Using (3.V.A.1), the simplified form of the system model equation (3.3), we have derived (see appendix 3.V.A) the integrating, or incremental, k-step-ahead prediction model in a robustified form

$$A_m(q^{-1})y(t+k) - A_m(1)y(t) = \mathcal{A}(q^{-1})\Delta y(t) + \mathcal{B}(q^{-1})\Delta u(t) + \sum_j \mathcal{G}_j(q^{-1})\Delta v_j(t) \qquad (3.5)$$

Note that $A_m(q^{-1})$ is the model polynomial, that shapes the output in response to the reference $y^{ref}(t)$ as $A_m(q^{-1})y(t) = A_m(1)y^{ref}(t-\mathrm{k})$

This prediction model allows us to estimate the coefficients of the controller directly.

We then considered a simple quadratic cost functional

$$\underset{\Delta u}{Min}\, \tfrac{1}{2}\left\{\left[A_m(q^{-1})y(t+k) - A_m(1)y^{ref}(t)\right]^2 + \lambda_o(\Delta u(t))^2\right\} \qquad \lambda_o > 0 \qquad (3.6)$$

and derive the control algorithm (see appendix 3.V.B)

$$\Delta u(t) = \frac{\beta_o}{\beta_o^2 + \lambda_o}\{A_m(1)[y^{ref}(t) - y(t)] - \mathcal{A}(q^{-1})\Delta y(t) - (\mathcal{B}(q^{-1}) - \beta_o)\Delta u(t) - \sum_j \mathcal{G}_j(q^{-1})\Delta v_j(t)\} \tag{3.7}$$

Remarks: 1) The idea of bringing the predicted system output to a desired value is a naturally appealing one. Furthermore this idea is not limited to linear systems; it can be used with nonlinear systems (in some cases in a straightforward fashion).

2) This control law allows us to handle systems in which $A(q^{-1})$, $B(q^{-1})$ and $C(q^{-1})$ have common roots on the unit circle. This allows us to treat uncontrollable disturbances in the model.

3) The inclusion of measurable disturbance signals, v,(t), in the model gives feedforward action, **i.e.** the control input is instantaneously adjusted (prior to the disturbances having an effect on the system output), to compensate for changes in the measurable disturbances.

Properties of the deterministic controller

a) Using the control law of equation (3.7) and the system model **(3.3),** the resulting closed-loop system (see figure 3.12) is described by

$$\left\{A_m(q^{-1})B(q^{-1}) - \frac{\lambda_o}{\beta_o}\Delta A(q^{-1})\right\}y(t+k) = A_m(1)B(q^{-1})y^{ref}(t) \tag{3.V.C.1}$$

and

$$\left\{A_m(q^{-1})B(q^{-1}) - \frac{\lambda_o}{\beta_o}\Delta A(q^{-1})\right\}u(t) = A_m(1)A(q^{-1})y^{ref}(t) \tag{3.V.C.2}$$

In our analysis, for clarity and ease of presentation, the effect of the **feedforward** signals [v_1(t) the average, over one control interval, of the uncontrollable traffic flow and $v_2(t)$ the average, over one control interval, of the buffer size] are neglected since these are always bounded (by CAC and the finite buffer size). If one wishes to include them the analysis follows in exactly the same fashion as for the input $u(t)$ by using the principle of superposition (**i.e.** we can simply add the individual responses in order to obtain the overall system response due to all inputs). Also, since a deterministic system is assumed the system noise is zero.

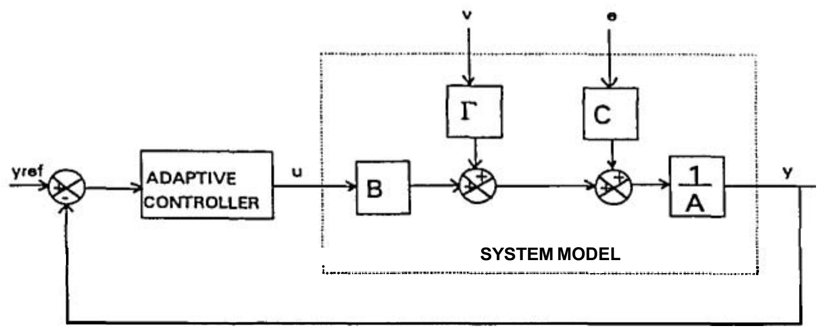


Figure *3.12.* Closed loop system block diagram

Proof:

from the k-step ahead prediction (3.5)

$$-\mathcal{A}(q^{-1})\Delta y(t)-\mathcal{B}(q^{-1})\Delta u(t)-\sum_j \mathcal{G}_j(q^{-1})\Delta v_j(t)=A_m(1)y(t)-A_m(q^{-1})y(t+k)$$

substituting into the control law (3.7), we have

$$\frac{\lambda_o}{\beta_o}(1-q^{-1})u(t)=A_m(q^{-1})y(t+k)-A_m(1)y^{ref}(t)$$

and thus

$$\left\{A,(q^{-1})y(t+k)-A_m(1)y^{ref}(t)-\frac{\lambda_o}{\beta_o}(1-q^{-1})u(t)\right\}=0. \qquad (3.V.C.3)$$

We seek relationships between y^{ref} and $y(t)$ and $u(t)$.

Multiplying *(3*.V.C.3) by $B(q^{-1})$

$$\left\{B(q^{-1})A_m(q^{-1})y(t+k)-A_m(1)B(q^{-1})y^{ref}(t)-\frac{\lambda_o}{\beta_o}(1-q^{-1})B(q^{-1})u(t)\right\}=0$$

and using the system model equation $A(q^{-1})y(t)=B(q^{-1})u(t-k)$ (for zero noise and no feedforward signals)

$$\left\{B(q^{-1})A_m(q^{-1})y(t+k)-A_m(1)B(q^{-1})y^{ref}(t)-\frac{\lambda_o}{\beta_o}(1-q^{-1})A(q^{-1})y(t+k)\right\}=0$$

and thus

$$\left\{A_m(q^{-1})B(q^{-1})-\frac{\lambda_o}{\beta_o}(1-q^{-1})A(q^{-1})\right\}y(t+k)=A_m(1)B(q^{-1})y^{ref}(t) \qquad (3.V.C.4)$$

as required for the relationship between y^{ref} and $y(t)$.

Now, multiplying (3.V.C.3) by $A(q^{-1})$

$$\left\{A(q^{-1})A_m(q^{-1})y(t+k)-A_m(1)A(q^{-1})y^{ref}(t)-\frac{\lambda_o}{\beta_o}(1-q^{-1})A(q^{-1})u(t)\right\}=0$$

and using the system equation

$$\left\{A_m(q^{-1})B(q^{-1})u(t)-A_m(1)A(q^{-1})y^{ref}(t)-\frac{\lambda_o}{\beta_o}(1-q^{-1})A(q^{-1})u(t)\right\}=0$$

and thus

$$\left\{A_m(q^{-1})B(q^{-1})-\frac{\lambda_o}{\beta_o}(1-q^{-1})A(q^{-1})\right\}u(t)=A_m(1)A(q^{-1})y^{ref}(t) \qquad (3.V.C.5)$$

as required for the relationship between y^{ref} and $u(t)$.

VVV

b) The resulting closed-loop system has bounded inputs and outputs (and hence the system is stable) provided that:

i) All modes of the "inverse" models (relating y^{ref} to $u(t)$ and y^{ref} to $y(t)$), i.e. the zeros of the polynomial

$$A_m(z^{-1})B(z^{-1})-\frac{\lambda_o}{\beta_o}(1-z^{-1})A(z^{-1})$$

lie inside or on the unit circle. Additionally any modes of the "inverse" model on the unit circle have a Jordan block size of 1.

note: z is the complex variable of the z-transform

ii) All controllable modes of the "inverse" models relating y^{ref} to $u(t)$ and y^{ref} to $y(t)$, i.e. the zeros of the transfer functions

$$\frac{1}{B(z^{-1})}\left[A_m(z^{-1})B(z^{-1})-\frac{\lambda_o}{\beta_o}(1-z^{-1})A(z^{-1})\right]$$

$$\frac{1}{A(z^{-1})}\left[A_m(z^{-1})B(z^{-1})-\frac{\lambda_o}{\beta_o}(1-z^{-1})A(z^{-1})\right]$$

lie strictly inside the unit circle

Note that by appropriate choice of λ_o and $A_m(q^{-1})$ one can ensure the stability of the system (even for systems which are neither stable or stably invertible).

Proof:

We first note that the ARMA models given by equations (3.V.C.4) and (3.V.C.5) relating y^{ref} to $u(t)$ and y^{ref} to $y(t)$ are equivalent to observable state-space models. Then, if conditions b.i) to b.iii) are satisfied, we can use the following property of observable state space models:

> For an observable linear time-invariant system of input $u_o(t)$ and output $y_o(t)$, satisfying conditions b.i) to b.iii), there exist constants $(0 \le m_1 < \infty,\ 0 \le m_2 < \infty)$ which are independent of t, such that the linear boundedness between input and output are satisfied, i.e.
>
> $$|y_o(t)| \le m_1 + m_2 \max_{1\le\tau\le N}|u_o(\tau)| \quad \text{for all } 1 \le t \le N$$
>
> Proof See Goodwin and Sin [88], Lemma B.3.3, page 486.

Therefore since $\{y(t)\}$ and $\{u(t)\}$ are linearly bounded by $\{y^{ref}\}$ (i.e. a guaranteed bounded sequence) then these sequences must also be bounded. Note that the feedforward sequences {v, (t)} and $\{v_2(t)\}$ are bounded by the structure of the control system. Hence the system is stable.

VVV

Proof of converaence of the deterministic adaptive regulator

For the deterministic adaptive control case, the regulator is globally convergent (that is the system is closed loop stable and it asymptotically achieves zero tracking error). This is stated as theorem 3.1.

Theorem 3.1 Subject to assumption 3.1 below, the adaptive control algorithm **(3.7)** when applied to the system (3.3) is globally convergent, that is the following properties are satisfied:

i) $\{y(t)\}$, $\{u(t)\}$, $\{v_1(t)\}$ and $\{v_2(t)\}$ are bounded sequences for all t.

ii) $\lim_{t\to\infty}\left[y(t)-y^{ref}(t)\right]=0$.

Remark: The boundedness of the sequences also implies global stability.

Assumption 3.1

i) the time delay k is known

ii) **An** upper bound for the orders of the polynomials in the system model (3.3) is known.

iii) conditions b.i) and b.ii) of the deterministic case, described earlier, apply.

iv) the scalar $\kappa_o = \frac{\lambda_o}{\beta_o}$ **is** specified, such that in the cost function the control penalty is given by $\lambda_o = \kappa_o\beta_o$ (since λ_o is required to be positive in the cost function *(3.6)* then κ_o must have the same sign as β_o).

v) sequences $\{v_1(t)\}$ and $\{v_2(t)\}$ are bounded.

Note that:

- assumption i) is required due to the look ahead nature of the controller;
- assumption ii) is of importance since it allows the system order to be overestimated thus ensuring that the controller has adequate degrees of freedom;
- assumption iii) is necessary in order to achieve perfect tracking and closed loop stability (these are the same assumptions required for the deterministic case; we cannot expect to do better than the deterministic case);

- assumption iv) is necessary to ensure that the original cost function is satisfied, otherwise a cost function of the same form is satisfied, but with a different control penalty, due to the possible difference between β_o and its estimate; and
- assumption v) is always satisfied by the design of the ACFC algorithm.

Using these assumption the proof of global convergence of the regulator follows.

<u>Proof:</u>

The k-step-ahead prediction (3.5) is

$$A_m(q^{-1})y(t) - A_m(1)y(t-\mathrm{k}) = \mathcal{A}(q^{-1})\Delta y(t-\mathrm{k}) + \mathcal{B}(q^{-1})\Delta u(t-\mathrm{k})$$

where again for simplicity of exposition (without any loss of generality) the feedfonvard terms have been set to zero.

We wish to manipulate this expression in a form that will be linear in the minimising control effort of equation (3.7), i.e.

$$A_m(q^{-1})y(t+k) - A_m(1)y(t) - \mathcal{A}(q^{-1})\Delta y(t) - \mathcal{B}'(q^{-1})\Delta u(t-1) - \beta_o \Delta u(t) = 0$$

where $\mathcal{B}'(q^{-1}) = q\left[\mathcal{B}(q^{-1}) - \beta_o\right]$.

Since the minimising control law $\Delta u(t)$ for the desired objective as derived earlier is

$$\Delta u(t) = \frac{\beta_o}{\beta_o^2+\lambda_o}\left\{A_m(1)[y^{ref}(t) - y(t)] - \mathcal{A}(q^{-1})\Delta y(t) - (\mathcal{B}(q^{-1}) - \beta_o)\Delta u(t)\right\}$$

and our objective is to obtain a direct adaptive control algorithm, which is linear in the parameters, we multiply the above equation by $\dfrac{\beta_0}{\beta_o^2 + \lambda_0}$ to obtain

$$\frac{\beta_0}{\beta_o^2+\lambda_0}\left\{A_m(q^{-1})y(t+k) - A_m(1)y(t) - \mathcal{A}(q^{-1})\Delta y(t) - \mathcal{B}'(q^{-1})\Delta u(t-1)\right\} - \frac{\beta_o^2}{\beta_o^2+\lambda_0}\Delta u(t) = 0$$

Now, manipulating to obtain $\Delta u(t)$ [by adding and subtracting $\dfrac{\lambda_0}{\beta_o^2+\lambda_0}\Delta u(t)$]

$$\Delta u(t)=\frac{\beta_0}{\beta_o^2+\lambda_0}\left\{A_m(q^{-1})y(t+k)-A_m(1)y(t)+\frac{\lambda_o}{\beta_0}\Delta u(t)-\mathcal{A}(q^{-1})\Delta y(t)-\mathcal{B}'(q^{-1})\Delta u(t-1)\right\}$$

we obtain $\Delta u(t)$ in terms of a linear control law

$$\Delta u(t)=\overline{\varphi}^T(t)\theta_o \qquad (3.V.C.6)$$

where

the parameter vector is

$$\theta_0=\left[\frac{\beta_o}{\beta_o^2+\lambda_o}\quad \frac{\beta_o}{\beta_o^2+\lambda_o}\alpha_1\ ,\ldots,\ \frac{\beta_o}{\beta_o^2+\lambda_o}\alpha_{n_a}\quad \frac{\beta_o^2}{\beta_o^2+\lambda_o}\ ,\ldots,\ \frac{\beta_o}{\beta_o^2+\lambda_o}\beta_{n_\beta}\right]^T \qquad (3.V.C.7)$$

and the data vector is

$$\overline{\varphi}(t)=\left[\left\{A_m(q^{-1})y(t+k)-A_m(1)y(t)+\frac{\lambda_o}{\beta_o}\Delta u(t)\right\}\ -\Delta y(t)\ldots-\Delta y(t-n+1)\right.$$
$$\left.-\Delta u(t-1)\ldots-\Delta u(t-n+1)\right]^T \qquad (3.V.C.8)$$

Now the derived **minimising** optimal control can be expressed in a linear form

$$\Delta u^*(t)=\varphi^T(t)\theta_o \qquad (3.V.C.9)$$

where

$$\varphi(t)=[\{A_m(1)y^{ref}(t)-A_m(1)y(t-k)\}\ -\Delta y(t)\ldots-\Delta y(t-\mathrm{n}+1)$$
$$-\Delta u(t-1)\ \ldots-\Delta u(t-\mathrm{n}+1)]^T \qquad (3.V.C.10)$$

In order to deal with the remote possibility of division by zero in finding $\Delta u(t)$ the following mild conditions are assumed: there exists a lower bound $\left|\frac{\beta_0}{\beta_o^2+\lambda_0}\right|_{\min}$; and the ratio $\frac{\lambda_o}{\beta_0}$ is given (assumption 3.1, part iv) with the sign of β_0 known (to ensure that λ_o is positive as required in the cost function). Note that these are of more significance in a theoretical rather than a practical sense.

Now we can state the adaptive control algorithm as follows.

Step 1:

The RLS is

$$\hat{\theta}(t)=\hat{\theta}(t-1)+\frac{P(t-k-1)\overline{\varphi}(t-k)}{1+\overline{\varphi}^{T}(t-k)P(t-k-1)\overline{\varphi}(t-k)}\left(\Delta u(t-k)-\overline{\varphi}^{T}(t-k)\hat{\theta}(t-1)\right)$$

$$P(t-k)=P(t-k-1)-\frac{P(t-k-1)\overline{\varphi}(t-k)\overline{\varphi}^{T}(t-k)P(t-k-1)}{1+\overline{\varphi}^{T}(t-k)P(t-k-1)\overline{\varphi}(t-k)}$$

$$P(0)=kI, \qquad 0<k<\infty; \qquad \theta(0)=\theta_{initial}$$

and $\hat{\vartheta}_1(t)sign(\beta_o)$ is constrained to lie above $\left|\frac{\beta_0}{\beta_o^2+\lambda_0}\right|_{\min}$.

Step 2:

The control law is generated from (by substituting the parameters with their estimates)

$$\Delta u(t)=\varphi^{T}(t)\hat{\theta}(t).$$

The convergence properties of this algorithm now follow immediately.

The convergence properties of the parameter estimator algorithm are:

i) $$\lim_{t\to\infty}\frac{\overline{\varphi}^{T}(t-k)\tilde{\vartheta}(t-1)}{\left[1+\kappa_2\overline{\varphi}^{T}(t-k)\overline{\varphi}(t-k)\right]^{1/2}}=0$$

where

$\tilde{\theta}(t)=\hat{\theta}(t)-\theta_o$, and

$\kappa_2=\lambda_{\max}P(-1)$ and A, is the maximum eigenvalue.

ii) $$\lim_{t\to\infty}\left\|\hat{\theta}(t)-\hat{\theta}(t-k)\right\|=0$$ for any finite k.

Proof See Goodwin and Sin [88], page 60.

Thus from i) and ii) it follows that

$$\lim_{t\to\infty}\frac{\overline{\varphi}^T(t-k)\tilde{\vartheta}(t-k)}{[1+\kappa_2\overline{\varphi}^T(t-k)\overline{\varphi}(t-k)]^{1/2}}=0$$

and

$$\overline{\varphi}^T(t-k)\tilde{\theta}(t-k)=\overline{\varphi}^T(t-k)\hat{\theta}(t-k)-\overline{\varphi}^T(t-k)\theta_o$$

$$=\overline{\varphi}^T(t-k)\hat{\theta}(t-k)-\Delta u(t-k) \qquad \text{(using 3.V.C.6)}$$

$$=\overline{\varphi}^T(t-k)\hat{\theta}(t-k)-\varphi^T(t-k)\hat{\theta}(t-k)$$

[and using **3.V.C.8**, the definition of $\overline{\phi}(t)$, and 3.V.C. ***10***, the definition of $\phi(t)$]

$$=\hat{\theta}_1(t-k)\left[A_m(q^{-1})y(t)-A_m(1)y(t-k)+\frac{\lambda_o}{\beta_o}\Delta u(t-k)-A_m(1)y^{ref}(t)+A_m(1)y(t-k)\right]$$

$$=\hat{\theta}_1(t-k)\left[A_m(q^{-1})y(t)+\frac{\lambda_o}{\beta_o}\Delta u(t-k)-A_m(1)y^{ref}(t)\right].$$

Observe that $\hat{\vartheta}_1(t)sign(\beta_o)$ is constrained to lie above $\left|\frac{\beta_0}{\beta_o^2+\lambda_0}\right|_{\min}$ and hence

$$\left|\frac{\overline{\varphi}^T(t-k)\tilde{\vartheta}(t-k)}{\hat{\vartheta}_1(t-k)[1+\kappa_2\overline{\varphi}^T(t-k)\overline{\varphi}(t-k)]^{1/2}}\right|\le\left|\frac{\overline{\varphi}^T(t-k)\tilde{\vartheta}(t-k)}{\left|\frac{\beta_o}{\beta_o^2+\lambda_o}\right|_{\min}[1+\kappa_2\overline{\varphi}^T(t-k)\overline{\varphi}(t-k)]^{1/2}}\right|$$

therefore

$$\lim_{t\to\infty}\frac{\eta(t)}{[1+\kappa_2\overline{\varphi}^T(t-k)\overline{\varphi}(t-k)]^{1/2}}=0$$

where

$$\eta(t)=A_m(q^{-1})y(t)+\frac{\lambda_o}{\beta_o}\Delta u(t-k)-A_m(1)y^{ref}(t). \qquad (3.V.C.11)$$

Multiplying both sides of 3.V.C.11 by $A(q^{-1})$, we have

$$A(q^{-1})\eta(t) = A_m(q^{-1})A(q^{-1})y(t) + \frac{\lambda_o}{\beta_o}A(q^{-1})\Delta u(t-k) - A_m(1)A(q^{-1})y^{ref}(t)$$

and since $A(q^{-1})y(t) = B(q^{-1})u(t-\mathrm{k})$

$$A(q^{-1})\eta(t) = \left[A_m(q^{-1})B(q^{-1}) + \frac{\lambda_o}{\beta_o}A(q^{-1})\Delta\right]u(t-k) - A_m(1)A(q^{-1})y^{ref}(t). \quad (3.V.C.12)$$

Also multiplying both sides of 3.V.C.11 by $B(q^{-1})$, we have

$$B(q^{-1})\eta(t) = A_m(q^{-1})B(q^{-1})y(t) + \frac{\lambda_o}{\beta_o}B(q^{-1})\Delta u(t-k) - A_m(1)B(q^{-1})y^{ref}(t)$$

and again using $A(q^{-1})y(t) = B(q^{-1})u(t-\mathrm{k})$

$$B(q^{-1})\eta(t) = \left[A_m(q^{-1})B(q^{-1}) + \frac{\lambda_o}{\beta_o}A(q^{-1})\Delta\right]y(t) - A_m(1)B(q^{-1})y^{ref}(t). \quad (3.V.C.13)$$

Since $y^{ref}(\mathrm{t})$ is a bounded sequence, from part iii) of the assumptions, and from above equations [(3.V.C.12) and (3.V.C.13)] it follows that the growth of sequences $\{y(t)\}$ and $\{u(t-\mathrm{k})\}$ is stably related to the growth of $\eta(t)$.

<u>Proof</u>: there exist constants $0 \le m_1 < \infty$, $0 \le m_2 < \infty$, $0 \le m_3 < \infty$, $0 \le m_4 < \infty$ which are independent of t, such that (using the property of observable state space models, given on page 89)

i) the growth of $\{u(t-\mathrm{k})\}$ is stably related to the growth of $\eta(t)$

$$|\Delta u(t_x - k)| \le m_1 + m_2 \max_{1\le\tau\le t}\|\eta(\tau)\| \qquad \text{for all } 1 \le t_x \le t$$

and

ii) the growth of $\{y(t)\}$ is stably related to the growth of $\eta(t)$

$$|y(t_x)| \le m_3 + m_4 \max_{1\le\tau\le t}\|\eta(\tau)\| \qquad \text{for all } 1 \le t_x \le t\ .$$

♦♦♦

Next we apply the following Key Technical Lemma.

<u>Kev technical Lemma (KTL)</u>

If the following conditions are satisfied for some given sequences $\{s(t)\}$, $\{\sigma(t)\}$, $\{b_1(t)\}$, $\{b_2(t)\}$:

1. $$\lim_{t\to\infty} \frac{s(t)^2}{\left[b_1(t)+b_2(t)\sigma^T(t)\sigma(t)\right]} = 0$$

where $\{s(t)\}$, $\{b_1(t)\}$, $\{b_2(t)\}$ are real scalar sequences and $\{\sigma(t)\}$ is a real (p x 1) vector sequence.

2. Uniform boundedness condition

$$0 < b_1(t) < K < \infty \quad \text{and} \quad 0 < b_2(t) < K < \infty$$

for all $t \geq 1$.

3. Linear boundedness condition

$$\|\sigma(t)\| \leq C_1 + C_2 \max_{1\leq\tau\leq t} |s(\tau)|$$

where $0 < C_1 < \infty$ and $0 < C_2 < \infty$.

Then, it follows that

i) $\lim_{t\to\infty} s(t) = 0$

ii) $\{\|\sigma(t)\|\}$ is bounded.

Proof See Goodwin, Ramadge and Caines [144].

Now we show that the 3 conditions of the KTL are satisfied:

a) Condition 1 of the KTL is satisfied with $s(t) = \eta(t)$, $\sigma(t) = \overline{\varphi}(t)$, $b_1(t) = 1$ and $b_2(t) = \kappa_2$.

b) condition 2) of the KTL is satisfied, since $b_1(t) = 1$ and $b_2(t) = \kappa_2$.

c) condition 3) of the KTL is satisfied, as shown below.

Proof of condition 3 of KTL:

Using 3.V.C.8, the definition of $\overline{\phi}(t)$, we can show that

$$\|\overline{\varphi}(t-k)\| \leq \left| A_m(q^{-1})y(t) - A_m(1)y(t-k) + \frac{\lambda_o}{\beta_0}\Delta u(t-k)\right| +$$

$$p\left\{\max(m_1,m_3)+\max(m_2,m_4)\times\max_{1\le\tau\le t}|\eta(\tau)|\right\}. \qquad (3.V.C.14)$$

But

$$|\eta(t)|\ge\left|A_m(q^{-1})y(t)+\frac{\lambda_o}{\beta_o}\Delta u(t-k)\right|-\left|A_m(1)y^{ref}(t)\right|\ge\left|A_m(q^{-1})y(t)+\frac{\lambda_o}{\beta_o}\Delta u(t-k)\right|-m_5$$

therefore

$$|\eta(t)|+m_5\ge\left|A_m(q^{-1})y(t)+\frac{\lambda_o}{\beta_o}\Delta u(t-k)\right|$$

and

$$\begin{aligned}
m_7+m_8\max_{1\le\tau\le t}|\eta(\tau)| \quad &\ge \max(m_5,A_m(1)m_3)+\left\{\max(1,A_m(1)m_4)\max_{1\le\tau\le t}|\eta(\tau)|\right\}\\
&\ge |\eta(t)|+m_5+\left|A_m(1)\left\{m_3+m_4\max_{1\le\tau\le t}|\eta(\tau)|\right\}\right|\\
&\ge |\eta(t)|+m_5+|A_m(1)y(t-k)|\\
&\ge \left|A_m(q^{-1})y(t)+\frac{\lambda_o}{\beta_0}\Delta u(t-k)\right|+|A_m(1)y(t-k)|\\
&\ge \left|A_m(q^{-1})y(t)-A_m(1)y(t-k)+\frac{\lambda_o}{\beta_0}\Delta u(t-k)\right|.
\end{aligned}$$

Substituting into **3.V.C.**14

$$\|\overline{\varphi}(t-k)\|\le m_7+m_8\max_{1\le\tau\le t}|\eta(\tau)|+p\left\{\max(m_1,m_3)+\max(m_2,m_4)\times\max_{1\le\tau\le t}|\eta(\tau)|\right\}$$

therefore

$$\|\overline{\varphi}(t-k)\|\le C_1+C_2\max_{1\le\tau\le t}|\eta(\tau)|$$

where $0<C_1<\infty$ and $0<C_2<\infty$

and this proves that condition **3** of the KTL is also satisfied [namely that $\|\overline{\varphi}(t-k)\|$ is bounded by $\eta(t)$]. ◆◆◆

Since the KTL is satisfied, we conclude that the sequences $\{y(t)\}$ and $\{u(t)\}$ are bounded and that the $\lim_{t+w} \eta(t)=0$.

And since

$$\Delta u(t)-\Delta u^*(t)=\overline{\varphi}^T(t)\theta_o-\varphi^T(t)\theta_o \qquad \text{using (3.V.C.10) and (3.V.C.11)}$$

$$=\theta_{o,1}(t)\left[A_m(q^{-1})y(t+k)+\frac{\lambda_o}{\beta_o}\Delta u(t)-A_m(1)y^{ref}(t+k)\right]$$

$$=\vartheta_{o,1}\eta(t+k)$$

it follows that $\lim_{t\to\infty}[\Delta u(t)-\Delta u^*(t)]=0$.

It now follows that since in the long term the control input converges to the optimal control input, then the optimality condition also implies that in the long term the output converges to the desired model reference output, i.e.

$\lim_{t\to\infty}\left[A_m(q^{-1})y(t)-A_m(1)y^{ref}(t-k)\right]=0$. To enhance the clarity of the proofs on the strong properties of the ACFC (given in appendix 3.V.D) we will assume, without any loss of generality, that the model reference model is equal to unity and the time delay is zero (i.e. $A_m(q^{-1})=1$ and $k=0$) to obtain $\lim_{t\to\infty}\left[y(t)-y^{ref}(t)\right]=0$ as given in theorem 3.1.

This completes the proof of global convergence for the deterministic adaptive control system described above. Note that closely related algorithms to that described above have been proposed by [87] and [88].

$\nabla\nabla\nabla$

Discussion

The key conclusions are that:

- Closed-loop stability is achieved.
- The output tracking error asymptotically goes to zero.

The extension of the proof to the stochastic case follows along the same lines as the above proof, however the analysis tools are now derived from the probabilistic framework. For example, see Goodwin and Sin [88], section 11.3.4 for a rigorous analysis of convergence of some stochastic adaptive control algorithms.

D) Proof of some of the properties of ACFC

Define the following continuous random variables,

$u(t)$	the controllable input entering the system (or equivalently the controller, ACFC, output)
$v_1(t)$	the uncontrollable traffic entering the system (i.e. the disturbance)
$y(t)$	the system output, given in terms of the p^th percentile of the stationary distribution of V_∞ (see equation 3.IV.6, page 80)
y^{ref}	the reference value, also given in terms of the p^th percentile of the stationary distribution of V_∞
T	sampling period
A_n	amount of work entering the system during the nth sampling interval, i.e. $A_n = \int_{nT}^{(n+1)T} \{u(t) + v_1(t)\}dt$
B_n	amount of work that can be processed by the server during the nth sampling interval, i.e. $B_n = C^{link}$ the link cell server rate in cells/time unit T
Y_n	net input process, given by $Y_n = A_n - B_n$, $n \geq 0$,
V_n	unfinished work at the beginning of the nth sampling interval. For the case of infinite buffer, V_n satisfies the following recurrent equation $V_{n+1} = (V_n + Y_n)^+$, $n \geq 0$, where $V_o = 0$,
m	mean value of Y_n, i.e. $m = E\{Y_n\}$, where $E\{Y_n\} = E\{A_n\} - E\{B_n\} = E\{u(t) + v_1(t)\} - C^{link}$,
ρ	network utilisation (unity indicates 100% utilisation),
$E_{overflow}$	Expected overflow, for example 1 cell out of 10^9 is expected to overflow above the reference value of y^{ref}.
k_i	a constant

The following proofs are based on the <u>assumptions i - iv</u>, given below:

assumption i) The adaptive ACFC controller has converged (i.e. $\lim_{t\to\infty}(\dot{y} - y^{ref}) = 0$, and the system variables $\{u(t)\}$ and $\{y(t)\}$ remain bounded). See proof in section 3.V.C and discussion in section 3.3.2.1.

assumption ii) the uncontrollable source variable $\{v_1(t)\}$ remains bounded (using **CAC** algorithm 1 the total peak rate of the uncontrollable sources does not exceed the link rate, i.e. $v_1(t) \le h_{total} \le C^{link}$)

assumption iii) the controlled source is saturated, i.e. there is an infinite supply of cells to the controllable queue at all times.

assumption iv) the feedback measurement is consistent and unbiased.

Note: 1) the third assumption is only required for the proof of unity utilisation

2) the unbiasedness of the feedback measurement introduced in the last assumption can be removed, at the expense of a more complicated proof. The output will still be bounded, however the bias must be added to it.

<u>a) The long term network utilisation is equal to unity</u>.

<u>Proof</u> Using assumptions i, ii and iv

$$E\{y(t)\} = E\{y^{ref}\} \;\Rightarrow\; E\{V_{n+1}\} = E\{V_n\} = k_1 \times E\{y^{ref}\} = k_2,$$

and therefore since

$$E\{V_{n+1}\} = E\{V_n\} + E\{Y_n\} \;\Rightarrow\; E\{Y_n\} = 0$$

now, using assumption **iii** (the controlled source is saturated) and since the excess work in the system is equal to zero the system utilisation is unity, i.e. $\rho = 1$ VVV

<u>b) The long term network **stability** is guaranteed</u>.

<u>Proof</u> From assumption **i** and **ii,** the system variables (i.e. $\{u(t)\}$, $\{y(t)\}$ and $\{v_1(t)\}$) are bounded.

Hence the system is globally stable. VVV

<u>c) The **long** term buffer occupancy is bounded **(with** a probability of</u> $1 - E_{overflow}$).

<u>Proof</u> Using assumptions i, ii and **iv,** $E\{y(t)\} = E\{y^{ref}\}$ therefore the buffer occupancy is bounded below y^{ref} with a probability of $(1 - E_{overflow})$.

∇∇∇

<u>d) The long term QoS for uncontrollable sources is **guaranteed**.</u>

<u>i) The total long term end-to-end delay is bounded.</u>

<u>Proof</u>: Let the total end-to-end delay equal

$$D_T = \sum_{i=1}^{i=M} \tau_i^q + \sum_{i=1}^{i=M-1} \tau_i^p + \tau_{pd}$$

where

τ_i^q is the delay term caused by the queueing at each node spanned by the connection;

τ_i^p is the propagation delay at each link i along the connection path;

τ_{pd} is all the other fixed delay terms, including the packetisation and depacketisation delays.

Using assumptions i, ii and iv $E\{y\} = E\{y^{ref}\}$, therefore the long term queueing delay is upper bounded (with a probability of $1 - E_{overflow}$) to

$\lim_{t\to\infty}(\tau_i^q) = \frac{y^{ref}}{\text{[illegible]}}$ (where C^{link} is in units of cellslsec)

$$\therefore \quad \lim_{t\to\infty}(D_T) = \sum_{i=1}^{i=M} \frac{y_i^{ref}}{C_i^{link}} + \sum_{i=1}^{i=M-1} \tau_i^p + \tau_{pd} = \tau_{total}, \text{ i.e. a constant.}$$

Note that this is an upper bound on the expected value of the end-to-end delay (with a probability equal to $1 - E_{overflow}$). VVV

<u>ii) the long term losses are bounded</u>, at least with a probability of $1 - E_{overflow}$ (assuming that the buffer size is greater than y^{ref}).

<u>Proof:</u> Using assumptions i, ii and iv, $E\{y\} = E\{y^{ref}\}$ therefore the buffer occupancy is upper bounded to y^{ref} (at least with a probability of $1 - E_{overflow}$).

VVV

<u>iii) the long term CDV is bounded</u>. Again using similar reasoning as in i) we can show that the expected value of the CDV is upper bounded by τ_{total} with a probability of $1 - E_{overflow}$.

VVV

Appendix 3.VI: Simulation cases A-F

case A) Simulation run for 744000 celltimes (sum of the peak bit rate of the uncontrollable sources is less than the link rate)

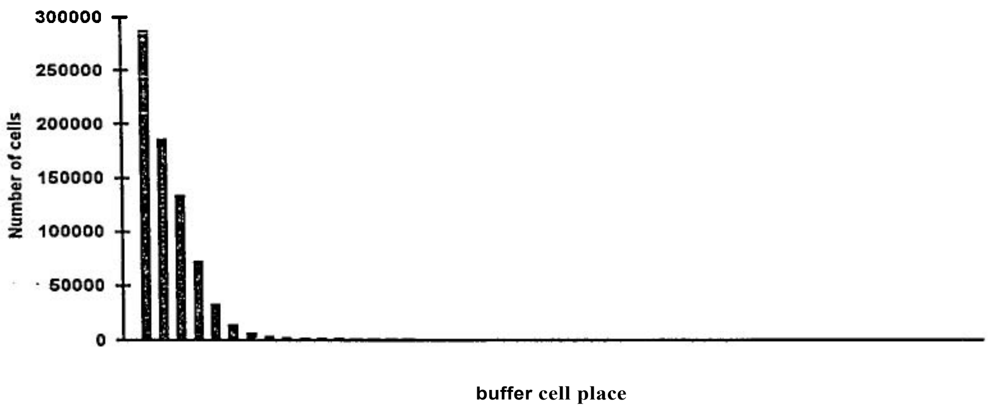
In this case we use Connection mix 1 to ensure that the link rate is never exceeded. This represents CAC algorithm 1. For this simulation run of 3/4 of a million celltimes no loss occurred, the delay for the uncontrollable traffic remained bounded below 83 µsec (for a link rate of 155 Mbit/sec), yet the throughput is a high 0.89 (compare to the throughput, of 0.60, obtained by setting the controllable source output to zero). Note that (for this setting of y^{ref} and $E_{overflow}$) a 48 % increase in the throughput rate has been achieved, over a scheme that admits on just the peak rate.

A summary of the results is shown in the table:

Simulation run length (celltimes)	reference on cell places to hold 99.999% of the cells	actual buffer places used to hold 99.999% of the cells	Throughput	Maximum instantaneous buffer places occupied	Actual cell-loss over the length of the simulation run
744000	25	30	0.89	31	0 out of 744000 cells

Observe that the actual cell-buffer places used to accommodate 99.999% of the cells passing through is equal to 30 in this run time; which can be compared to the reference value of 25. A very tight control is achieved (and hence delays and losses are bounded). Note that the cell-loss, and the cell-delay and cell-delay-variation for the uncontrollable sources, can be kept (assuming strict controllability) at any desired value, by appropriate choice of y^{ref} and $E_{overflow}$.

The histogram and table below show the buffer occupancy up to 32 cell places. Figure 3.14 shows two typical segments of the simulation run.



Buffer cell place	0	1	2	3	4	5	6	7	8
Occupancy of cell place (in number of cells)	286725	185436	133415	71839	32379	13708	5846	3044	1971
% Occupancy of cell place	38.5	24.9	17.9	10	4	2	0.8	0.4	0.26

9	10	11	12	13	14	15	16	17	18	19	20
1467	1369	1241	1061	838	726	599	455	434	328	237	207
0.2	0.18	0.16	0.14	0.11	0.09	0.08	0.06	0.05	0.04	0.03	0.03

21	22	23	24	25	26	27	28	29	30	31	32
172	140	119	70	57	37	30	21	15	10	4	0
0.02	0.02	0.02	0.01	0.008	0.005	0.004	0.003	0.002	0.001	0.0005	0

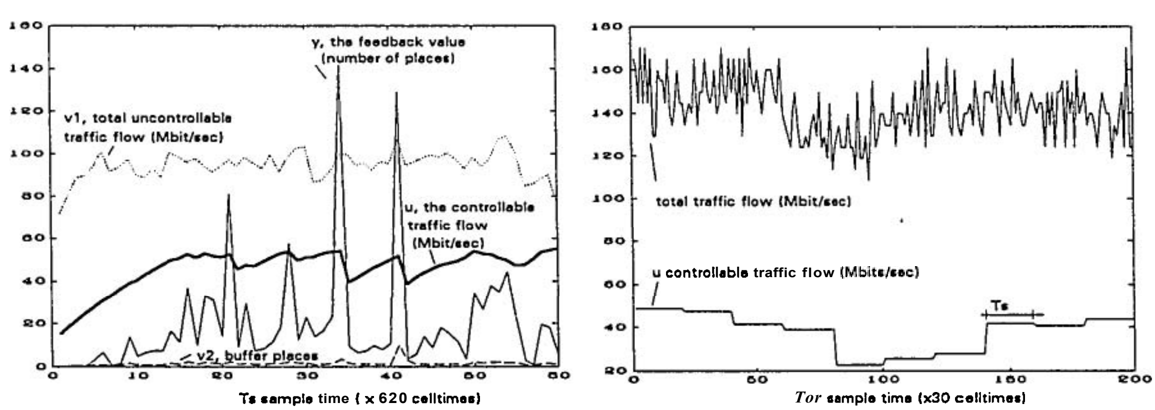


Figure 3.14. Typical segments of the simulation run: 37200 and 6000 celltimes.

Case B) Simulation runs for a video connection and disconnection:

Run 1) video 1 remains connected during the whole simulation run of 37200 celltimes.

Run 2) video 1 is disconnected after 20000 celltimes (it could, for example, be interactive video).

Run 3) video 1 is connected after 20000 celltimes.

Note that the adaptive controller settings are: $\lambda_o = 15$; dominant pole $\mathbf{p} = 0.5$; the reference is set at $y^{ref} = 25$ cell places and $E_{overflow} = 10^{-9}$. The control loop regulates the controllable traffic such that a buffer of 25 cell-places is expected to accommodate $(1 - 10^{-9}) \times 100\%$ of the total cells served.

Observe that no noticeable degradation has occurred in any of the cases considered. The throughput remained constant at about 0.83 for all cases even though there was a video (of 25 Mbit/sec) connection and disconnection—that is the controller has increased or decreased its output to suit the changed circumstances. The maximum instantaneous buffer occupancy remained at a low 10 cell places for all three cases.

A summary of the results is shown in the table:

Run number	Throughput	Maximum Instantaneous buffer places occupied	Maximum feedback value	Maximum calculated cell-loss probability	Actual cell-loss over the length of the simulation run
Run 1	0.837	10	45	10^{-20}	0
Run 2	0.828	10	45	10^{-20}	0
Run 3	0.828	10	52	10^{-20}	0

<u>RUN 1: video 1 remains connected throughout the simulation run.</u>

A throughput of 0.837 is achieved for this simulation run.

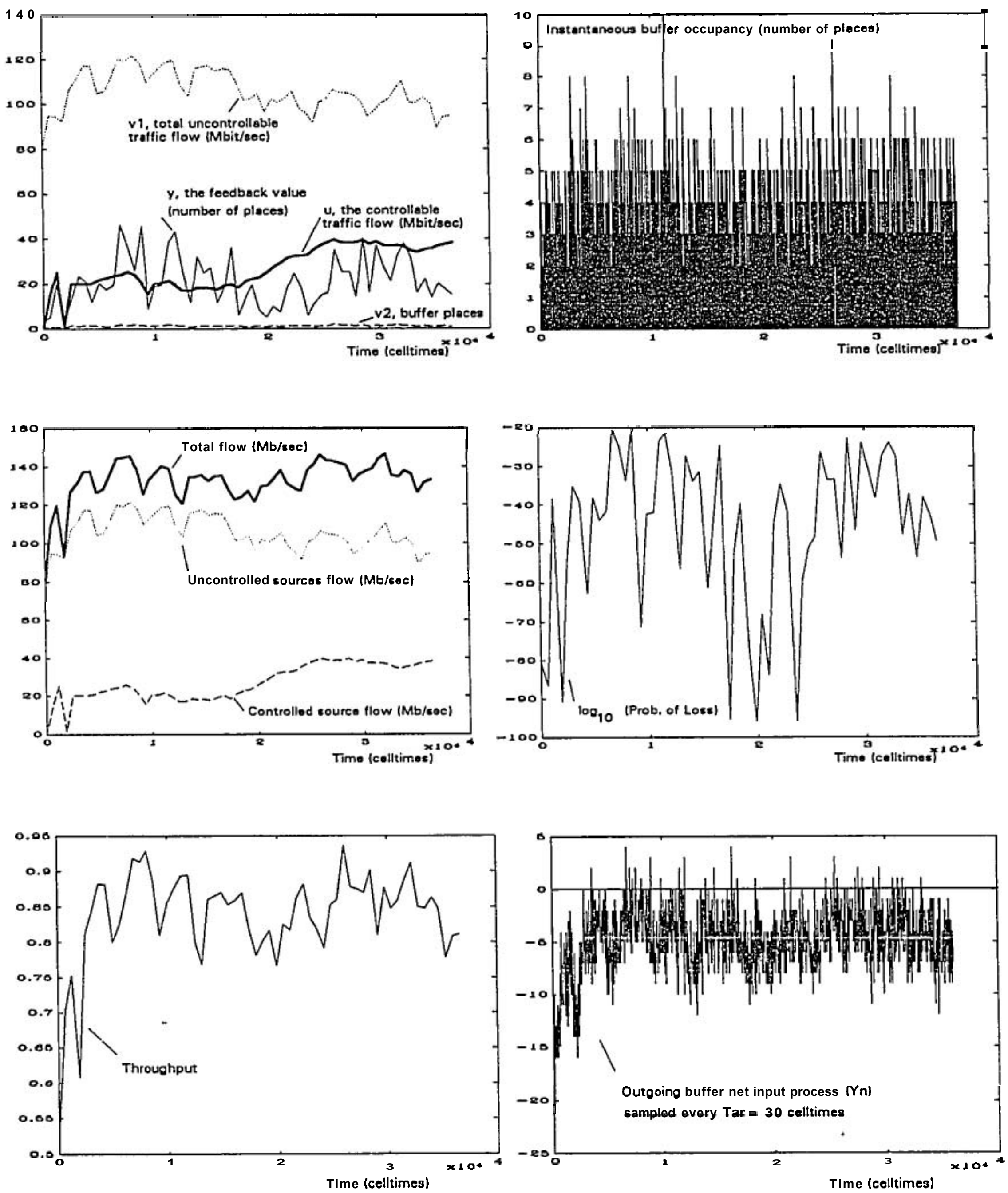


Figure 3.15. Video connection and disconnection simulations: <u>video remains connected</u> throughout simulation run.

RUN 2: Video 1 is disconnected after 20000 celltimes have elapsed.

Notice that there is no visible degradation, and that the controllable load is adaptively controlled to keep the output close to its prescribed value. A throughput of 0.828 is achieved, as in the previous case, without any degradation in performance.

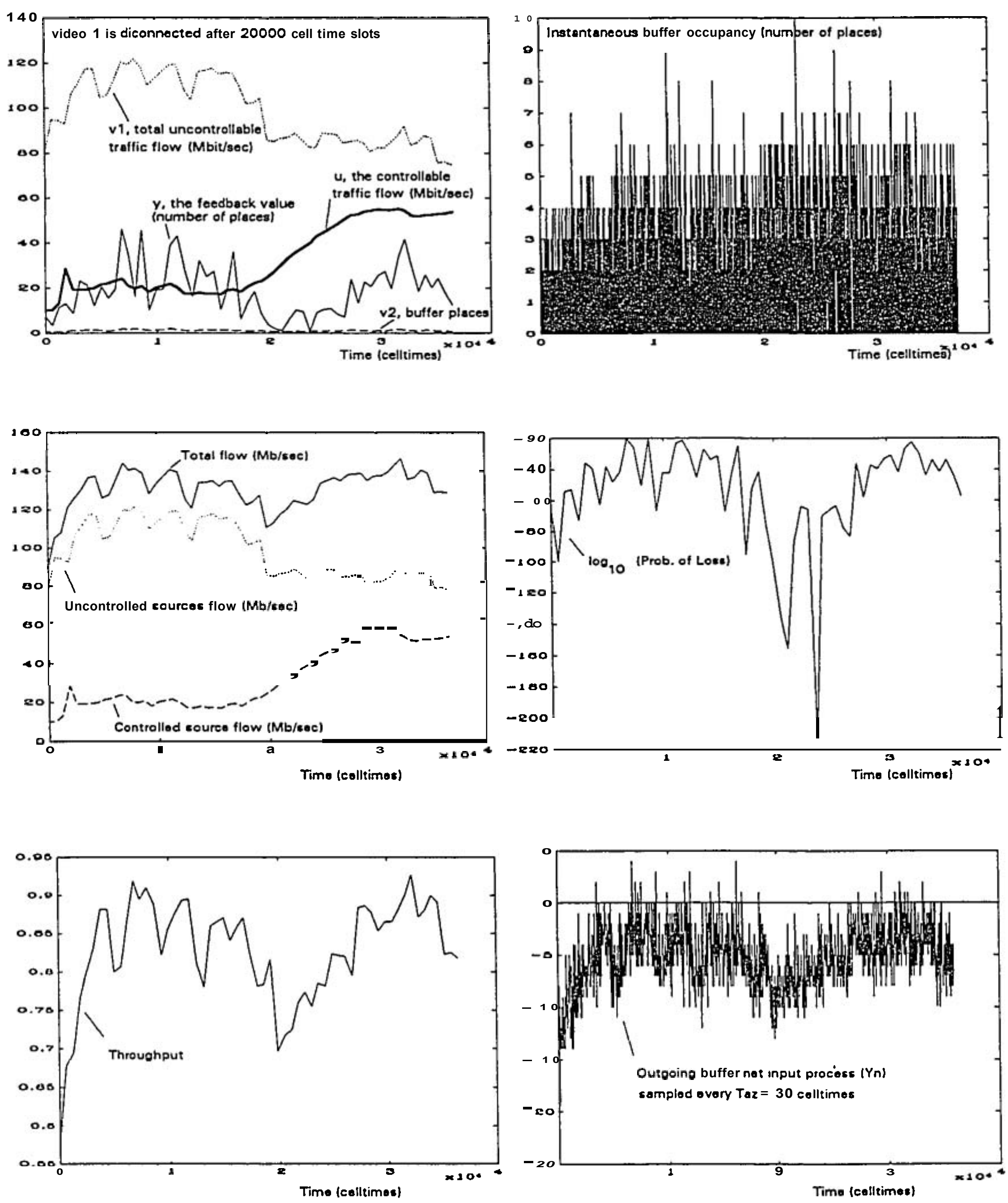


Figure 3.16. Video connection and disconnection simulations: video disconnected after 20000 celltimes elapsed.

RUN 3: Video 1 is connected after 20000 celltimes.

Notice that there is no visible degradation, and that the controllable load is adaptively controlled to keep the output close to its prescribed value. A throughput of 0.827 is achieved, as in the previous case, without any degradation in performance.

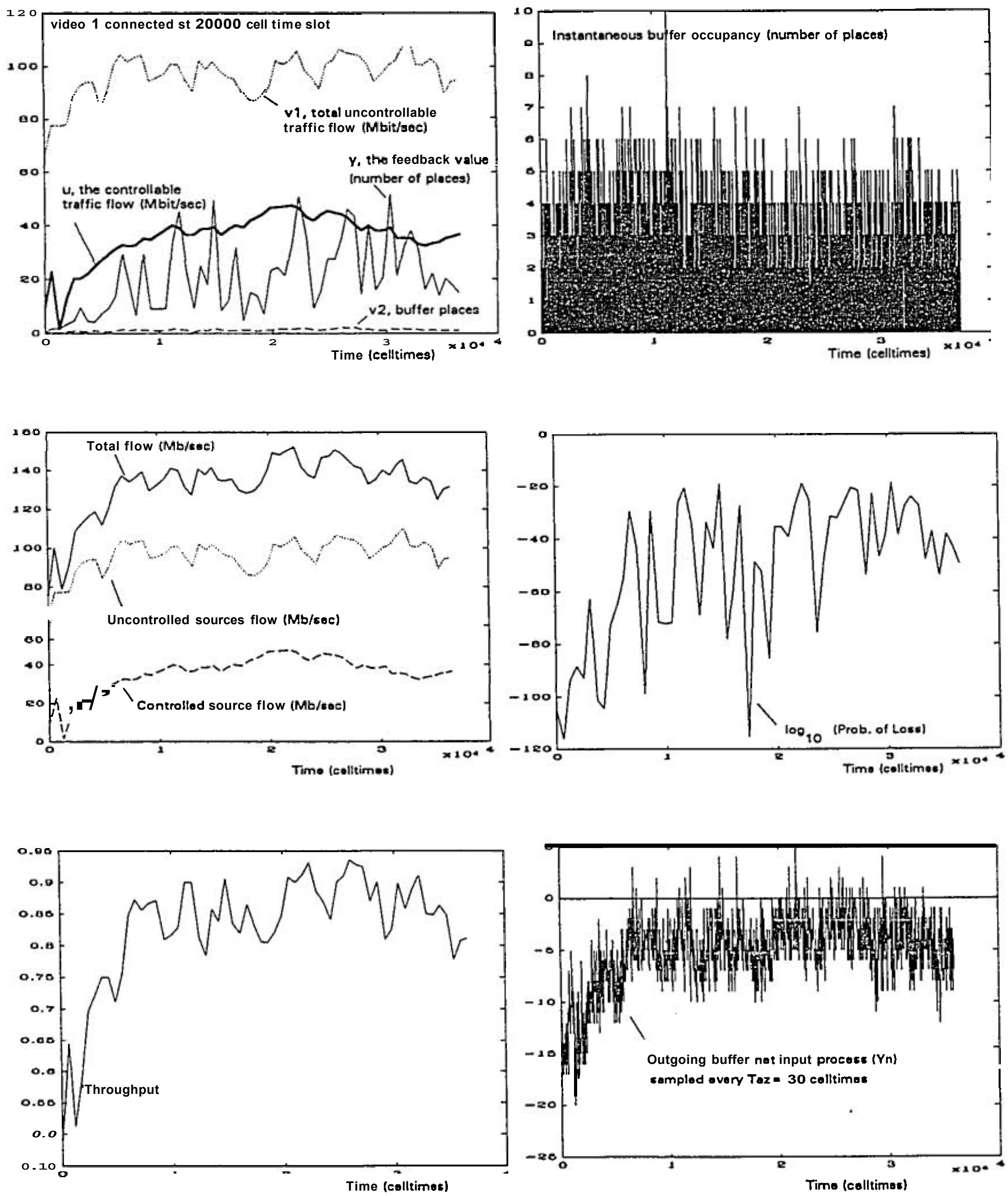
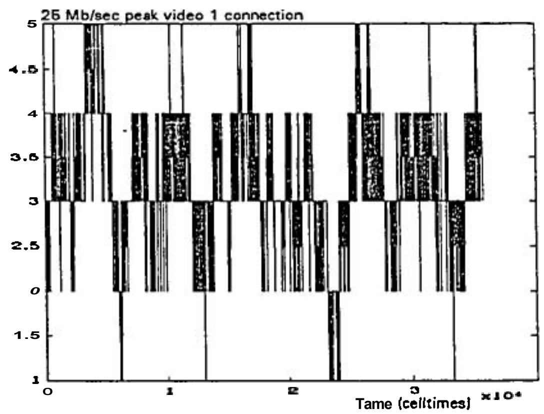
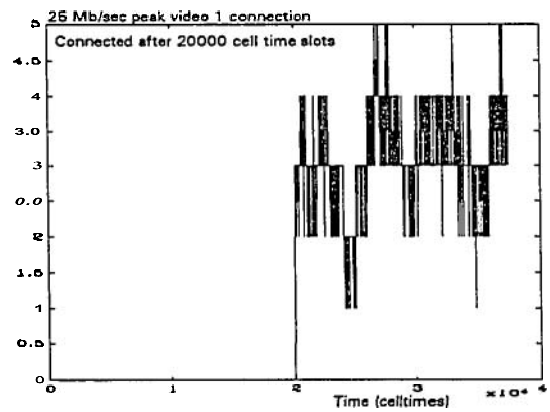


Figure 3.17. Video connection and disconnection simulations: video connected after 20000 celltimes elapsed.

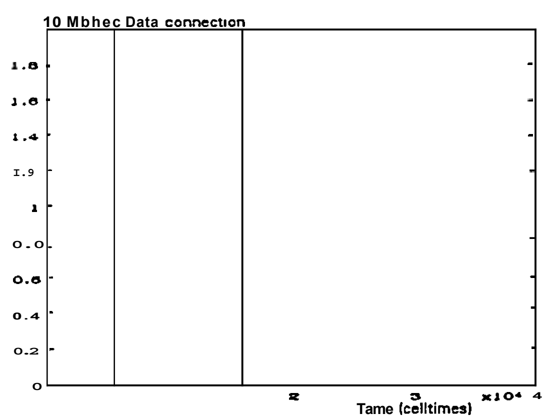
Examples of video and data connections (sampled every T_{az})



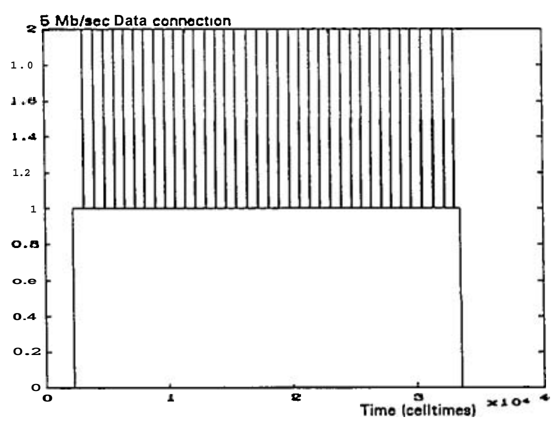
a) Video 1 connection cell rate.



b) Video 1 step connection cell rate.



c) 10 Mbit/sec data connection cell rate.



d) 5 Mbit/sec data connection cell rate.

Figure 3.18. Examples of video and data connections; in number of assigned cells during each sampling interval T_{az} (T_{az} set to 30 celltimes) plotted versus celltimes.

<u>Case **C)** Simulation runs for three **different** reference settings:</u>

Run 1) Reference set to $y^{ref} = 20$ cell-buffer-places and $E_{overflow} = 10^{-9}$.

Run *2)* Reference set to $y^{ref} = 50$ cell-buffer-places and $E_{overflow} = 10^{-9}$.

Run 3) Reference set to $y^{ref} = 75$ cell-buffer-places and $E_{overflow} = 10^{-9}$.

Note that the control loop regulates the controllable traffic such that a buffer of y^{ref} cell-places is expected to accommodate $(1 - 10^{-9}) \times 100\%$ of the total cells served.

This set of simulation runs demonstrates that the utilisation of the system increases as the reference value is increased (for the same pseudorandom uncontrollable traffic sequence). Also the ability (possibly from a higher level) to influence the behaviour of the system, by changing the reference value, is demonstrated. **As** the reference value increases the throughput increases, however with an increase in the maximum instantaneous buffer occupancy. Note that the value of $E_{overflow}$ can also be used to influence local behaviour

A summary of the results is shown in the table:

Run	Reference value y^{ref}	Throughput	Maximum instantaneous buffer places occupied	Actual cell-loss over the length of the simulation run
Run 1	20	0.85	8	0
Run2	50	0.89	10	0
Run3	75	0.91	34	0

Note that the adaptive controller settings are: $\lambda_o = 15$; and dominant pole $p = 0.5$.

RUN 1: Reference = 20

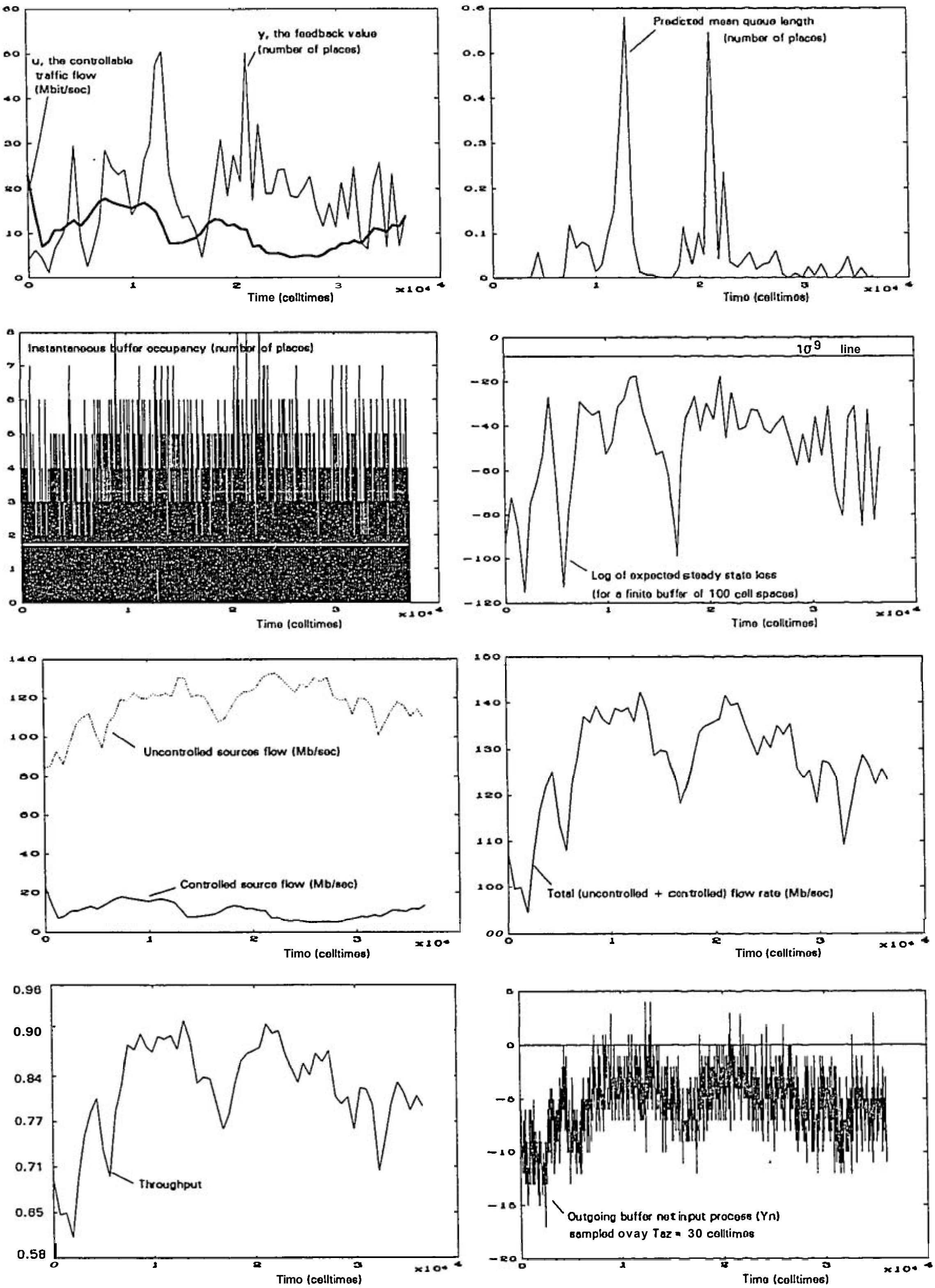


Figure 3.19. Reference set to $y^{ref} = 20$ buffer cell places to accommodate $(1-10^{-9})$ x 100% of the total cells served.

RUN 2: Reference = 50

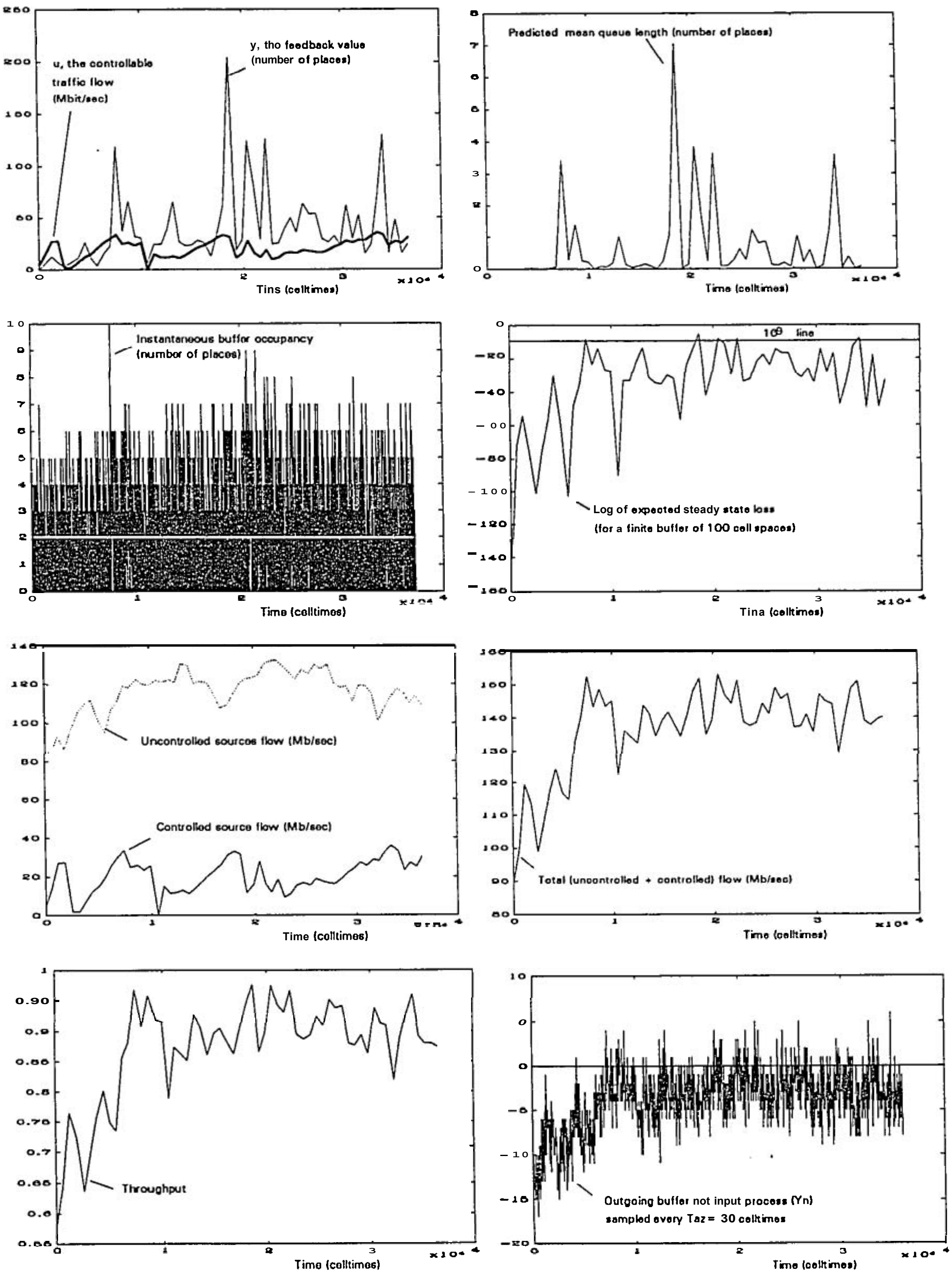


Figure 3.20. Reference set to $y^{ref} = 50$ buffer cell places to accommodate $(1-10^{-9}) \times 100\%$ of the total cells served.

RUN 3: Reference = 75

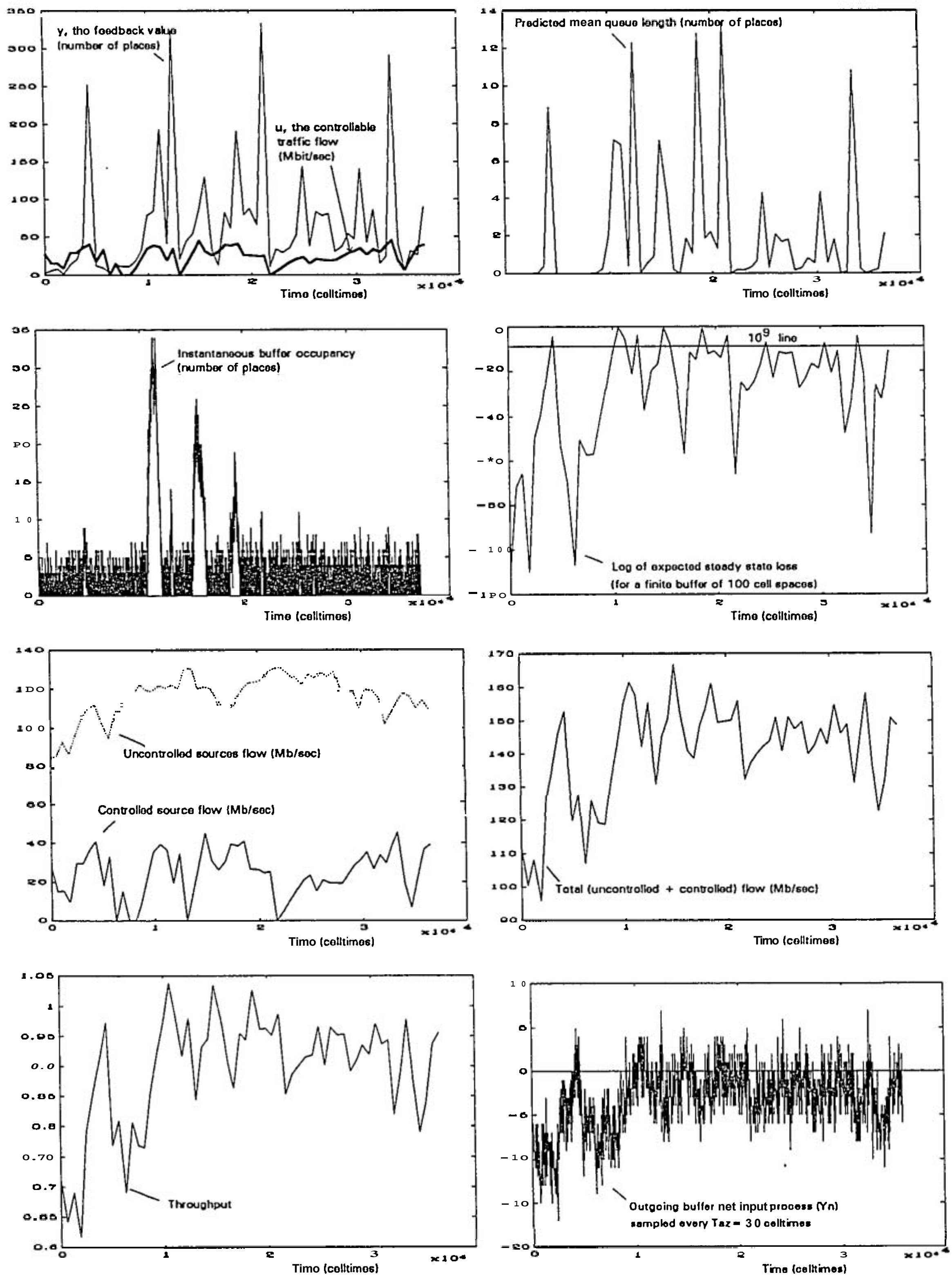


Figure 3.21. Reference set to $y^{ref} = 75$ buffer cell places to accommodate $(1 - 10^{-9}) \times 100\%$ of the total cells served.

Case D) Simulation runs for different control penalty weights:

Run 1) the control penalty $\lambda_o = 0.05$.

Run 2) the control penalty $\lambda_o = 15$.

Run 3) the control penalty $\lambda_o = 150$.

Note that the adaptive controller settings are: the dominant pole $p = 0.5$; and the reference is kept at $y^{ref} = 25$ cell places and $E_{overflow} = 10^{-9}$. The control loop regulates the controllable traffic such that a buffer of y^{ref} cell-places is expected to accommodate $(1-10^{-9})\times 100\%$ of the total cells served.

Observe the tradeoff between output and control variance. In the first run, the controller gives excessive weight to its goal to minimise the variance at the output and it does not prevent the loss (for the finite buffer of 100 cell places). As the control penalty increases in Run 2 and Run 3 we obtain a smoother control response and no losses are experienced.

A summary of the results is shown in the table:

	control penalty λ_o	Throughput	Maximum Instantaneous buffer places occupied	Maximum calculated cell-loss probability	Actual cell-loss over the length of the simulation run
Run 1	0.05	0.79	100	1	**361**
Run 2	15	0.835	8	10^{-18}	0
Run3	150	0.84	8	10^{-17}	0

RUN 1) the control penalty $A, = 0.05$

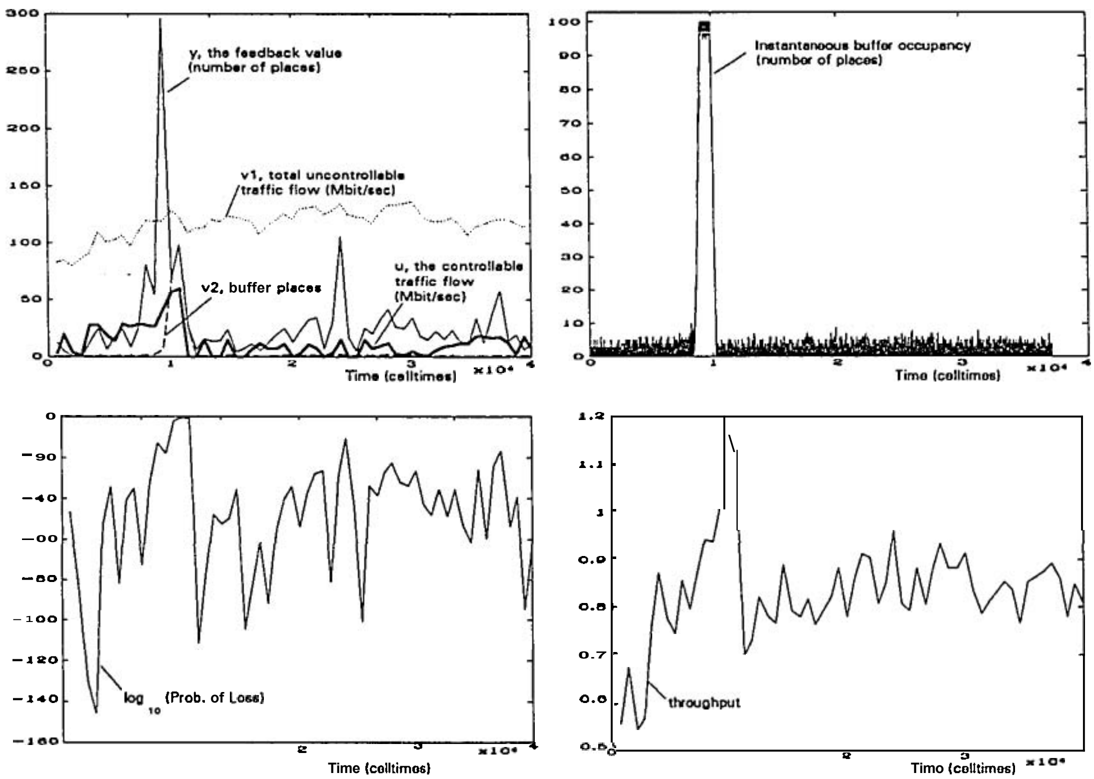


Figure **3.23.** Simulation run for control penaltv weight $A, = 0.05.$

RUN 2) the control penaltv $A, = 15$

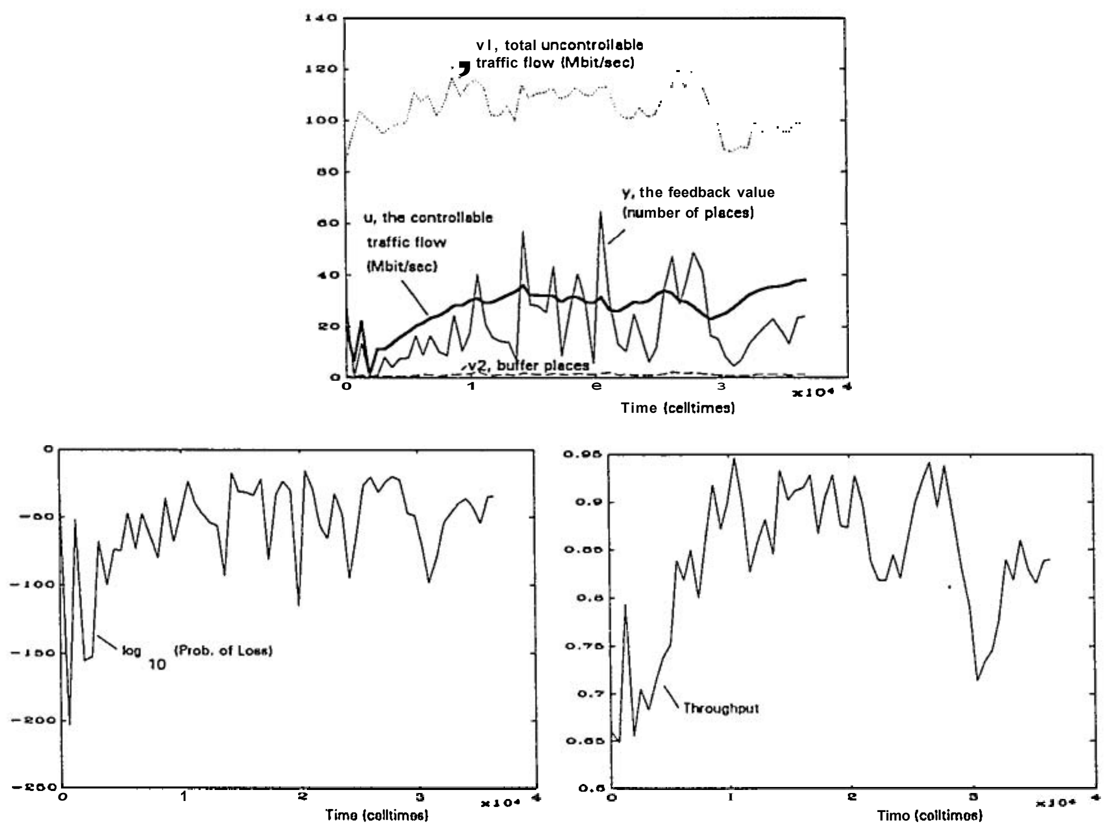


Figure **3.22.** Simulation run for control penalty weight $A, = 15.$

RUN 3) the control penalty $\lambda_o = 150$

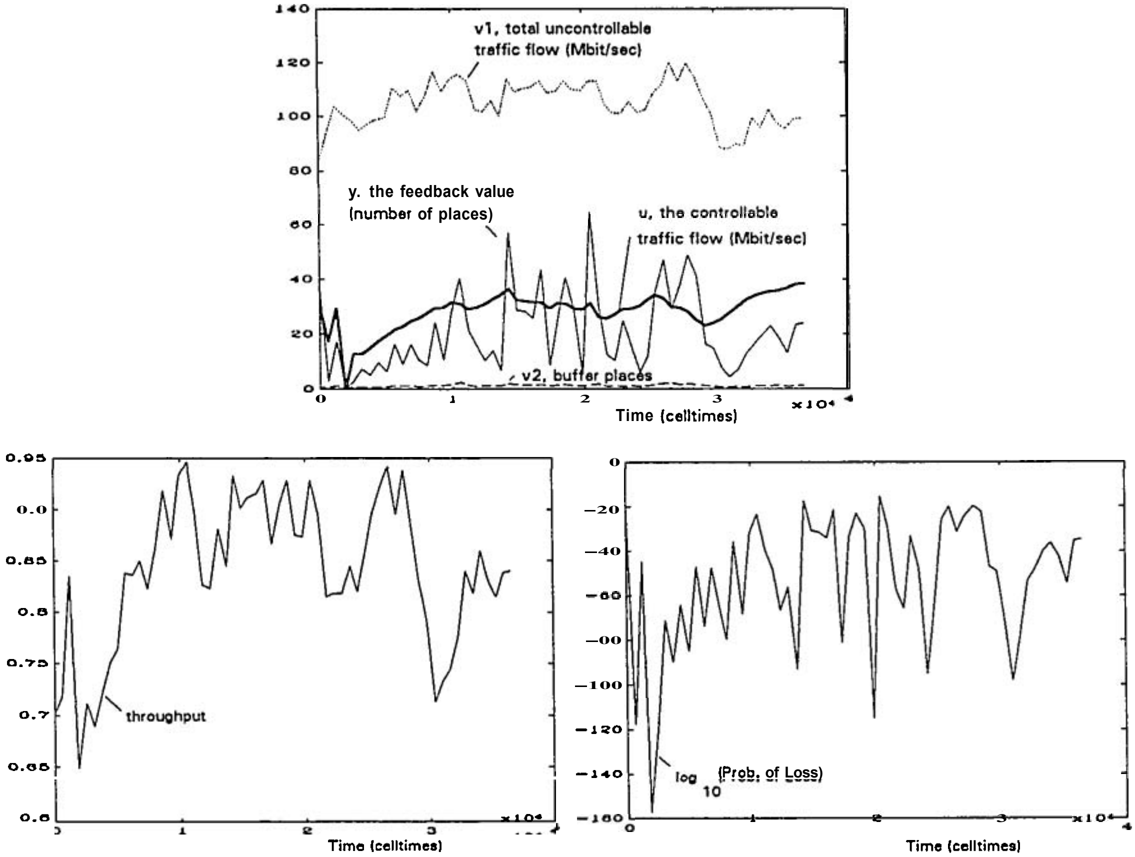


Figure 3.24. Simulation run for control penalty weight $\lambda_o = 150$

Note that a typical buffer histogram is as shown below (for either $\lambda_o = 15$ or $\lambda_o = 150$).

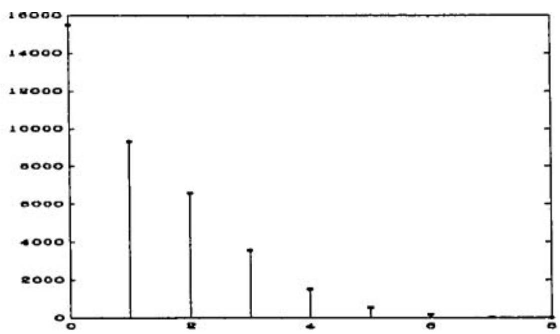


Figure 3.25. Typical buffer histogram.

Buffer place	0	1	2	3	4	5	6	7	8
Number of cells	15484	9316	6563	3559	1529	555	172	21	1

Case E) Simulation runs with and without feedforward compensation

Two cases were considered:

Run 1) with feedforward compensation.

Run 2) without feedforward compensation.

The adaptive controller settings are: number of control parameters $n_\alpha = 1$, $n_\beta = 2$, $n_{\gamma_1} = 2$ (run 1 only) and $n_{\gamma_2} = 1$ (run 1 only); the controller initial conditions are zero with the exception of $\beta_o = 1$, which represent a control system starting up as a simple integrator (common approach in the case of no prior knowledge about the system dynamics), and $\gamma_o^1 = 1$ (run 1 only); dominant pole $p = 0.5$; control penalty $\lambda_o = 15$; reference $y^{ref} = 25$ cell places; and $E_{overflow} = 10^{-9}$. The control loop regulates the controllable traffic such that a buffer of y^{ref} cell-places is expected to accommodate $(1-10^{-9})$ x 100% of the total cells served.

The identical pseudorandom uncontrollable traffic sequences were used in both runs. The results from the two cases are summarised in the table below and in figure 3.26 and 3.27. The feedback signal, the control effort, the probability of loss and the throughput can be seen in figure 3.26. Figure 3.27 compares the evolution of the estimated control parameters over the length of the simulation run. As can be seen from the table and the figures the case with feedforward compensation clearly outperforms the case without.

	cost functional $\times 10^5$ (equation *3.6*, page 40)	throughput
Run 1 (with feedfonvard compensation)	0.87	0.8527
Run 2 (without feedforward compensation)	2.46	0.8378

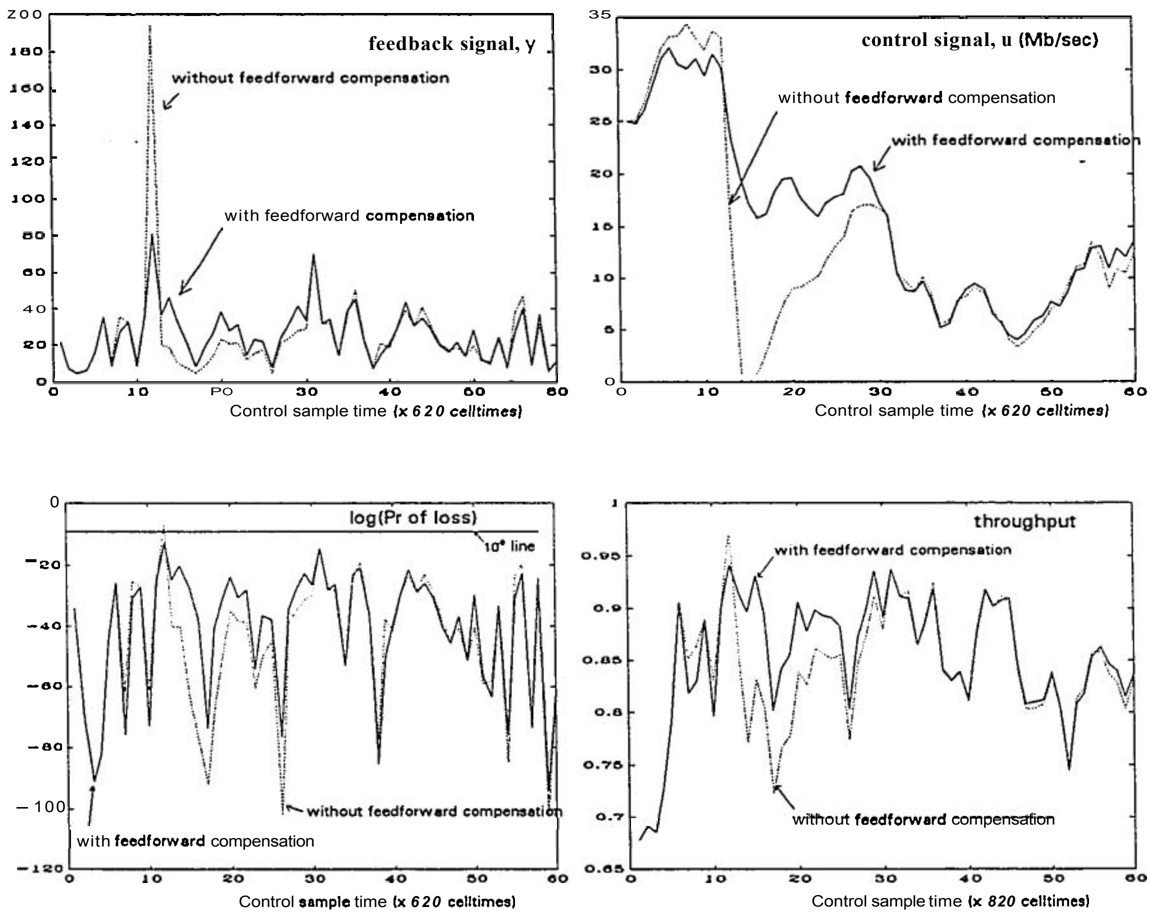


Figure *3.26.* Comparison of the controlled system performance with and without feedforward compensation.

Note that our simulative experience shows that the degree of improvement (by the case with feedforward over the case without) depends on the initial controller parameter set. Also observe that the time axis is labelled in terms of the control sample time.

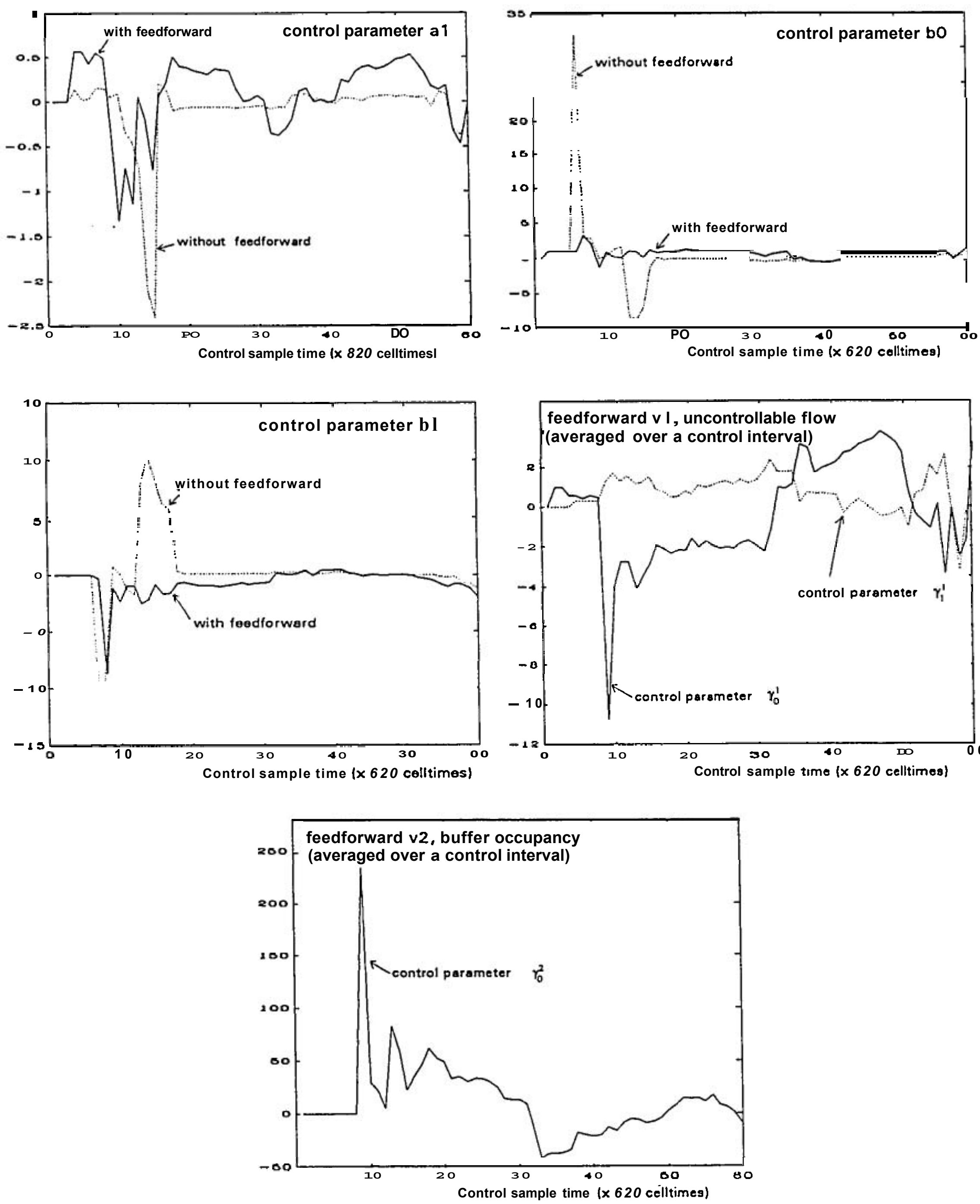


Figure **3.27.** Evolution of the estimated control parameters over the length of the simulation run.

Case F) Simulation runs for fixed controller output settings (i.e. open loop control):

The constant transmission-rate for the controlled source was set as follows:

Run 1) 15 Mbit/sec.
Run 2) 25 Mbit/sec.
Run 3) 35 Mbit/sec.
Run 4) 45 Mbit/sec.

The identical pseudorandom uncontrollable traffic sequence is used for each run, with a mean of approximately 100 Mbit/sec. As can be observed, at some critical rate there is a huge degradation in performance as the rate of the controlled source is increased. (Note that this highlights the sensitivity of buffer occupancy to uncertainty in traffic modelling). If an open loop CAC policy was employed and it predicted that a source with the appropriate statistics (of say an average of 15 and a peak of 35 Mbit/sec) is acceptable to connect to the network, then the network can experience periods of heavy losses, without the ability to minimise the loss experienced.

transmission-rate of the controlled source (Mbit/sec)	15	25	35	45
losses (cells in 37200 celltimes)	0	47	1008	2338

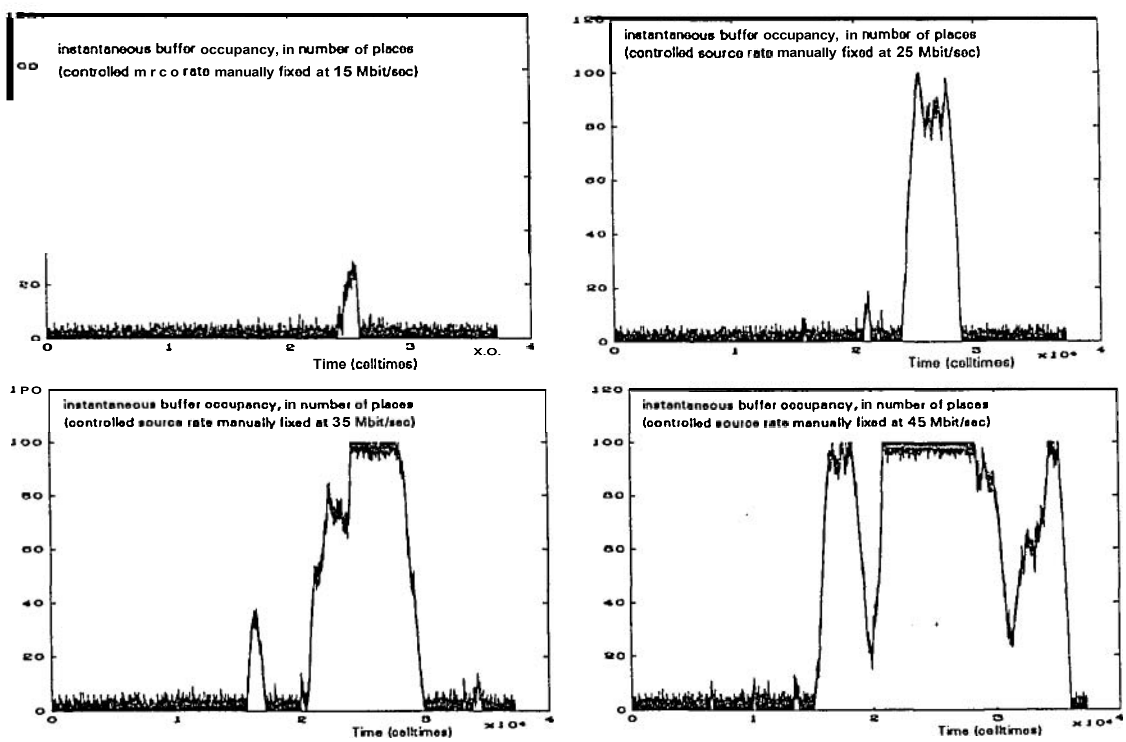


Figure 3.28. Sensitivity of buffer occupancy to changes in the incoming traffic.

CHAPTER 4

A NOVEL DYNAMIC SERVICE-RATE CONTROL SCHEME FOR B-ISDN

4.0 Introduction

Service-rate (bandwidth, capacity) control[#1] is often employed for the prevention of network congestion and the subsequent network throughput degradation, by allocating enough resources to communicating entities. A variety of techniques for service-rate control have been proposed in the literature and these will be reviewed in section 4.1. The majority of these schemes are open-loop, static, single level, and as such they cannot deal with non-stationary network behaviour and network wide objectives in an efficient manner (essential attributes for an effective BISDN control system; see discussion in Chapter 2).

After motivating dynamic service-rate control in section 4.2, a novel scheme is proposed in section 4.3, in which the state of the buffers in the network is used as a feedback signal to dynamically allocate the server service-rate. The proposed feedback scheme is applicable to general queueing systems and it is particularly useful when network capacity is a scarce resource.

Two illustrative examples (using multilevel and single level approaches) for solving the dynamic service-rate control problem at the VP level are presented in section 4.5. In particular, we make use of the VP concept (modelled in terms of a dynamic fluid flow type equation; presented in a unified form in section 4.4) to formulate precise problems for the control of service-rate at the VP level. The interactions within the nodes spanned by a VP, as well as the interactions between the VPs sharing a link are addressed in the problem formulation. The form of the solution is suitable for incorporation in an overall hierarchically organised control structure (an essential form, as discussed in Chapter 2). Some open questions, with regard to their implementation are discussed in section 4.5.1.5 (for the multilevel implementation) and in section 4.5.2 (for the single level implementation).

The performance of the derived algorithms is illustrated via simulation in section 4.6.

#1 Management and control; we make no distinction between the two terms and use control for either. (see discussion in the footnote of page 25).

4.1 Review of some related service-rate control schemes and nonlinear multilevel control

Existing literature on service-rate control and non-linear multilevel optimal control are reviewed in the next two sections. The section on service-rate control highlights some of their limitations. The section on nonlinear multilevel optimal control reviews a methodology which can be productively applied to dynamic network control problems.

4.1.1 Service-rate control schemes

Service-rate control in message and packet-switched networks has possibly been introduced as early as 1964 by Kleinrock in [145]. Since then numerous researchers have published results in this field. The control of service-rate can be addressed at different levels. The majority of the literature focuses on the control of service-rate at a single level, in an open loop fashion. It does not explicitly recognise the fact that, due to the complexity of the problem (see discussion in chapter 2), more that one level is necessary for its effective control. The interaction between and within levels and integration into an overall solution is hardly discussed. Additionally, the dynamic behaviour of the system is rarely taken into consideration. We review some of these schemes, organised in the various levels to which they apply:

- static single level schemes have been proposed for the following levels: network level (e.g. Herzberg [146]; Evans [147]; Gerla et al [148]; Harris et al [149]); call level (e.g. Gallassi et al [61],Guerin et al [63], Joos et al [58], Dziong et al [150] and Kelly and coworkers [151]; also see report [52]); Burst level (e.g. Boyer et al [152], and as part of a multilevel approach by Hui [24] and Filipiak [25]); Cell level schemes (e.g. Hyman et al [153], Takagi et al [154], Dighe et al [155]).

  These schemes cannot handle nonstationary, or transient behaviour (due for example to network malfunction, natural disasters, or simply changing demand patterns). They can possibly form a part of a hierarchically organised overall service-rate control scheme.

- static multilevel approaches: These schemes are appealing, since they are organised in a hierarchical structure. However they are mainly static and narrowly focused on one aspect of network control (for example: Bolla et al [43] only consider the problem of service-rate allocation at one single node viewed over two layers; Hui [24], [156], Hui et al [157] consider a 3 layer vertical decomposition; Ren et al [44] consider only two layers; and Filipiak [25] describes a general multi-strata framework for flow control based on system tables, updated by a higher level whose design is based on heuristics).

- dynamic single level schemes: Ohta, Sato and colleagues [158], [159], [50] have provided some basic analytic results on the control of VPs, based on simple heuristic service-rate control schemes.
- dynamic multilevel schemes: These schemes have only been considered by a few researchers. Some notable examples follow. John Burgin [160], one of the originators of the VP concept, proposed a two level scheme (note that this is a limiting control architecture; more horizontal and vertical levels may be necessary). At the top level he proposed a dynamic service-rate control scheme in which the service-rate allocations are updated centrally, based on a traffic loss function calculated from the estimated offered traffic (estimated from the peak rate of the measured carried traffic and the peak rate of the declared blocked traffic) and the actual service-rate allocations. At the lower level he proposed CAC control, featuring multiple control units, located at the admitting nodes (no details of CAC were given, but presumably they are based on admitting on the declared peak cell-rate, and if so they would be very inefficient). Pitsillides et al [37] presented a general hierarchical structure for the control of BISDN. One level, that of the dynamic service-rate control scheme (allocates service-rate to VPs based on the state–described by a fluid flow equation– of the network queues spanned by the VP), was discussed in detail. Herzberg and Pitsillides [161] presented a hierarchically organised four level scheme for service-rate control (three levels of the integrated structure were discussed in detail).

Based on the above review of the control of service-rate we can summarise their shortcomings as follows: the vast majority of works concentrate on the control of service-rate at a particular single level; few works take into account the fact that other levels may be necessary, and even fewer offer a solution based on more than one level; the dynamic aspects of the different levels are largely ignored.

4.1.2 Nonlinear multilevel optimal control

The classical calculus of variations and optimal control theory of Bolza-McShane-Pontryagin and others has been used on feedback control problems with considerable success. The general, nonlinear, optimal control problem was formulated by Pontryagin and colleagues (generally known as the Pontryagin maximum principle) around the later part of the 1950s. They collected their works in a book [162]. it is worth noting that around the 1960's Linear Quadratic (LQ) theory provided a powerful tool for multivariable feedback synthesis with quadratic optimality.

Optimal control theory has been used by a large number of researchers to solve telecommunication problems. One of the contributing factors is that it provides the "best" possible solution (in accordance with the chosen objective function). It also allows the formulation of the control objectives, even under nonstationary conditions, in a precise and easy fashion (see Tipper's Ph.D. thesis [72], and [18]). Some notable examples of the use of optimal control theory to solve communication network problems are: Segall [163], the first to propose a state model approach; Moss and Segall [164]; Filipiak [165], [166]; Tipper and Sundareshan [167], [168]; Economides, Ioannou and Sylvester [169], [170]; Stassinopoulos and Kostantopoulos [171]; Iftar and Davison [172]; Sarachik and Ozguner [173]; and Cassalino Davoli, Minciardi and Zoppoli [174].

Note that the vast majority of these schemes concentrate on the routing and flow control problem. None of the above works deals with the service-rate allocation control problem.

The works of Tipper [72], Tipper and Sundareshan [18], discuss the use of dynamic modelling techniques, as well as the use of optimal control methodology for communication networks, and as such have been widely drawn upon in our works. However the adoption of nonlinear multilevel optimal control methodology based on decomposition and coordination for the solution of large scale nonlinear systems we believe, is also novel. This allows the solution to be implemented in a decentralised coordinated form. A multilevel control theoretic approach has been adopted by relatively few researchers, some notable examples are: Garcia and Hennet [175] for a two level structure solving a static optimisation problem at the higher level and a linear feedback control law, based on a learning automaton, at the lower level; Muralidhar and Sundareshan who have presented two level schemes for routing and flow control, formulating the lower level as a minimum hop routing problem in [42], and a linear state model for the network in [176]; and Tipper [72] who uses two levels for the optimal buffer management problem, formulating the lower level problem using a queueing model, and at the higher level uses heuristics to provide the coordination inputs to the local units in order to achieve network wide performance objectives, after showing that it is difficult to formulate a precise optimisation problem.

Nonlinear optimal control of large scale systems has not evolved to the extent of the linear control case due to the lack of a unifying design theory. However in the late seventies a concerted research effort was devoted to the study of hierarchical techniques for the optimal control of nonlinear systems by M. G. Singh, M. F. Hassan, W. Findeisen, K. Malinowski, A. Titli, M. S. Mahrnoud, A. Sage, C. Tzafestas and M. Jamshidi, to name but a few.

In appendix 4.1 we provide a brief discussion of some of the works on the optimal control of large scale nonlinear systems that is closely related to the hierarchical successive approximation approach. In particular, we describe one technique–the costate coordination technique–which provides solutions that can be implemented in a decentralised coordinated form and which, as reported, is computationally appealing since it avoids the solution of a TPBV problem at the lower levels.

4.2 Motivation of the service-rate control approach

Some limitations of the existing service-rate control schemes, as seen by us, based on the above discussion, include:

- no attempt is made to integrate the various service-rate control schemes proposed for different time scales (due to the complexity of the problem, decomposition into a number of levels is necessary; see discussion in chapter 2). Generally a particular scheme is designed in isolation without any attempt to integrate the higher and lower levels. Thus the interactions between and within the levels cannot be taken into account, resulting in network inefficiency with a possible network instability (manifested in the form of congestion) caused by large perturbations, from the equilibrium values.
- no attempt is made to use feedback from the state of the network over the short to medium term (some schemes exist that use feedback from the state of the network at the cell-time scale to mark and/or discard cells [41]; we see this as a necessary last resort). Single level implementations, based on long term averages of the traffic demand and/or open loop approaches (based on user declared parameters) are usually considered. The shortcomings of the approach taken by these schemes are that they are not able to adapt to fast changing loads and that, by not taking into account the network state, out of control situations can arise (maybe leading to instability), due to the network condition changing from its nominal state. Note that on higher levels of a hierarchically organised solution,

open loop approaches may be tolerated, since the lower levels will respond to changes from the assumed model.

- static or quasi static models, mainly derived from probabilistic models, have been used for the vast majority of existing service-rate control strategies. A major shortcoming of this simplifying assumption (that of stationary loads and steady state conditions) is that even though a precise policy for service-rate allocation may be obtained for the specific steady state conditions, the performance of these schemes can be far from the optimum (with the possibility of instability for some schemes) when the steady state conditions are not present, as for example when the traffic varies from its nominal values.

A dynamic hierarchically organised service-rate control scheme is expected to provide many enhancements of network performance (see Chapter 2). Among these are:

- Adaptability to unexpected traffic variations and network failures, thus increasing the network reliability.
- Increased network throughput, due to increased link capacity utilisation.
- Reduced losses.

In the next section we provide the formulation of the service-rate control problem at one level (that of the VP). We designate this scheme the VP Control, or VPC for short. For illustrative purposes we derive two schemes that can be incorporated in a broader overall service-rate control scheme. An example of a hierarchically organised overall service-rate control scheme, in which VPC is integrated, is presented in Chapter 5.

4.3 The dynamic service-rate allocation concept

Our novel concept is shown in figure 4.1. It is aimed at allocating service-rate economically while maintaining low loss and delay. It uses local (fast acting) feedback from the network queues to dynamically control the server service-rate. Therefore the allocated server service-rate is dependant on the local network state. Note that the state of the queues is directly related to queueing delays, but more importantly (for ATM) the buffer overflow probability is related to the average queue length. Hence keeping the queue lengths controlled achieves the twofold objective of minimising the buffer overflow probabilities (and hence cell-losses) and the average delay. This'scheme uses only local feedback, therefore global objectives cannot be met unless it is appropriately coordinated, hence it is essential to provide for coordination of the local units from higher levels (a slower time scale task). The proposed scheme is applicable not only to ATM systems, but to general queueing systems, where capacity is a scarce resource (for example leased line, or leased capacity systems). The user can optimally allocate capacity in a dynamic fashion, and hence minimise costs whilst maximising throughput, by appropriate formulation of the objective function.

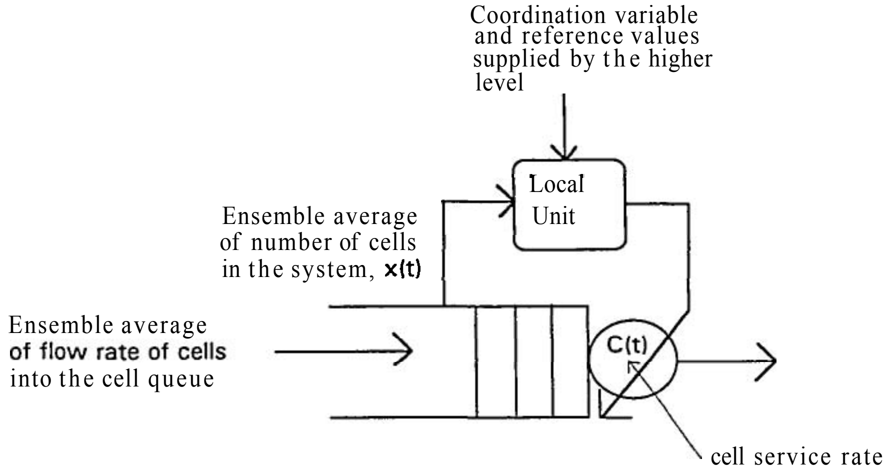


Figure 4.1. The control concept for the dynamic allocation of service-rate.

Using the above control concept, different optimal formulations can be considered. In this thesis we illustrate the concept by formulating and solving the dynamic VP service-rate control (VPC) problem in BISDN. The VP, modelled in terms of a dynamic fluid flow model using equation (4.2) (described later), forms the basis for the control scheme. In a series of papers Pitsillides and colleagues have presented examples of the single node VP case, for a FIFO buffer discipline [177]; the single level, multiple node VP optimal and equilibrium costate solutions, for a FIFO buffer discipline [181]; the nonquadratic multilevel, multiple node VP solution, for a FIFO buffer discipline [178]; the quadratic multilevel, multinode VP solution, for a FIFO server discipline [179]; the quadratic multilevel, multinode VP solution, for a cyclic server discipline [37].

Two illustrative examples are presented in the next section that demonstrate the concept and some computational considerations. Before we discuss the problem formulation and its solution, we digress in order to present the choice of an appropriate mathematical model.

4.4 A unified state modei for VPs

As defined earlier[#1] a VP is a one-way, preestablished, connection between an Origin-Destination (O-D) pair, spanning several ATM switching nodes, into which VCs can be grouped. The VP is a convenient basis for dynamic service-rate control since it represents aggregated flows of traffic (rather than individual connections).

For implementation of a dynamic service-rate control scheme based on the VP concept, we require a model capable of capturing the dynamic behaviour of the VP at various time scales (levels). (See discussion in chapter 2.)

A state model approach offers the principal advantage of flexibility in establishing various performance measures that can be used in the optimal control objective function (for example see Tipper [72]). By using fluid flow arguments one can derive a state model that can describe the essential part of the dynamic behaviour of a queue at the transmission links in the VP in terms of time varying mean quantities.

For a single queue, assuming no losses, the rate of change of the average number of cells queued at the link buffer can be related to the rate of cell arrivals and departures by a differential equation of the form:

$$\dot{x}(t) = -f_{out}(t) + f_{in}(t) \tag{4.1}$$

where:

$x(t)$ - the state of the queue is given by the ensemble average of the number of cells $N(t)$ in the system at time t, ie. $x(t) = E\{N(t)\}$

$f_{out}(t)$ - is the ensemble average of cell flow $O(t)$ out of the queue at time t, ie. $f_{out}(t) = E\{O(t)\}$

$f_{in}(t)$ - is the ensemble average of cell flow $I(t)$ into the queue at time t, ie. $f_{in}(t) = E\{I(t)\}$

This equation is intuitive in nature and can be found in several places in the literature. For example: Agnew [80]; Rider [180]; Filipiak [166]; Tipper et al [18]; Lovegrove et al [19]; Economides et al [170]; Bolot et al [20]; Pitsillides et al [181]; Sharma et al [182]. This "fluid flow" equation is quite general and can be used to model a wide range of queueing and contention systems.

Using fluid flow arguments, as discussed above, a unified nonlinear state equation is presented to describe the dynamic behaviour of a VP. We consider a VP, spanning M

[#1] For references see section 2.3. page 17.

ATM nodes (or switches or switching nodes), with or without blocking, a deterministic propagation delay τ_i between them, carrying S different classes of traffic with (possibly) different priorities, and (possibly) with interference from background traffic at each node. This VP can be described by the following unified fluid flow type nonlinear equation

$$\dot{x}(t) = \underbrace{G(t,\tau,x)\left(C(t,\tau)\otimes e^{s'}\right)}_{-f_{out}(t)} + \underbrace{h(t,x,\lambda)}_{f_{in}(t)} \tag{4.2}$$

where

n is the dimension of the system. It is equal to $n = s \times M$, where s is the number of classes S plus one for the interfering background traffic[#1] (i.e. $s = S+1$), and M is the number of ATM switches.

$x(t)$ is the n x 1 vector of the state of the VP.

$G(t,\tau,x)$ is an n x n nonlinear matrix function, $G = [g_1(t),\dots,g_n(t)]$, of the state vector $x(t)$. It represents the utilisation of the link, and it is dependant on the queueing discipline and probability service distribution. Note that $x \in [0^n, m^n)$ and the utilisation $G(t,\tau,x) \in [0^{n\times n}, e^{n\times n})$, with $G(t,\tau,0^n) = 0^{n\times n}$ and $G(t,\tau,\infty^n) = e^{n\times n}$ ($0^{n\times n}$ and $e^{n\times n}$ are n x n matrices of 0 and 1 respectively). To represent the effects of congestion, it must be a nonnegative strictly concave function.

$h(t,x,\lambda)$ is a nonnegative n x 1 differentiable, nonlinear, vector function of the state vector $x(t)$ and the cell flow rate $\lambda(t)$. It represents the amval rate of cells into the VP, and the blocking, if any, that a queue offers to its input traffic. Note that if there is no blocking (infinite waiting places at the queue) then $h(t,x,\lambda) = \lambda(t)$.

τ is the M x 1 vector of the finite propagation delays between the nodes spanned by a VP,

0^n, e^n, ∞^n are n x 1 vectors of 0, 1 and ∞ respectively.

$C(t,\tau)\otimes e^{s'}$ is an n x 1 vector of the allocated service-rate, that can describe different service disciplines and $\otimes$ is the kronecker tensor product. For example, for a FIFO cell-server discipline, with one type of path traffic and background traffic, spanning 2 nodes, we have $C = [C_1 \;\; C_2]^T$ and $e^{s'} = [1 \;\; 1]^T \Rightarrow C \otimes e^{s'}$ $= [C_1 \;\; C_1 \;\; C_2 \;\; C_2]^T$. Remark: $e^{s'}$ is an s' x 1 vector of ones; s' depends on the service discipline (e.g. for a FIFO $s' = s$).

[#1] **The important role of modelling a "lumped interfering traffic component to allow the interactions between the VPs sharing a finite capacity link server to be taken into account will be discussed later in section 4.5.1.1.**

Equation (4.2) is general and can describe a variety of queueing systems, with or without blocking, featuring multiple classes of traffic, with and without priority, and different queueing disciplines, as for example First-In-First-Out (FIFO) or Cyclic-Queues (CQ).

Equations of this kind are particularly appealing since they can capture the essential dynamics of a single queue carrying S different classes of traffic with different priorities, plus background traffic, with just s differential equations, and for a VP spanning M nodes by just M x s differential equations. This must be compared to a full discrete state space representation for which the number of states is equal to $\prod_{i=1}^{M}(N_{b,i}+1)^{s}$, where N_b is the buffer size. Let us consider an illustrative example to solidify the weight of the argument.

Example: Consider a 3 node (100 cell buffer places at each node) VP with just 1 type of VP traffic and the background traffic, i.e. $M=3$, $S=1$, $s=2$, $N_b=[100\ 100\ 100]$. The number of states are:

fluid flow equation $= 6$ discrete state space $>10^{12}$. VVV

As it can be seen from the above discussion the fluid flow type equations hold promise for control model candidates. Equation (4.2) can also be compared with the fundamental Chapman-Kolmogorov equation for determining the time-dependant state probability distribution for a queue, [17], which is notoriously difficult to solve analytically due to the time varying coefficients (numerical solutions are only available by complex and time consuming mathematical approaches, see for example, Tipper et al [18], Neuts [183], Robertazzi [184], and Reibman [185]). Tipper and Sundareshan [18] and the references therein, provide a good description for modelling different queues using fluid flow approximations; as they show, these methods lend themselves to ease of computation, and are accurate within reasonable bounds (for control purposes).

Examples that show the generality of equation (4.2) to describe different systems are in the appendix 4.11.

4.5 Formulation and solution of the dynamic sewice-rate control problem

We formulate two illustrative examples of the dynamic VP service-rate control scheme (VPC) as follows. In the first example we make use of a nonlinear multilevel optimal control theoretic approach, based on a physical decomposition (of the VP), to formulate a general coordinated decentralised controller for VPC (featuring desirable properties; see discussion in section 4.5.1.4). In the second example we use a single level of control (at the VP level) to formulate an optimal strategy, which turns out to be computationally complex (having to solve a TPBV problem which is notoriously difficult; see discussion in section 4.5.2.1). Hence an approximating suboptimal solution is offered, based on the costates .attaining equilibrium values, which turns out to possess desirable properties.

Their integration to an overall hierarchically organised control scheme is incorporated in the problem formulation.

4.5.1 The VPC optimal service-rate control algorithm: multilevel implementation with a quadratic objective function

We propose to solve the service-rate control problem in a distributed form by using a nonlinear multilevel control methodology (other examples appear in [178], [37]). In particular we derive a distributed optimal control algorithm, which is suitable for non-linear time-delay systems. It dynamically allocates server bandwidth to VPs in BISDN. The form of the solution is suitable for use in a multilevel control structure, with fast local control using local feedback, and slower high level coordination achieved using free parameters at the local level. This is a suitable component for incorporating in a broader overall hierarchical structure. The derivation of this controller takes into account the nonlinear delayed nature of the system. The nonlinear state variable model (equation 4.2) is adopted to describe the behaviour of a VP, for multiple classes of traffic, in terms of time-varying mean quantities (estimates of the ensemble average). The dynamic behaviour is optimised in terms of time varying averages. The problem of interest is to optimally select the service-rate allocations so that a benefit function, or performance measure, is minimised, subject to any physical constraints that may exist. Note that the proposed formulation in addition to taking into account the interactions among the nodes spanned by a VP also allows for the interactions between the VPs sharing a common link. The benefit function is formulated as a tradeoff between different conflicting objectives, ie. cost of network buffer capacity (and delay, and indirectly cell-loss) versus the cost of using extra link capacity. Note that the validity of treating service-rate and cell buffer capacity as substitute resources is demonstrated later (also see section 2.5 and [22], [23]). In terms of this performance objective the control strategy for the optimal service-rate control $C(t)$ (i.e. the service-rate control $C(t)$ which achieves the setpoint and optirnises the performance measure) is given by algorithm 4.1, page 136.

4.5.1.1 problem formulation: multilevel implementation

For each $VP_{j,p}$, (OD pair j, path p) spanning $M_{j,p}$ ATM switching nodes (outgoing links) the following dynamic optimisation problem (P1) has to be solved

Problem P1

Given: The $VP_{j,p}$ topology and the flow rates into the $VP_{j,p}$ queues $\lambda_1^v(t)$ and $\lambda_i^b(t)$
$(v = 1,\ldots,S_{j,p}.\ i = 1,\ldots,M_{j,p}.)$

Minimise: $J_{j,p}^{VPC}$ the $VP_{j,p}$ performance measure

With respect to: $C_{j,p}(t) = [C_{j,p}^1(t),\ldots,C_{j,p}^{M_{j,p}}(t)]^T$

Subject to the following constraints:

$$\text{i) } \dot{x}_{j,p}(t) = G_{j,p}(t,\tau,x)\left(C_{j,p}(t,\tau)\otimes e^{s'_{j,p}}\right) + h_{j,p}(t,x,\lambda) = f_{j,p}(x,C,t,\tau)$$
$$x(t_o) = x_o$$

$$\text{ii) } 0 \le C_{j,p} \le C_{j,p}^{\max}$$

$$\text{iii) } 0 \le x_{j,p} \le x_{j,p}^{\max}$$

$$\text{iv) } \sum_{j=1}^{N_p}\sum_{p=1}^{P_j} \delta^i_{j,p} C^i_{j,p} \le C_i^{link} \qquad i = 1,\ldots,M_{j,p}$$

where:

$$J_{j,p}^{VPC} = \int_{t_o}^{t_f}\left\{(x_{j,p} - x_{j,p}^d)^T Q_{j,p}(x_{j,p} - x_{j,p}^d) + (C_{j,p} - C_{j,p}^d)^T R_{j,p}(C_{j,p} - C_{j,p}^d)\right\}dt$$

is a quadratic objective function in which deviations from the references are penalised in a squared sense.

$\lambda_1^v(t)$ is the cell flow rate into the queue, at the origin node, due to VP traffic, type v.

$\lambda_i^b(t)$ is the background traffic cell flow rate interfering with the VP traffic at each node i along the VP path. Its role in minimising the interactions among the VPs is discussed later under point 1, page 133.

$\dot{x}_{j,p} = f_{j,p}(x_{j,p}, C_{j,p}, \tau_{j,p}, t)$ is an $n_{j,p} = s_{j,p} \times M_{j,p}$ dimensional vector dynamic equation describing the state evolution of the VP, as given by the unified fluid flow equation (4.2). Its initial state is x_o.

$x_{j,p}$ is an $n_{j,p}$ dimensional vector containing the queue state for $VP_{j,p}$. Note that $x_{j,p}^{m,v'}$ is the ensemble average of the number of cells in the queue, for class v' (v' is either $v = 1,\ldots,S_{j,p}$ the path traffic or b the background traffic), at the cell buffer of the outgoing link for the m^{th} node.

$x_{j,p}^{d}$ is an $n_{j,p}$ dimensional vector containing the queue desired state for $VP_{j,p}$. It is either set up by the higher levels based on the desired grade of service, or it could be set to zero, i.e. aim to run the buffers empty.

$x_{j,p}^{\max}$ is an $n_{j,p}$ dimensional vector containing the maximum allowable queue state (supplied by the higher levels) for $VP_{j,p}$. It can be set equal to the finite buffer size, or it can be set by the higher levels so that cell-delay and cell-loss constraints are not exceeded.

$C_{j,p} \otimes e^{s_{j,p}}$ is the calculated $n_{j,p}$ dimensional vector of optimal service-rates.

$C_{j,p}^{d}$ is an $n_{j,p}$ dimensional vector containing the desired optimal service-rate, which is supplied by the higher levels.

$C_{j,p}^{\max}$ is an $n_{j,p}$ dimensional vector containing the maximum service-rate (supplied by the higher levels) which the $VP_{j,p}$ can use. It can be set equal to the link server size, or it can be set by the higher levels so that fairness objectives (among the VPs) can be maintained.

C_i^{link} is the finite link server service-rate at link i.

$\delta^{i}_{j,p}$ takes the value of 1 if $VP_{j,p}$ uses link i.

N_p is the number of VPs in the network.

P, number of predetermined possible W paths between an OD pair. Note that this formulation allows for multiple VPs between an OD pair.

$\boldsymbol{\theta}_{j,p}$ and $R_{j,p}$ are sernipositive definite and positive definite weighting matrices (called weights for short), which can be used (possibly by a higher level) to influence the tradeoff between service-rate and buffer-space (note, as discussed earlier that buffer-space and capacity can be treated as substitute resources).

Remarks:

Constraint i): The dynamic behaviour constraint can be satisfied by using optimal control theoretic techniques to solve for the optimal trajectories and optimal control.

Constraints ii) and iii): The nonnegativity constraint is satisfied by the structure of the state equation (as long as the cell flow rate into the VP and subsequent links is nonnegative–which of course it is). The upper limit constraint can be explicitly taken into account in the derivation of the algorithm, however at the expense of a higher computational burden; see, for example [186]. For simplicity we have chosen to study the case with soft constraints[#1] (the hard constraints are replaced by the weights and/or the references provided by the higher levels of control).

Constraint iv): The link constraint ensures that the sum of the allocated service-rates to VPs sharing a link does not exceed the physical capacity of the link. This constraint can only be satisfied by performing a dynamic constraint optimisation over the whole network at all times. However due to the large state space dimensionality and large geographic spread of the state space the solution to this problem is intractable, for even very small networks. Therefore, we aim to minimise (rather than neutralise) these interactions and suggest the combination of two different approaches:

1. the use of an interfering background traffic component in the model. This allows the optimisation of the VP performance measure, without the need to have any knowledge of the individual interactions from other interfering VPs sharing a link, since these interactions are captured as one lumped interfering traffic component by the background traffic estimate. Note however that only the short term local interfering traffic can be captured (lumped as one parameter) hence network wide objectives cannot be pursued.

2. to use soft constraints. The soft constraints are satisfied by penalising the squared deviations of the calculated controls and states from their desired trajectories. These desired trajectories are decided at a higher level at which the couplings between all the VPs sharing the links in the network are taken care of. In Chapter 5 we discuss the formulation of two higher levels (the VPAM, or the VPOSU), that can provide the references for the service-rates; these references are the optimal service-rate allocations, obtained by performing a

[#1] Soft constraints are commonly employed by control theoreticians, in order to avoid having to solve a (possibly) computationally difficult or intractable control problem global constrained optimisation taking into account the interactions among the VPs (at a much slower time scale than that for the VPC). The references on the buffer state can be chosen as zero so that delay (and cell-loss) is minimised, or they can also be decided by a higher level.

As stated above we make use of both techniques, as they are aimed at minimising both the local short term and the global longer term interactions.

Thus, for computational tractability the constraints ii) & iii) are replaced by soft constraints. Constraint iv) is replaced by a combination of the estimate of the interactions at a node (the background traffic) and soft constraints. Therefore these constraints are removed from the optimisation problem, to obtain the following simplified problem formulation.

Problem P1'

Given: The $VP_{j,p}$ topology and the flow rates into the $VP_{j,p}$ queues $\mathbf{I}_1^v(t)$ and $\mathbf{I}_i^b(t)$ $(v = 1,\ldots,S_{j,p}.\ i = 1,\ldots,M_{j,p}.)$

Minimise: $J_{j,p}^{VPC}$ the $VP_{j,p}$ performance measure

With respect to: $C_{j,p}(t)=[C_{j,p}^1(t),\ldots,C_{j,p}^{M_{j,p}}(t)]^T$

Subject to the following constraint:

$$\text{i) } \dot{x}_{j,p}(t)=G_{j,p}(t,\tau,x)\left(C_{j,p}(t,\tau)\otimes e^{s^{\prime}{}_{j,p}}\right)+h_{j,p}(t,x,\lambda)=f_{j,p}(x,C,t,\tau)$$
$$x(t_o)=x_o$$

Based on the above problem formulation and replacing the expectations with their estimates, various controller implementations can be derived. For example fully centralised, fully decentralised, or coordinated decentralised. In [37] we adopt a coordinated approach based on multilevel control theoretic considerations, and in [181] we develop a suboptimal decentralised controller based on the costate equilibrium solution.

In this section, we extend the results in [37] by treating the delayed case explicitly in the problem formulation of the coordinated decentralised controller. Our approach is to fix the trajectories of the delayed terms at the lower levels of the algorithm and treat these as coordination variables provided by the higher levels. Using a straightforward application of this idea to derive a computationally feasible algorithm proves to be difficult, therefore we use a simple heuristic to derive the multilevel algorithm. We believe that our approach is more practical than the ones proposed in Jamshidi et al [187], Permar et al [188] and Xu [189] since the first two algorithms are based on the linearisation of the plant around a nominal trajectory, (with all its associated problems, as for example performance deterioration and even instability for large deviations from the nominal trajectory; in particular these algorithms are not suitable for systems without a fixed equilibrium point) and the last algorithm is only suitable for linear systems with a particular structure. Our algorithm can handle general nonlinear time delayed systems, and it is an extension of the one proposed by Hassan, Singh and colleagues [190], [191] for the general nonlinear case; which in turn is an extension of the works of Takahara [192] for systems comprised of linear interconnected dynamical subsystems. This method is based on a successive approximation type algorithm. As reported in [193] a solution based on the above algorithm appears to offer a greater computational advantage as compared with the single level solution.

The derivation of the algorithm for nonlinear time delay systems, is developed by modifying the constraints in the problem formulation **P1'** in order to force a coordinated decentralised solution. Note that we have dropped the *j,p* subscripts for notational simplicity.

Modified constraints for problem **P1'** are:

i) $\dot{x}(t) = A(\hat{x},\hat{C},t)x + B(\hat{x},\hat{C},t)C + D(\hat{x},\hat{C},\hat{x}(t-\tau),\hat{C}(t-\tau),t), \quad x(t_0) = x_o$ (4.3)

ii) $x = \hat{x}$ (4.4)

iii) $\mathrm{C} = \hat{C}$ (4.5)

iv) $x(t-7) = \hat{x}(t-\mathrm{r})$ (4.6)

v) $C(t-\tau) = \hat{C}(t-\tau)$ (4.7)

where:

for notational simplicity the time dependencies of $x(t)$, $C(t)$, $\hat{x}(t)$ and $\hat{C}(t)$ have been dropped.

$\hat{C}$, $\hat{x}$, $\hat{C}(t-\tau)$, $\hat{x}(\mathrm{t}-\tau)$ are the trajectories predicted by the higher levels.

$$D(\hat{x},\hat{C},\hat{x}(t-\tau),\hat{C}(t-\mathrm{z}),\mathrm{t}) = f(\hat{x},\hat{C},\hat{x}(t-\tau),\hat{C}(t-\tau),\mathrm{t}) - A(\hat{x},\hat{C},t)\hat{x} - B(\hat{x},\hat{C},t)\hat{C} \quad (4.8)$$

note that $D(\hat{x},\hat{C},\hat{x}(t-\tau),\hat{C}(t-\tau),t)$ contains: the rest of the nonlinearities; off diagonal terms of (4.2); the delayed states; and the delayed service-rates.

$\dot{x}(t) = A(\hat{x},\hat{C},t)x + B(\hat{x},\hat{C},t)C$ is a modified form of (4.2) linearised around the predicted trajectories.

A and B are block diagonal nonstationary matrices with N blocks corresponding to N subsystems. They are functions of $\hat{C}$, $\hat{x}$ and t since the state function is linearised around the predicted trajectories, i.e.

$$A(\hat{x},\hat{C},t) = \left.\frac{\partial f^T}{\partial x}\right|_{\substack{x=\hat{x}\\ C=\hat{C}}} \text{ and } B(\hat{x},\hat{C},t) = \left.\frac{\partial f^T}{\partial C}\right|_{\substack{x=\hat{x}\\ C=\hat{C}}}.$$

$Q(t)$, $R(t)$ are block diagonal matrices with blocks $Q_i(t) \geq 0$, and $R_i(t) > 0$, $i = 1,\ldots,N$.

Due to the dependency of the delayed states on the nondelayed states (similarly for the controls) the derivation of a computationally feasible algorithm (that treats the constraints iv & v explicitly) appears nearly impossible.

Therefore, we propose a simple heuristic to enable us to derive a computationally attractive algorithm that still allows us to treat the delays.

Our heuristic

At the lower levels of the multilevel solution ignore the constraints on the delayed states and controls (constraints iv & v). On the assumption that the predictions of the delayed terms equal their true values the optimal solution can be easily derived. However, constraints iv & v are not satisfied if this assumption is incorrect. Therefore, we satisfy the constraints iv & v, only at the highest level of the hierarchy, by forcing the predictions of the delayed terms to equal their true values. Successive iterations (if they converge) will ensure that the predicted trajectories are the same as the true trajectories.

4.5.1.2 solution; based on the multilevel algorithm

The derived optimal multilevel algorithm can handle system nonlinearities and time delays. It is given by algorithm 4.1; the proof is provided in the appendix 4.111.

Algorithm 4.1 *The VPC* optimal *multilevel* control algorithm, as *formulated* in PI', *for* sewice-rate control *C(t),* is obtained by solving:

at each Local Unit. i *(LUi)*:

$$C_i = -R_i^{-1} B_i^T \nu_i - R_i^{-1} \beta_i + C_i^d \tag{4.9}$$

$$\dot{x}_i = A_i x_i + B_i C_i + D_i \qquad x(t_o) = x_o \tag{4.10}$$

and, at the *higher* levels:

Supremal Unit Level *1* (*SUL1*):

$$\dot{\nu} = -Q(x - x^d) - A^T \nu - \pi \qquad \nu(t_f) = 0 \tag{4.11}$$

$$\pi = \left[\frac{\partial [Ax]}{\partial \hat{x}} + \frac{\partial [BC]}{\partial \hat{x}} + \frac{\partial [D^T]}{\partial \hat{x}} \right] \nu \tag{4.12}$$

$$\beta = \left[\frac{\partial [Ax]}{\partial \hat{C}} + \frac{\partial [BC]}{\partial \hat{C}} + \frac{\partial [D^T]}{\partial \hat{C}} \right] \nu \tag{4.13}$$

Supremal Unit Level 2 (*SUL2*):

$$\hat{x} = x \tag{4.14}$$

$$\hat{C} = C \tag{4.15}$$

$$\hat{x}(t - \tau) = x(t - \tau) \tag{4.16}$$

$$\hat{C}(t - \tau) = C(t - \tau) \tag{4.17}$$

For notational simplicity the dependencies of the variables have been dropped (with the exception of the time delayed variables).

To avoid having to solve the TPBV problem at the same level (see appendix 4.1 and report [194]) we have organised our solution on a three level hierarchical structure in which:

i) the third level predicts x, C and their delayed states (4.14)-(4.17)

ii) the second level uses a form of costate prediction (4.11)-(4.13), and

iii) the lowest level, i.e. at the LUs, solves only simple low order linear equations (4.9) and (4.10).

The problem at each of these levels is simpler than the TPBV problem. .

The control structure and information flows are shown in figure 4.2

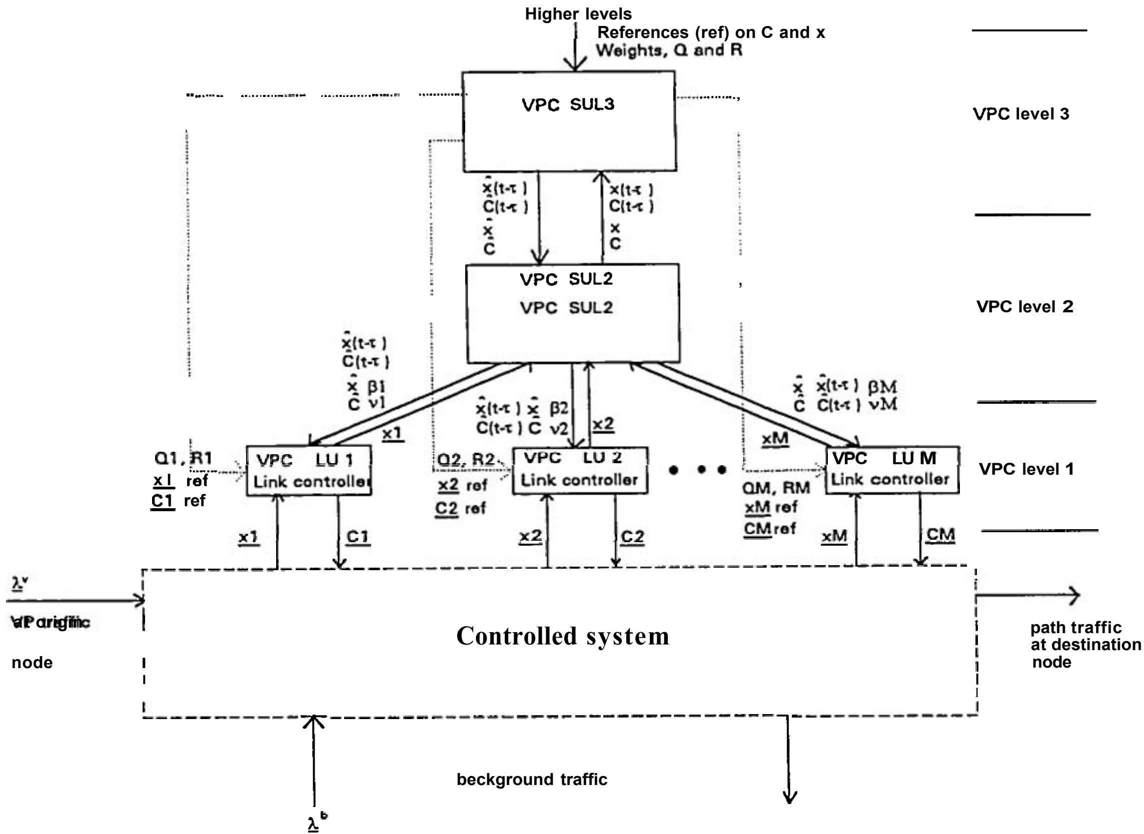


Figure 4.2. The VPC control structure and information flow.

Note that for practical reasons (as for example in the formulation of the lower level Link Server Protocol, LSP; (discussed in chapter 5) it is sometimes more convenient to set the references on service-rate as ratios of the link server rate $C_{j,p}^{link,i}$ at each link i along the VP, rather than absolute values. We therefore briefly present the outline of the modified algorithm 4.1.

Modified algorithm 4.1 outline

A modified objective function is used, which aims to minimise the deviations from the references–provided as ratios of the link server rate:

$$J_{j,p}^{VPC} = \int_{t_o}^{t_f} \{(x_{j,p} - x_{j,p}^d)^T Q_{j,p}(x_{j,p} - x_{j,p}^d) + (C_{j,p} - C_{j,p}^d)^T R_{j,p}(C_{j,p} - C_{j,p}^d)\} dt$$

where

$C_{j,p}$ is an $n_{j,p}$ x 1 vector of the allocated service-rate as a ratio of the link server service-rate, and

$C^{d}_{j,p}$ is an $n_{j,p} \times 1$ vector of the desired service-rate as a ratio of the link server service-rate.

The allocated service-rate at each link, for each service type, is obtained by multiplying element by element the members of $C_{j,p}$ and $\left[C^{link}_{j,p} \otimes e^{s'}\right]$ (where $C^{link}_{j,p}$ is an $M_{j,p} \times 1$ vector of the link service-rates along the VP j,p, and $e^{s'}$ depends on the server discipline, as discussed earlier in section 4.1.4).

The state equation is appropriately modified to reflect this as follows:

$$\dot{x}_{j,p}(t) = G_{j,p}(t,\tau,x)\left[C_{j,p} \circ \left(C^{link}_{j,p}(t,\tau) \otimes e^{s'_{j,p}}\right)\right] + h_{j,p}(t,x,\lambda) = f_{j,p}(x,C,t,\tau)$$
$$x(t_o) = x_o$$

where

$C_{j,p} \circ \left(C^{link}_{j,p}(t,\tau) \otimes e^{s'_{j,p}}\right)$ is an $n_{j,p} \times 1$ vector of the allocated capacities for each service type at each link along the VP j,p

$\circ$ is an element by element multiplier

The derivation of the modified version of algorithm 4.1 follows along the same lines as for algorithm 4.1.

An illustrative example appears next to show the details of algorithm and the control structure (see also [194]).

4.5.1.3 Illustrative example 4.1

To illustrate the details of formulating a more precise multilevel algorithm we employ a fluid flow model, derived from a M/M/1 queueing model, to represent the VP. (An M/D/1 queueing model is, probably, more appropriate to describe an ATM VP, however the M/M/1 queueing model may still give adequate results for our purpose, without the increase in complexity offered by the M/D/1 model; note that the validity of using an M/M/1 queue to approximately describe the queueing delays, even though the packet length is fixed, is discussed by Gerla et al [148]). To use this model to describe a VP, the following standard assumptions are made: cells at any node along the VP arrive as independent Poisson processes; the cells have exponentially distributed lengths; the service time is exponentially distributed with mean $1/C_i$, where C_i is the link server capacity at node **i**; and the buffer lengths at each node are assumed infinite. We also assume that the link has a cyclic queue (CQ) service discipline, and a separate logical queue for each VP. Additionally, we assume that accurate estimates of the ensemble averages are provided. The general model is derived in the appendix 4.11. In this section, we state the VP model for a specific 3-node VP example, and then use it to derive the optimal service-rate control algorithm.

a) Model

In this example we consider a **3** node VP with one stream of VP traffic λ_1^v plus background traffic λ_i^b at each node along the VP. The dynamics of the VP are given by

$$\dot{x}_1^1(t) = -C_1^1(t)\left(\frac{x_1^1(t)}{1+x_1^1(t)}\right) + \lambda_1^1(t).$$

$$\dot{x}_1^b(t) = -C_1^b(t)\left(\frac{x_1^b(t)}{1+x_1^b(t)}\right) + \lambda_1^b(t)$$

$$\dot{x}_2^1(t) = -C_2^1(t)\left(\frac{x_2^1(t)}{1+x_2^1(t)}\right) + C_1^1(t-\tau_1)\left(\frac{x_1^1(t-\tau_1)}{1+x_1^1(t-\tau_1)}\right)$$

$$\dot{x}_2^b(t) = -C_2^b(t)\left(\frac{x_2^b(t)}{1+x_2^b(t)}\right) + \lambda_2^b(t)$$

$$\dot{x}_3^1(t) = -C_3^1(t)\left(\frac{x_3^1(t)}{1+x_3^1(t)}\right) + C_2^1(t-\tau_2)\left(\frac{x_2^1(t-\tau_2)}{1+x_2^1(t-\tau_2)}\right)$$

$$\dot{x}_3^b(t) = -C_3^b(t)\left(\frac{x_3^b(t)}{1+x_3^b(t)}\right) + \lambda_3^b(t)$$

and in the unified form as (dropping the time dependencies, with the exception of the time delayed variables):

$$h(t) = [\lambda_1^1 \quad \lambda_1^b \quad 0 \quad a; \quad 0 \quad a:]$$

$$G(t,\tau,x)=\begin{bmatrix} -\frac{x_1^1}{\chi_1^1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{x_1^b}{\chi_1^b} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{x_2^1}{\chi_2^1} & 0 & 0 & 0 & \frac{x_1^1(t-\tau_1)}{\chi_1^1(t-\tau_1)} & 0 \\ 0 & 0 & 0 & -\frac{x_2^b}{\chi_2^b} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{x_3^1}{\chi_3^1} & 0 & 0 & \frac{x_2^1(t-\tau_2)}{\chi_2^1(t-\tau_2)} \\ 0 & 0 & 0 & 0 & 0 & -\frac{x_3^b}{\chi_3^b} & 0 & 0 \end{bmatrix}$$

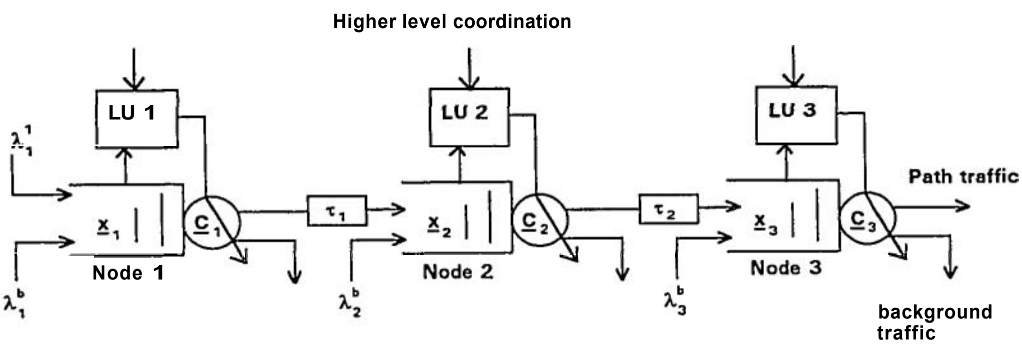
where $\chi_i^j = 1 + x_i^j$

$$C(t,\tau)\otimes e^{s'} = \begin{bmatrix} C_1^1 & C_1^b & C_2^1 & C_2^b & C_3^1 & C_3^b & C_1^1(t-\tau_1) & C_2^1(t-\tau_2) \end{bmatrix}^T$$

b) Decomposition

For the multilevel implementation, we use a geographic decomposition of the VP into three LUs, each of which is physically located at a node along the VP.

$$LU_1 : \underline{x}_1 = \begin{bmatrix} x_1^1 \\ x_1^b \end{bmatrix} \qquad LU_2 : \underline{x}_2 = \begin{bmatrix} x_2^1 \\ x_2^b \end{bmatrix} \qquad LU_3 : \underline{x}_3 = \begin{bmatrix} x_3^1 \\ x_3^b \end{bmatrix}$$



c) The control algorithm

The detailed dynamic equations and controls at each LU are:

At the LU 1

the dynamics are given by

$$\begin{bmatrix} \dot{x}_1^1 \\ \dot{x}_1^b \end{bmatrix} = \begin{bmatrix} -\frac{\hat{C}_1^1}{(\hat{\chi}_1^1)^2} & 0 \\ 0 & -\frac{\hat{C}_1^b}{(\hat{\chi}_1^b)^2} \end{bmatrix} \begin{bmatrix} x_1^1 \\ x_1^b \end{bmatrix} + \begin{bmatrix} -\frac{\hat{x}_1^1}{\hat{\chi}_1^1} & 0 \\ 0 & -\frac{\hat{x}_1^b}{\hat{\chi}_1^b} \end{bmatrix} \begin{bmatrix} C_1^1 \\ C_1^b \end{bmatrix} + \begin{bmatrix} \lambda_1^1 + \frac{\hat{C}_1^1 \hat{x}_1^1}{(\hat{\chi}_1^1)^2} \\ \lambda_1^b + \frac{\hat{C}_1^b \hat{x}_1^b}{(\hat{\chi}_1^b)^2} \end{bmatrix}$$

where

$$\hat{\chi}_i^j = 1 + \hat{x}_i^j$$

and the optimal service-rate controls are calculated by

$$\begin{bmatrix} C_1^1 \\ C_1^b \end{bmatrix} = -R_1^{-1} \begin{bmatrix} -\frac{\hat{x}_1^1}{\hat{\chi}_1^1} & 0 \\ 0 & -\frac{\hat{x}_1^b}{\hat{\chi}_1^b} \end{bmatrix} \begin{bmatrix} \nu_1^1 \\ \nu_1^b \end{bmatrix} - R_1^{-1} \begin{bmatrix} \beta_1^1 \\ \beta_1^b \end{bmatrix} + \begin{bmatrix} C_1^{1,d} \\ C_1^{b,d} \end{bmatrix}.$$

<u>At the LU 2</u>

the dynamics are given by

$$\begin{bmatrix} \dot{x}_2^1 \\ \dot{x}_2^b \end{bmatrix} = \begin{bmatrix} -\frac{\hat{C}_2^1}{(\hat{\chi}_2^1)^2} & 0 \\ 0 & -\frac{\hat{C}_2^b}{(\hat{\chi}_2^b)^2} \end{bmatrix} \begin{bmatrix} x_2^1 \\ x_2^b \end{bmatrix} + \begin{bmatrix} -\frac{\hat{x}_2^1}{\hat{\chi}_2^1} & 0 \\ 0 & -\frac{\hat{x}_2^b}{\hat{\chi}_2^b} \end{bmatrix} \begin{bmatrix} C_2^1 \\ C_2^b \end{bmatrix} + \begin{bmatrix} \frac{\hat{C}_1^1(t-\tau_1)\hat{x}_1^1(t-\tau_1)}{\hat{\chi}_1^1(t-\tau_1)} + \frac{\hat{C}_2^1 \hat{x}_2^1}{(\hat{\chi}_2^1)^2} \\ \lambda_2^b + \frac{\hat{C}_2^b \hat{x}_2^b}{(\hat{\chi}_2^b)^2} \end{bmatrix}$$

and the optimal service-rate controls are calculated by

$$\begin{bmatrix} C_2^1 \\ C_2^b \end{bmatrix} = -R_2^{-1} \begin{bmatrix} -\frac{\hat{x}_2^1}{\hat{\chi}_2^1} & 0 \\ 0 & -\frac{\hat{x}_2^b}{\hat{\chi}_2^b} \end{bmatrix} \begin{bmatrix} \nu_2^1 \\ \nu_2^b \end{bmatrix} - R_2^{-1} \begin{bmatrix} \beta_2^1 \\ \beta_2^b \end{bmatrix} + \begin{bmatrix} C_2^{1,d} \\ C_2^{b,d} \end{bmatrix}.$$

<u>**At** the LU **3**</u>

the dynamics are given by

$$\begin{bmatrix} \dot{x}_3^1 \\ \dot{x}_3^b \end{bmatrix} = \begin{bmatrix} -\frac{\hat{C}_3^1}{(\hat{\chi}_3^1)^2} & 0 \\ 0 & -\frac{\hat{C}_3^b}{(\hat{\chi}_3^b)^2} \end{bmatrix} \begin{bmatrix} x_3^1 \\ x_3^b \end{bmatrix} + \begin{bmatrix} -\frac{\hat{x}_3^1}{\hat{\chi}_3^1} & 0 \\ 0 & -\frac{\hat{x}_3^b}{\hat{\chi}_3^b} \end{bmatrix} \begin{bmatrix} C_3^1 \\ C_3^b \end{bmatrix} + \begin{bmatrix} \frac{\hat{C}_2^1(t-\tau_2)\hat{x}_2^1(t-\tau_2)}{\hat{\chi}_2^1(t-\tau_2)} + \frac{\hat{C}_3^1\hat{x}_3^1}{(\hat{\chi}_3^1)^2} \\ \lambda_3^b + \frac{\hat{C}_3^b\hat{x}_3^b}{(\hat{\chi}_3^b)^2} \end{bmatrix}$$

and the optimal service-rate controls are calculated by

$$\begin{bmatrix} C_3^1 \\ C_3^b \end{bmatrix} = -R_3^{-1} \begin{bmatrix} -\frac{\hat{x}_3^1}{\hat{\chi}_3^1} & 0 \\ 0 & -\frac{\hat{x}_3^b}{\hat{\chi}_3^b} \end{bmatrix} \begin{bmatrix} v_3^1 \\ v_3^b \end{bmatrix} - R_3^{-1} \begin{bmatrix} \beta_3^1 \\ \beta_3^b \end{bmatrix} + \begin{bmatrix} C_3^{1,d} \\ C_3^{b,d} \end{bmatrix}.$$

Note that the predicted trajectories of $\hat{x}$, $\hat{C}$, $\hat{x}(t-\tau)$ and $\hat{C}(t-\tau)$, the references C^d and the coordination variables v and β are supplied by higher levels, and the initial condition of the state is $x(t_o) = x_o$

The coordination part of this algorithm is derived next.

d) Coordination

SUL2 the following coordination variables $v(t)$ and $\beta(t)$ are calculated:

$$\dot{v} = -Q(x - x^d) - A^T v - \pi = -Q(x - x^d) - (A^T + \mathcal{E})v \qquad v(t_f) = 0$$

$$\text{where } \mathcal{E} = \left[\frac{\partial[Ax]}{\partial \hat{x}} + \frac{\partial[BC]}{\partial \hat{x}} + \frac{\partial[D^T]}{\partial \hat{x}} \right]$$

$$\beta = \left[\frac{\partial[Ax]}{\partial \hat{C}} + \frac{\partial[BC]}{\partial \hat{C}} + \frac{\partial[D^T]}{\partial \hat{C}} \right] v = \mathcal{F} v$$

Let $\tilde{x}_i = x_i - \hat{x}_i$ and $\tilde{C}_i = C_i - \hat{C}_i$ be the error between the predicted and actual trajectories.

Then

$$\mathcal{E}=\begin{bmatrix} \frac{2\tilde{x}_1^1C_1^1}{(\chi_1^1)^3}-\frac{\tilde{u}_1^1C_1^1}{(\chi_1^1)^2} & 0 & \frac{\hat{u}_1^1(t-\tau_1)C_1^1(t-\tau_1)}{(\chi_1^1(t-\tau_1))^2} & 0 & 0 & 0 \\ 0 & \frac{2\tilde{x}_1^bC_1^b}{(\chi_1^b)^3}-\frac{\tilde{u}_1^bC_1^b}{(\chi_1^b)^2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{2\tilde{x}_2^1C_2^1}{(\chi_2^1)^3}-\frac{\tilde{u}_2^1C_2^1}{(\chi_2^1)^2} & 0 & \frac{\hat{u}_2^1(t-\tau_2)C_2^1(t-\tau_2)}{(\chi_2^1(t-\tau_2))^2} & 0 \\ 0 & 0 & 0 & \frac{2\tilde{x}_2^bC_2^b}{(\chi_2^b)^3}-\frac{\tilde{u}_2^bC_2^b}{(\chi_2^b)^2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{2\tilde{x}_3^1C_3^1}{(\chi_3^1)^3}-\frac{\tilde{u}_3^1C_3^1}{(\chi_3^1)^2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{2\tilde{x}_3^bC_3^b}{(\chi_3^b)^3}-\frac{\tilde{u}_3^bC_3^b}{(\chi_3^b)^2} \end{bmatrix}$$

and

$$\mathcal{F}=\begin{bmatrix} -\frac{\tilde{x}_1^1}{(\chi_1^1)^2} & 0 & \frac{\hat{x}_1^1(t-\tau_1)}{\chi_1^1(t-\tau_1)} & 0 & 0 & 0 \\ 0 & -\frac{\tilde{x}_1^b}{(\chi_1^b)^2} & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{\tilde{x}_2^1}{(\mathrm{xi})'} & 0 & \frac{\hat{x}_2^1(t-\tau_2)}{\chi_2^1(t-\tau_2)} & 0 \\ 0 & 0 & 0 & -\frac{\tilde{x}_2^b}{(\chi_2^b)^2} & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{\tilde{x}_3^1}{(\mathrm{xi})^2} & 0 \\ 0 & 0 & 0 & 0 & 0 & -\frac{\tilde{x}_3^b}{(\chi_3^b)^2} \end{bmatrix}$$

Observe that the form of these matrices is pleasing, since if the error between actual and predicted trajectories converges to zero, then the only non zero coordination term is for the interactions between the nodes along the VP (π_3 and π_5, i.e. entries $\mathcal{E}(3,1)$ and $\mathcal{E}(3,5)$, and β_3 and β_5, i.e. entries $\mathcal{F}(3,1)$ and $\mathcal{F}(3,5)$).

Also note that the predicted trajectories $\hat{x}$, $\hat{C}$, $\hat{x}(t-\tau)$ and $\hat{C}(t-\tau)$ and the references x^d and C^d are supplied by SUL3.

<u>SUL3</u> the updating of the predicted variables is achieved by mere substitution.

$$\hat{x}=x$$

$$\hat{C} = C$$

$$\hat{x}(t-\tau) = x(t-\tau)$$

$$\hat{C}(t-\tau) = C(t-\tau)$$

Based on the above separation of tasks, the following control structure can be implemented.

e) Control structure

The overall VPC control structure for this example, as well as the information flow are shown in figure 4.3.

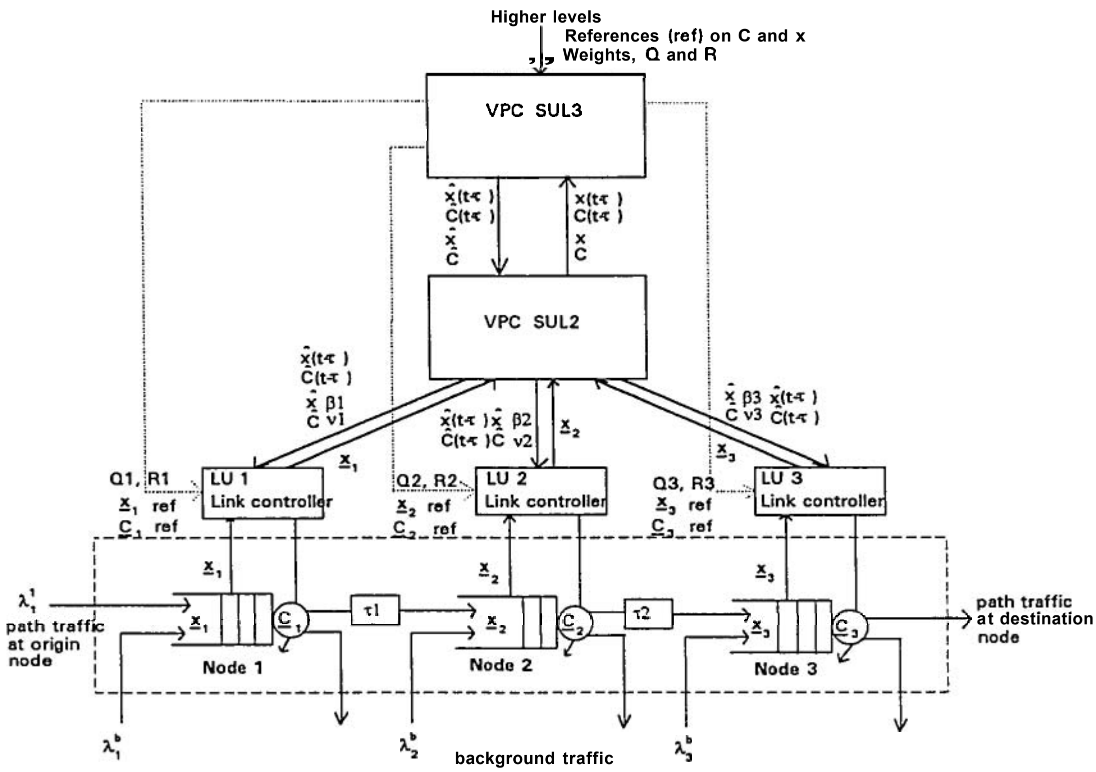


Figure 4.3. The VPC control structure and information flow for the 3-node example.

4.5.1.4 Properties

Algorithm 4.1 possesses the desirable properties of:

- the solution is computationally attractive (as compared to a single level solution, which necessitates the solution of a TPBV problem; see the discussion in section 4.5.2.1, page 150).

- the interactions between the nodes traversed by a VP are dealt with (by the coordination, or interaction, variables of the multilevel controller).
- the interactions between the VPs sharing a link are minimised (by using a combination of the estimated interfering background traffic and the coordination provided by updating the references and/or the weights by a higher level–see earlier discussion).
- its implementation is in a coordinated decentralised form, derived by taking into account the propagation delays.
- the necessary coordination handles are provided by the algorithm, thus making it suitable for incorporating in an overall hierarchically organised structure.

4.5.1.5 Implementation aspects

The implementation of this scheme does not seem to impose a prohibitive computational burden (if it turns out to be computationally difficult for this application then suboptimal policies will be necessary, but at least the optimal solution will be available for comparative purposes; see the illustrative suboptimal strategy derived in example 2 in the next section). However a number of practical questions need to be addressed, as for example:

- Convergence and convergence rate need to be considered. A proof of convergence has been provided, for general hierarchical successive approximation algorithms for non-linear systems, by Hassan and Singh [193]. They proved that there is an open interval of time such that for all values of the final time taken in this interval the algorithm will converge to the optimal solution. However the convergence rate of this type of algorithms still remains an open question.
- Overall stability and stability margins are important. See for example [195] where strong results appear for linear systems, and [196] for general multivariable nonlinear stochastic optimal regulators where strong results appear for the special case of linear in the control variables.
- The sensitivity of the solution to deviations of system and component parameters from their nominal values is important–see for example [197], [198].
- The resulting optimal control law (which belongs to the general class known as Open-Loop-Feedback-Optimal [199]) is dependant on the boundary conditions, and hence has to be repeated if the boundary conditions change, making an off-

line evaluation of the controller coefficients impractical (see Singh et al [28], Chapter 12, for a discussion of different methods of making the control insensitive to initial conditions).

- The ensemble average of the state and the flows must be estimated. A robust, yet computationally simple estimator is required. It is worth pointing out that Warfield et al [98] have presented a simple, recursive, real time estimator of the amval rate.

Since there is limited possibility of incorporating all important robustness issues into one systematic design method defined in mathematical terms [200], in most practical situations one is content with protecting the control structure against perturbations and failures by developing and implementing the jacketing hardware and software. One can use a variety of techniques for this, as for example expert systems, artificial intelligence– see for example [201] for a proposed real-time network management system using a combination of conventional optimisation techniques and artificial intelligence, neural nets etc.

4.5.2 The VPC optimal service-rate control algorithm: single level of control

As a further illustrative example we use a single level of control to derive the optimal service-rate allocations for a single VP. The computational difficulties presented by the derived algorithm (having to solve a TPBV problem) are discussed. Since it is not practical to obtain a closed form solution to this problem, the special case of costate equilibrium is also derived, in which a closed form solution is presented. Its (sub)optimality is compared, via simulation of a specific example, to the full solution in section 4.6.2.

Again the problem of interest is to optimally select the VP service-rate allocations (at the outgoing links spanned by the VP), so that a VP performance measure is minimised, subject to any physical constraints that may exist. We consider a VP described by the unified fluid flow equation (4.2). As already discussed different performance measures can be easily formulated, in terms of the fluid flow equation, due to the generality of the approach. For illustration a linear benefit function is employed here and the optimal service-rate control strategies are derived in terms of the unified VP model (for a quadratic performance index see [194]). To simplify the derivation we only consider a single VP with interfering background traffic, i.e. we assume that the interaction between the VPs sharing a link can be captured by the interfering background traffic.

4.5.2.1 problem formulation: single level implementation

Problem P2

Given: The VP topology and the flow rates into the VP queues $\lambda_1^v(t)$ $\lambda_i^b(t)$ v = 1,...,S. $i = 1,\ldots,$ M.

Minimise: J_{VPC} the VP performance measure

With respect to: $C(t)$

Subject to the following constraints:

(i) $\dot{x}(t)$ the unified VP state equation as defined in (4.2) with $x(t_o) = x_o$

(ii) $0 \le C(t) \le C^{\max}(t)$

(iii) $0 \le x(t) \le x^{\max}(t)$

As discussed, for illustrative purposes we consider the performance measure given by:

$$J_{VPC} = \int_{t0}^{tf} \{ w^x(t)x(t) + w^c(t)C(t) \} dt \tag{4.18}$$

where $w^C(t)$ and $w^x(t)$ are the weights reflecting the tradeoff between service-rate and buffer-space. This performance measure seeks to minimise a linear combination of $x(t)$, the state of the buffer, and $C(t)$ the service-rate allocation. Since service-rate and buffer capacity are substitute resources (discussed in section 4.2.1) we formulate the performance objective so that it allows the tradeoff of one for the other. Note that $x(t)$ is equivalently the delay experienced by cells waiting in the buffer; it is also related to the cell-loss probability. To influence local behaviour, the time varying weights can be updated by a higher level supervisor. The details of a supervisor that updates the weights are not discussed in this thesis (in chapter 5 we discuss a higher level supervisor that updates the references which can also be used to influence the local behaviour; see discussion in section 4.2.1). Note that, if desired, the objective function can be modified to rninimise deviations from the desired states and controls.

Additionally, to simplify the computational aspects of the solution, the following fbrther assumptions on the constraints are used:

Additional simplifying assumption:

- the constraints on $C(t)$ ie. $0 \le C(t) \le C^{\max}(t)$ and the upper limit on the state i.e. $x(t) \le x^{\max}(t)$ will be satisfied by soft constraints, and hence can be removed from the optirnisation problem posed at this level.

Note that the constraints on the network state $x(t)$, i.e. the lower limit of constraint (iii), are always satisfied due to the structure of the state equation.

Therefore using the above simplifying assumption the constraints ii) and iii) are removed from the optimisation problem P2 stated earlier, to obtain:

Problem P2'

Given: The VP topology and the flow rates into the VP queues $\lambda_i^v(t)$ and $\lambda_i^b(t)$
$v = 1,...,S.$ $i = 1,..., M.$

Minimise: $J_{VPC} = \int_{t_o}^{t_f} \{w^x(t)x(t) + w^C(t)C(t)\}dt$ as defined in (4.18)

With respect to: $C(t)$

Subject to the following constraint:

(i) $\dot{x}(t)$ the unified VP state equation as defined in (4.2) with $x(t_o) = x_o$.

In terms of the problem formulation P2', the optimal service-rate $C(t)$ control strategy is given by algorithm 4.2. This strategy achieves the setpoint and optimises the performance measure. Its proof can be found in the appendix 4.111.

Algorithm 4.2: *The* solution of the problem, formulated in *P2'*, for the optimal *service-*rate control *C(t)* is obtained *by* the solution of equations *(4.19) - (4.21)*

$$\dot{\nu}(t) = -w^x(t) - \left\{ \frac{\partial h^T(t,x,\lambda)}{\partial x(t)} + \left(\frac{\partial\left[G(x,t)\left(C(t)\otimes e^s\right)\right]^T}{\partial x(t)} \right) \right\} \nu(t) \qquad \nu(t_f) = 0 \qquad (4.19)$$

$$\dot{x}(t) = G(t,x)\left(C(t)\otimes e^s\right) + h(t,x,\lambda) \qquad x(t_o) = x_o \qquad (4.20)$$

$$0 = w^C(t) + \left(\frac{\partial\left[G(x,t)\left(C(t)\otimes e^s\right)\right]^T}{\partial C(t)} \right) \nu(t) \qquad (4.21)$$

where

$\nu(t)$ is the $n \times 1$ costate vector i.e. $\nu(t) = [\nu_1(t),..., \nu_n(t)]^T$,

$\nu(t_f)$ is the costate variable at the terminal time t_f, and

$x(t_o) = x_o$ is the initial state of the system.

Solving (4.19) - (4.21) will yield the necessary optimal solution for $C(t)$.

The solution of equations (4.19) - (4.21) involves the simultaneous solution of nonlinear differential equations with split boundary conditions—half the boundary conditions being specified at the initial time t_0 (i.e. $x(t_0) = x_o^T$) and the other half defined at the final time t_f (i.e. $\nu(t_f) = 0$)—which is notoriously difficult to solve [202] for any but the simplest cases. It is commonly referred to as the Two Point Boundary Value (TPBV) problem. The solution of the TPBV problem, for the majority of cases, necessitates the use of iterative techniques to determine the solution which satisfies all the boundary conditions. A variety of approaches have been proposed, such as the quasilinearisation techniques, relaxation techniques and decomposition-coordination techniques (see appendix **4.1** for a discussion on the decomposition-coordination techniques) [202], [28], [203]. We will not pursue these techniques any further, but rather concentrate on the analytic solution for the special case of the costates attaining equilibrium.

4.5.2. solution based on the costate equilibrium strategy

To solve (4.19) analytically for $\nu(t)$ is impractical (note though that numerical solutions, as discussed above, are available). However for the special case of the costate variables attaining equilibrium i.e. $\dot{\nu} = 0$, that is for the long-run stationary equilibrium (as $t \rightarrow \infty$, $\dot{\nu}(t) \rightarrow 0$), an analytic solution is obtained with relatively easy algebraic manipulations. This is given by algorithm 4.3; its proof can be found in the appendix 4.111.

Algorithm 4.3: The *costate equilibrium* solution of *VPC*, as formulated in P2′, for the optimal *service-rate* allocation *C(t)* is *given* by solving equation *(4.22) for C(t).*

$$\frac{\partial\left\{w^x(t)\left[\frac{\partial h(t,\lambda)}{\partial x(t)}+\frac{\partial G(t,x)\left(C(t)\otimes e^s\right)}{\partial x(t)}\right]^{-1}\left[G(x,t)\left(C(t)\otimes e^s\right)+h(t,\lambda)\right]\right\}}{\partial C(t)} = w^c(t) \tag{4.22}$$

Algorithm 4.3 possesses the desirable properties of:

- the solution is given in closed form.
- it is computationally simple.

- its implementation can be decentralised, either fully by estimating the flow rate of VP traffic into it, or coordinated by allowing an estimate of the VP flow rate to be provided by a supremal level.
- it can be coordinated by a higher level via the weights and/or the inclusion of references in the objective function.

A crucial question still remains though: that of how quickly, if at all, do the costates attain equilibrium. In the next section, for one specific example, it is demonstrated via simulation that the costates attain equilibrium, however for the general case this remains an open question. Additionally, the estimation of the ensemble averages may prove challenging and the incorporation of a real time estimator, as for example [98], is worthy of further investigation.

4.5.2.3 Illustrative example 4.2

To illustrate the details of formulating a more precise equilibrium costate solution, we employ the commonly used M/M/1 queueing model for each node along the VP (the validity of using an M/M/1 queueing model instead of an M/D/1, to approximately describe the queueing delays, even though the packet length is fixed, is discussed in section 4.5.1.3). To use this model to describe a VP, the following standard assumptions are made: cells at any node along the VP amve as independent Poisson processes; the cells have exponentially distributed lengths; the service time is exponentially distributed with mean $1/C_i$, where C_i is the link cell-server capacity at node i; and the cell buffer lengths at each node are assumed infinite. We also consider a VP with a FIFO buffer discipline, one stream of VP traffic λ_1^v plus background traffic λ_i^b at each node along the VP. The general state equation for this VP model is shown in the appendix **4.II.iii**. To write in a unified equation form, we define (for **M=3** and zero propagation delays):

$$C(t)=\begin{bmatrix}C_1 & C_2 & C_3\end{bmatrix}^T, \quad e^s=[1 \quad 1]^T, \quad h(t)=\begin{bmatrix}\lambda_1^v & \lambda_1^b & 0 & \lambda_2^b & 0 & \lambda_3^b\end{bmatrix}^T, \text{ and}$$

$$G(t,x)=\begin{bmatrix} -\frac{x_1^1}{\chi_1} & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{x_1^b}{\chi_1} & 0 & 0 & 0 & 0 \\ \frac{x_1^1}{\chi_1} & 0 & -\frac{x_2^1}{\chi_2} & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{x_2^b}{\chi_2} & 0 & 0 \\ 0 & 0 & \frac{x_2^1}{\chi_2} & 0 & -\frac{x_3^1}{\chi_3} & 0 \\ 0 & 0 & 0 & 0 & 0 & -\frac{x_3^b}{\chi_3} \end{bmatrix}$$

[for convenience the time dependencies are not explicitly shown and $\chi_i = 1 + x_i^1 + x_i^b$., i=1,2,3].

This allows us to obtain the VP equation in the unified form of equation (4.2). It is derived by matching the steady-state equilibrium point of the fluid flow equation (4.2) with that of the equivalent queueing theory model. As an additional simplifying assumption the deterministic delays between the nodes spanned by the VP, are set to zero, i.e. $\tau_i = 0,\ \ i = 1,\ldots, M-1$. The equilibrium costate solution for the service-rate allocation is then given by [181], [194]

$$C_i^* = \sqrt{\left\{\frac{w_{xi}}{w_{ci}}\left(1 + x_i^v + x_i^b\right)^2\left(\lambda_1^v + \lambda_i^b\right)\right\}} \tag{4.23}$$

Note that this defines a simple feedback relationship, since the optimum service-rate is a function of the local state variables (x_i^v due to path traffic and x_i^b due to background traffic) and the arrival rates (the VP traffic Λ_1^v and the background traffic λ_i^b). Note that an estimate of the VP traffic can either be made locally, or it can be supplied by the originating node of the VP, or by a higher level coordinator. This control is appealing since it can be implemented in a fully decentralised or coordinated decentralised form.

The form of this solution is also pleasing since (pointed out by Konheim [204]) it has a form similar to the one obtained by Kleinrock for the solution of the optimal capacity allocation problem (see page 329 equation 5.26 of [205]). Both solutions are functions of the square root of A (termed the square root capacity assignment formulations by Meditch in [206]). Kleinrock's solution is based on a different objective, that of minimising the steady state delay, and he uses a static M/M/1 queueing model (derived from probabilistic principles) to describe the network state, whereas in our formulation we use a dynamic fluid flow model, and design a controller (using optimal control theory) based on feedback from the queue state. Even so the two solutions to the problem of optimally allocating service-rate are similar in form.

4.6 Performance evaluation

4.6.1 Simulation test bed

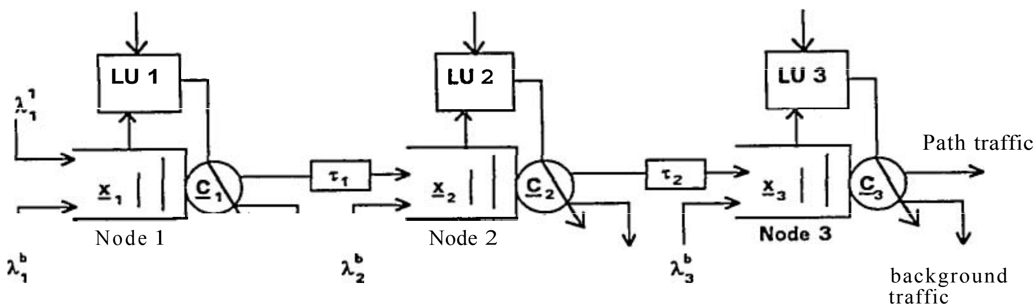
One of the difficulties in studying the performance of the dynamic service-rate allocation scheme via simulation is the lack of previous work in this area. However, using simulation, we will demonstrate some of the interesting aspects of these algorithms, as for example: a comparison of the costate equilibrium solution to the full solution; the existence of a costate equilibrium state; that the algorithms can be influenced (coordinated) by a higher level supervisor; the nature of interactions between the nodes spanned by the VP; and that the multilevel implementation can minimise the effect of the interactions between the nodes spanned by the VP. But firstly we will compare the simulation of the costate equilibrium solution by using both a Discrete Event (DE) simulation and a simulation based on the fluid flow model (control type simulation). It will be shown that the results obtained from the two techniques are very close. On computational grounds we will only use control type simulations for the rest of the simulations because DE simulation requires a large number of independent runs to be averaged to portray system behaviour [72].

4.6.1.1 DE Nonstationarv Simulation (DENS) testbed

Even though there are numerous techniques, as well as commercially available packages, that perform simulations based on DE simulation for steady state measures of probability (as for example the average delay, average throughput etc), not many works deal with the problem of capturing the nonstationary behaviour. For a discussion of DENS we refer the interested reader to the works of Tipper and colleagues: [72] section 3.2 discusses simulation techniques; [18]; [19]; and [207]. In particular, as noted in [72], a large number of independent runs (thousands) must be generated to get an accurate portrayal of the system behaviour (under non stationary conditions). Obviously this approach (DENS) will be computationally difficult when simulating a network of queues, therefore the DENS simulations will be limited to a single node (see appendix 4.IV, case A). Note that the DENS simulations were provided by Tipper [208].

4.6.1.2 Control system type simulation (CSTS) testbed

The CSTS simulation testbed consists of a VP spanning 3 nodes, modelled as a series of M/M/1 queues[#1], with one type of VP traffic and the background traffic. The fluid flow equation (4.2) is used to model a VP, since as already discussed it offers great computational advantages over more complex probabilistic models. An important assumption is that accurate ensemble averages are available.



For the multilevel implementation, we use a geographic decomposition of the VP into three LUs, each of which is physically located at a node along the VP.

$$LU_1:\ \underline{x}_1 = \begin{bmatrix} x_1^1 \\ x_1^b \end{bmatrix} \qquad LU_2:\ \underline{x}_2 = \begin{bmatrix} x_2^1 \\ x_2^b \end{bmatrix} \qquad LU_3:\ \underline{x}_3 = \begin{bmatrix} x_3^1 \\ x_3^b \end{bmatrix}$$

Typically, unless otherwise stated, we simulate the following VP traffic and background traffic:

λ_1^1 the ensemble average of VP traffic is 0.2 cells per time unit

λ_1^b the ensemble average of node 1 background traffic is 0.4 cells per time unit for the stationary case and $0.5 + 0.4\sin(t)$ for the nonstationary case.

λ_2^b the ensemble average of node 2 background traffic is 0.5 cells per time unit

λ_3^b the ensemble average of node 3 background traffic is 0.6 cells per time unit

τ_1 the internodal propagation delay between nodes 1 and 2 is equal to 3 time units

τ_2 the internodal propagation delay between nodes 2 and 3 is equal to 2 time units

4.6.2 Simulation results

Six simulation cases were considered.

case A) Simulation run to demonstrate the validity of the state model

[#1] See earlier discussion in section 4.5.1.3 about the choice of an M/M/1 queue instead of the M/D/1 one.

Two simulations are considered. One using DENS and the other using CSTS. The service-rate is dynamically allocated by using the optimal costate equilibrium equation (4.23) of page 152. Figure 4.6 compares simulation results for stationary background traffic and figure 4.7 for nonstationary background trafiic. As can be seen from these figures the behaviour of the DENS and CSTS simulations have similar trends (the worst observed deviation is 18% and for most cases the deviation is well within 5%). This suggests that the CSTS is useful with its major computational advantages.

See appendix 4.1V.A for the simulation run details.

case B) Simulation run to demonstrate the ability of a higher level to influence local behaviour

This set of simulation runs has been designed to show that the behaviour of the LUs can be influenced by changes in the weighting coefficients and the reference values. These have been proposed as the main coordination tools, via which a higher level can (possibly) influence local behaviour.

changing the weights:

The ability to significantly influence local behaviour is demonstrated in figures 4.8-4.15 (e.g. in figures 4.8 and 4.10 the VP traffic at node 1 ranges over 0.11 to 0.55 – a 400% difference; for the chosen changes in the weights). The tradeoff between buffer-space and service-rate (substitute resources) can also be observed from these figures.

changing references:

The results from this simulations are shown in the figures 4.16-4.21 where the ability to significantly influence the local behaviour is again demonstrated. Note that for the case of a lightly loaded network (case 4) constant allocations over the length of the run have been made (see figures 4.20-4.21), since the service-rate of the server is plentiful and the interactions between the VPs are insignificant (as shown in simulation case C).

See appendix 4.1V.B for the simulation run details.

case C) Simulation run to demonstrate that the interaction terms become more pronounced as the link service-rate becomes scarce

This set of runs shows that the interaction terms between the nodes spanned by the VP become less important, by several orders of magnitude, (figures 4.23 and 4.25) as the service-rate becomes plentifbl (i.e. for a lightly loaded network, see figures 4.20 and 4.21). But as the network load increases and the available service-rate becomes scarce (i.e. for a loaded network, see figures 4.16 - 4.19), then the interaction terms become more important (figures 4.22 and 4.24). This is indicated by an increase in their numerical values (for node 1 the costate variable increases from 0.005 to 0.15).

See appendix 4.1V.C for the simulation run details.

case D) Simulation run to demonstrate the abilitv to minimise the effect of the interactions between the nodes along a VP

For this simulation run the VP traffic at node 1 is changed from 0.1 to 0.2 and then to 0.4 cells per time unit to simulate unexpected VP traffic changes. The behaviour of the multilevel controller and its effect on the state of the system can be observed in figures 4.26, 4.28, 4.30 and 4.31. Figure 4.30 shows that the LU at node 1 allocates service-rate as required so that the influence of an unexpected change in VP traffic on the state of the buffers (shown in figure 4.28 for node 3) and service-rate allocations (shown in figure 4.31 for node 3) downstream is minimised. Figures 4.27 and 4.29 show the case of static service-rate allocations (static allocations are set equal to the service-rate reference values of the dynamic scheme, shown as dashed lines in figures 4.30-4.31). Comparing the dynamic scheme to the static service-rate allocation scheme we can observe a significant improvement on the buffer state of node 3 (80 % for the case when the VP traffic rate equals 0.4 cells/time unit). This demonstrates that the effect of the interactions between the nodes along a VP has been minimised by the use of coordinated decentralised LUs.

See appendix 4.1V.D for the simulation run details.

case E) Simulation run to compare the costate equilibrium solution with the full costate solution

The costate equilibrium solution is compared with the full costate solution, obtained using the multilevel approach of [178]. Figures 4.32 and 4.33 show the costate equilibrium solution. Figures 4.34 and 4.35 show the fill costate solution. It can be seen, for this specific example, that the equilibrium solution does provide comparable control to the full costate solution. The allocated service-rate is within a few percent for a service-rate weight of 0.1, but increasing to about 17 percent for a service-rate weight of 0.25. Full assessment of the suboptimality of the costate equilibrium solution remains a matter requiring further investigation.

See appendix 4.1V.E for the simulation run details.

case F) Simulation run to investigate whether the costates reach equilibrium as time tends to infinity

Figure **4.36** shows the fill **costate** solutions versus time. The full **costate** solution is again obtained by using the multilevel approach of [178]. Figure **4.36** suggests that the costates will reach equilibrium values at about 10 time units (i.e. $\dot{\lambda}(t) \rightarrow 0$ for $t \rightarrow \infty$).

See appendix **4.1V.F** for the simulation run details.

4.6.3 Highlights of the performance evaluation

In the previous section, using simulation the performance of VPC was investigated. For the simulated examples it was demonstrated:

- that a Control System Type Simulation (CSTS) closely resembles a Discrete Event Nonstationary Simulation (DENS).
- the ability to influence the local behaviour of the LUs via changes in the references and/or the weighting coefficients that appear in the objective function. These are proposed as the main coordination tools, via which a higher level can influence local behaviour.
- the interaction between the nodes spanned by a VP becomes more pronounced as the service-rate rate available at a link becomes scarce (as for example during a high load situation).
- the effect of the interactions between the nodes along a VP can be reduced by the use of coordinated decentralised LUs.
- that there is a tradeoff between buffer-space and service-rate.
- that, at least, for the simulated example the costate equilibrium solution exists (i.e. that the costates attain their equilibrium values as time tends to infinity) and that it is close to the fill costate optimum solution.

4.7 Conclusion

In this chapter we have presented the dynamic modelling of the VP, using fluid flow arguments, in a unified form featuring several enhancements from the ones commonly presented in the literature.

A novel service-rate control scheme (offering fast local control and slow coordination) that uses feedback from the network queues was then described. Two illustrative examples of its use to control service-rate at the VP level (VPC) were presented. The interactions within the nodes spanned by a VP, as well as the interactions between the VPs sharing a link are addressed in the problem formulation, and a solution offered. Their integration into an overall hierarchically organised control scheme is included in the problem formulation. In the first example we make use of nonlinear multilevel optimal control theory in the solution of the service-rate control problem at the VP level, to design coordinated fast acting local units. The propagation delay between these local units is taken into account by a novel extension of a multilevel algorithm to the case of nonlinear time-delayed systems. A computationally attractive algorithm (algorithm 4.1 and its modified version), as compared to the single level optimal formulation (algorithm 4.2) that requires the solution of a TPBV problem, has been presented to solve for the optimal service-rate controls for the VPC. As a second illustrative example a suboptimal strategy (algorithm 4.3) is presented (derived from the computationally complex single level formulation), based on the costate equilibrium case (using an example its existence is demonstrated via simulation), which is shown to be very simple computationally—it is obtained in a simple closed form, and therefore iterations toward a solution are not required—yet it appears, via simulation, to be reasonably close to optimality.

The performance of the algorithms has been studied via simulation; the highlights of the performance evaluation appear in section 4.6.3.

In conclusion, our concept for service-rate control using optimal (in particular multilevel) control methodology can be successfully employed for the control of service-rate at the VP level. The minimisation of interactions within a VP and between VPs are also addressed. The solution form (of VPC) is suitable for incorporation in an overall hierarchically organised control scheme.

Appendix 4.1: Nonlinear multilevel control theory

Large-scale systems are found in almost every facet of modem society, as for example: a steel mill; a human organisation; the telephone system, to name but a few. A large number of papers and books have been published dealing with large-scale systems and various approaches have appeared in the literature to handle them (see for example [28], [32], [34], [209]). These approaches can roughly be categorised into: i) the decomposition-coordination approach [27], [28]; ii) the aggregation approach [34], [210]; iii) the multi-time-scaleapproach [211]; and the filly decentralised approach [200], [34]. A good overview appears in [32]. In this section we describe one particular method: the costate-coordination method based on the decomposition-coordination approach.

<u>The costate-coordination method based on the decomposition-coordination approach</u>

The optimisation of nonlinear dynamic systems leads ultimately to the resolution of a nonlinear two-point boundary value (TPBV) problem. The difficulty in solving these nonlinear TPBV problems is well known (see for example Sage et al [202]; also see section 4.2.2.1). Invariably these problems require a successive approximation technique for their solution [202], [193]. Standard successive approximation techniques, such as quasilinearisation, involve the simultaneous solution of a large number of nonlinear differential equations. The solution of even low-order problems is by no means an easy task, let alone the solution of high order ones. Therefore the development of multilevel methods; which allow the solution of large-scale nonlinear problems, as well as provide computational advantages for even low order problems. These are based on *decomposition-coordination*techniques [27], [28], [212]. In particular hierarchical decomposition-coordination techniques have been extensively studied [28], [213], [31], [214], [215]. Various methods of decomposition and coordination provide us with different forms of the solution. We concentrate on the *costate coordination* structure [193], [215] because it avoids the possible existence of the duality gap between the primal and dual problem, which occurs when using the strong duality theorem of Lagrange [216]. But firstly we provide a brief discussion of the concepts of decomposition and coordination [27], [32], [28], [217], [193].

<u>Decomposition</u>

Decomposition of large scale systems can be based on either physical or mathematical grounds. Overlapping or nonoverlapping decompositions are discussed in [218]. Its worth pointing out that at higher levels overlapping decomposition may prove usefil. In this thesis we have concentrated on a physical decomposition, along the physical coupling links. This is prompted by our desire to design coordinated distributed (geographically) local units at the lower levels of a hierarchically organised solution.

Having decomposed the system, now the task is to ensure that the overall optimum solution is obtained from a combination of the local solutions (provided by the local systems). This is achieved by coordination.

Coordination

Coordination is defined as the task of the higher levels to enforce harmonious functioning (in order to obtain the overall optimum solution) of the local subsystems by manipulating their interactions, resolving the conflicts, and adjusting the goal and model interventions. In the case of costate coordination [193] predictive corrective updating rules are used, in a sequence of exchanges and iterations, to strive toward convergence of the individual local solutions toward the overall optimum. This class of hierarchical techniques is based on the prediction of the costate variable (or the adjoint vector), and it has the significant computational advantage of avoiding to solve a TPBV problem at the lower level.

Basic concept

The general idea of the decomposition-coordination approach (based on the costate-coordination method) for the solution of nonlinear large scale systems is:

- write the criterion functional to be minimised in the form of a separable quadratic part and a nonseparable quadratic part;
- write the nonlinear dynamic equations of the overall system in the form of a linear part which is separable by block (good idea is to make them correspond to physical subsystems) and another part that contains the nonlinearities interaction terms and delayed variables;
- organise the lower level so that it consists of low order dynamic optimisation problems, with the interactions, nonlinear terms, and delayed variables treated as constants. The values of these constants are updated by a higher level using coordination variables. Use the costate variable as part of the coordination variables so that the lower level will not have to solve the TPBV problem;
- organise the higher level so that it updates the nonseparable part in the criterion function, the delayed variables, the nonlinear part in the dynamic equations (i.e. the interactions, the delayed variables and the nonlinear terms are treated as

constants by the lower levels), as well as the costate variable (note that by adding a third level the solution of the TPBV problem does not have to be performed at the same level). By successive approximation and information exchanges between it and the lower levels the variables will be iterated to their optimal values.

It is worth pointing out that this technique, which applies the prediction principle, is of the *infeasible* type. This approach does not suffer from the "duality gap" problem that other methods may do (since it solves the primal problem), as for example methods based on the balance principle [27], however being of the infeasible type means that the intermediate results obtained between two successive iterations cannot be used as a suboptimal solution. This is to be contrasted to the *feasible methods* for which the intermediate results can be used as a suboptimal solution. Note however that the feasible methods only apply to a very limited class of problems; those for which the number of control variables is larger than or equal to the number of interconnections.

Appendix 4.11: VP dynamic models described by the unified fluid flow equation

Based on the unified fluid flow equation, (equation 4.2), reproduced here for convenience

$$\dot{x}(t) = G(t,\tau,x)\left(C(t,\tau)\otimes e^{s}\right) + h(t,x,\lambda) \tag{4.2}$$

several application dependant models can be derived. In this section we only derive models for the VP system.

As defined earlier[#1] a VP is a one-way, preestablished, connection between an Origin-Destination (O-D) pair, spanning several ATM switching nodes, into which VCs can be grouped. It is shown in figures 4.4 and 4.5. Multiple traffic service types $(1,\ldots,S)$ are permitted on the VP. The interactions with other VPs (or any other traffic not using the VP concept, as for example a VC connection not utilising a VP) sharing a link at a node $(1,\ldots,M)$ are modelled as background local traffic (note that the VP queue is located at one of the outgoing links of the node, however we will use the term node rather than stating explicitly that the VP queue is located at one of the outgoing links of the node).

For a VP we define:

$C_i(t)$ - cell service rate (capacity) allocated to the VP at node i,

$C_i^v(t)$ - cell service rate (capacity) allocated to the VP traffic, type v, at node i,

$C_i^b(t)$ - cell service rate (capacity) allocated to the background traffic at node i,

$x_i^v(t)$ - state of the VP traffic type v (i.e. ensemble average of number of cells) at the cell-queue of node i,

$x_i^b(t)$ - state of the background traffic (ie ensemble average of number of cells) at the cell-queue of node i,

$\lambda_1^v(t)$ - total arrival rate at cell-queue 1 due to VP traffic, type v

$\lambda_i^b(t)$ - total arrival rate at cell-queue i due to background traffic,

$\gamma_i^v(t)$ -is the VP traffic, type v, leaving the previous node $i-1$ and entering node i, delayed by a deterministic amount τ_{i-1} due to the transmission propagation.

[#1] For references see section 2.3, page 17.

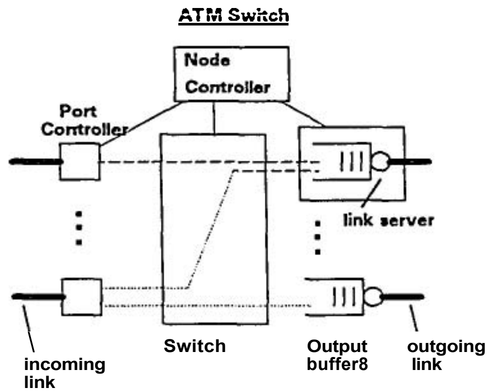


Figure 4.4. ATM switching node.

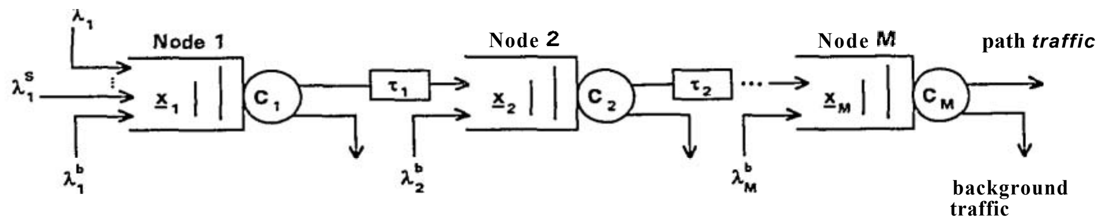


Figure 4.5. A VP modelled by a series of ATM switching nodes.

A commonly used approach to derive the fluid flow equation (4.2) is to match its steady-state equilibrium point with that of an equivalent (in terms of the queueing discipline that best describes the queueing system in question) queueing theory model. This method has been used and validated using simulation by a number of researchers, for example: for the M/M/1/N (Kendall notation[#1]) queue [166], [180], [18], [20]; for multi-class M/D/1 queues with and without priority [182]. It was shown to lead to an accurate approximation for the time-varying mean number in the system. The examples in 4.11.a and 4.11.b adopt this approach. An alternative approach is to use system identification techniques (for example, see the Ph.D. thesis of B. H. Soong [219] for such an approach) to identify the parameters of the utilisation function in the unified fluid flow equation of (4.2).

4.II.a) M/M/1/N_b based models

Firstly we consider the case of a system whose interamval and service distributions are exponentially distributed, with a single server, and N_b cell waiting places in the buffer (note that for an infinite number of cell waiting places it is designated as M/M/1).

i) VP spanning one node, carrying one class of traffic, no interfering background traffic and infinite cell waiting places in the queue

[#1] In 1952, D. G. Kendall described a shorthand notation that captures the essential characteristics of queuing systems. See the *glossary* for a brief description.

The simplest case is of a VP spanning one node, carrying one class of traffic, no background traffic and infinite waiting places in the queue. This queue can be modelled [80], [18], as

$$\dot{x}(t) = -\frac{x(t)}{1+x(t)}C(t) + \lambda(t) = G(x,t)\left(C(t)\otimes e^{s'}\right) + h(t) \qquad (4.\text{II}.1)$$

where

$$G(x,t) = \left[-\frac{x(t)}{1+x(t)}\right]$$

$e^{s'} = 1$, since $s' = 1$, and

$h(t) = \lambda(t)$.

<u>ii) VP spanning one node, with blocking, carrying one class of traffic and no interfering background traffic</u>

The extension to the M/M/1/N_b queue ie. an M/M/1 queue with N_b waiting places, follows from above

$$\dot{x}(t) = -\frac{x(t)}{1+x(t)}C(t) + \lambda(1 - F(x,t) = G(x,t)C(t) + h(t,x) \qquad (4.\text{II}.2)$$

where

$F(x,\mathrm{t})$ is a function of the blocking probability [220, 182].

<u>iii) A state model for VPs, with FIFO queue disciplines, and deterministic propagation delays between the nodes spanning the VP [178]:</u>

The state variable model representing each FIFO queue in the VP, for different classes of traffic v, (assuming infinite cell waiting places in the queues) is given by:

at node 1

$$\dot{x}_1^v(t) = -C_1(t)\left(\frac{x_1^v(t)}{1+x_1^b(t)+\sum_{k=1}^{S} x_1^k(t)}\right) + \lambda_1^v(t) \quad v = 1,\ldots,S,b. \qquad (4.\text{II}.3.\text{a})$$

at node i

$$\dot{x}_i^v(t) = -C_i(t)\left(\frac{x_i^v(t)}{1 + x_i^b(I) + \sum_{k=1}^{S} x_i^k(t)}\right) + \gamma_i^v(t) \qquad \begin{matrix} v = 1,\ldots,\mathrm{S}. \\ i = 2,\ldots,M. \end{matrix}$$

(4.II.3.b)

$$\dot{x}_i^b(t) = -C_i(t)\left(\frac{x_i^b(t)}{1 + x_i^b(t) + \sum_{k=1}^{S} x_i^k(t)}\right) + \lambda_i^b(t) \qquad i = 2,\ldots,M. \tag{4.II.3.c}$$

where

$$\gamma_i^v(t) = \left(C_{i-1}(t - \tau_{i-1}) \frac{x_{i-1}^v(t - \tau_{i-1})}{1 + x_{i-1}^b(t - \tau_{i-1}) + \sum_{k=1}^{S} x_{i-1}^k(t - \tau_{i-1})} \right) \qquad \begin{matrix} v = 1,\ldots,S. \\ i = 2,\ldots,M. \end{matrix} \tag{4.II.4}$$

The dynamics of a single path are hence given by

$$\dot{x} = f(x, C, r, \tau) = [\dot{x}_1, \quad . \quad . \quad . \quad , \dot{x}_M]^T. \tag{4.11.5}$$

This model ***(4.11.3)***and (4.II.4), or ***(4.11.5)***can be used to represent all possible paths for any OD pair.

To write in terms of the unified fluid flow equation, we define

$$h(t) = [\lambda_1 \;\; \lambda_2 \;\; \ldots \;\; \lambda_S \;\;\; \lambda_1^b \;\;\; 0^S \;\;\; \lambda_2^b \;\;\; 0^S \;\;\; \ldots]^T$$

and

$G(t,\tau,x) = [g_1 \quad . \quad . \quad . \quad g_M]^T$, where $g_i = [g_i^1 \quad g_i^2 \;\cdots\; g_i^S \quad g_i^b]^T$

Now for node 1,

$$g_1^1 = \left[-\frac{x_1^1(t)}{\chi_1(t)} \quad 0^S \quad 0^{s(M-1)} \quad \frac{x_1^1(t-\tau_1)}{\chi_1(t-\tau_1)} \quad 0^{s-1} \quad 0^{s(M-2)} \right]$$

$$g_1^2 = \left[0 \quad -\frac{x_1^2(t)}{\chi_1(t)} \quad 0^{S-1} \quad 0^{s(M-1)} \quad 0 \quad \frac{x_1^2(t-\tau_1)}{\chi_1(t-\tau_1)} \quad 0^{s-2} \quad 0^{s(M-2)} \right], \ldots$$

$$g_1^S = \left[0^{S-1} \quad \frac{x_1^S(t)}{\chi_1(I)} \quad 0 \quad 0^{s(M-1)} \quad 0^{s-1} \quad \frac{x_1^S(t-{}_I)}{\chi_1(t-\tau_1)} \quad 0^{s(M-2)} \right], \text{ and}$$

$$g_1^b = \begin{bmatrix} 0^S & -\dfrac{x_1^b(t)}{\chi_1(t)} & 0^{s(M-1)} & 0^{s(M-1)} \end{bmatrix},$$

where $\chi_1 = 1 + x_1^1 + ... + x_1^S + x_1^b$.

Similarly for the rest of the nodes.

$e^{s'} = e^{S+1}$, and

$$C(t,\tau) = \left[C_1(t),...,C_M(t)\ C_1(t-\tau_1),...,C_{M-1}(t-\tau_{M-1})\right]^T$$

to obtain,

$$\dot{x}(t) = G(t,\tau,x)\left(C(t,\tau)\otimes e^{s'}\right) + h(t,x,\lambda)$$

the VP equation in the unified form of (4.2).

<u>**iv) A** state model for **VPs** with a cyclic queue (CQ) server discipline, separate queues for each VP, and deterministic propagation delays between the nodes spanning the VP [37]:</u>

At each node i, we consider a cyclic server discipline [221] (each class in the VP is allocated a service rate of $C_i^v(t)$ and the background traffic is allocated $C_i^b(t)$). The state equation for the VP (assuming infinite cell waiting places in the queues) is:

$$\dot{x}_1^v(t) = -C_1^v(t)\left(\frac{x_1^v(t)}{1+x_1^v(t)}\right) + \lambda_1^v(t) \qquad v = 1,...,S. \quad i = 1.$$

$$\dot{x}_i^v(t) = -C_i^v(t)\left(\frac{x_i^v(t)}{1+x_i^v(t)}\right) + t \qquad v = 1,...,S. \quad i = 2,...,M \tag{4.II.6}$$

$$\dot{x}_i^b(t) = -C_i^b(t)\left(\frac{x_i^b(t)}{1+x_i^b(t)}\right) + \lambda_i^b(t) \quad i = 1,...,M.$$

where

$$\gamma_i^v(t) = \left(C_{i-1}(t-\tau_{i-1})\frac{x_{i-1}^v(t-\tau_{i-1})}{1+x_{i-1}^v(t-\tau_{i-1})}\right) \quad v = 1,...,S. \quad i = 2,...,M. \tag{4.II.7}$$

Following the same approach as the previous example the dynamics of a single path can be described by the unified nonlinear time delay fluid flow equation of (4.2).

$$\dot{x}(t) = G(t,\tau,x)\left(C(t,\tau)\otimes e^{s'}\right)+h(t,x) = \mathbf{f}(x,C,t,\tau) = \left[\dot{x}_1, \quad . \quad . \quad . \quad , \dot{x}_M\right]^T \quad (4.II.8)$$

This model (4.II.6) and (4.II.7), or (4.II.8) can be used to represent all possible paths for any OD pair.

Note that the **detailed** unified equation, for the 3-node case, is shown in example 4.1 of section 4.2

4.II.b) M/D/1/N_b based models

In this section we consider models that can describe M/D/1/N_b (exponential cell interamval times, fixed cell service distribution, 1 server, N_b cell waiting places in the queue) queueing systems carrying S different classes of traffic.

i) VP spanning one node. carrying one class of traffic, no interfering background traffic, and infinite waiting places in the queue

The simplest case is of a VP spanning one node and carrying one class of traffic. This can be modelled as [18]

$$\dot{x}(t) = -\left[\left(x(t)+1\right)-\sqrt{x^2(t)+1}\right]C(t)+\lambda(t) = G(x,t)\left(C(t)\otimes e^{s'}\right)+h(t) \qquad (4.II.9)$$

where

$$G(x,t) = -\left[\left(x(t)+1\right)-\sqrt{x^2(t)+1}\right]$$

$e^{s'} = 1$, since $s' = 1$, and

$h(t) = \lambda(t)$

ii) A state model for VPs, with FIFO aueue disciplines. and deterministic propagation delavs between the nodes spanning the VP

The VP spanning M nodes and carrying S classes of traffic is (assuming infinite cell waiting places in the queue) [182]:

At node 1,

$$\dot{x}_1^v = -\frac{2x_1^v\left[\sqrt{\left(\sum_{k=1}^{S} x_1^k + x_1^b\right)^2 + 1} - \sum_{k=1}^{S} x_1^k + x_1^b\right]}{\sqrt{\left(\sum_{k=1}^{S} x_1^k + x_1^b\right)^2 + 1} - \left(\sum_{k=1}^{S} x_1^k + x_1^b - 1\right)} C_1 + \lambda_1^v, \quad v = 1,\ldots,S,b \qquad (4.\text{II}.10.\text{a})$$

At node i, we have

$$\dot{x}_i^v = -\frac{2x_i^v\left[\sqrt{\left(\sum_{k=1}^{S} x_i^k + x_i^b\right)^2 + 1} - \sum_{k=1}^{S} x_i^k + x_i^b\right]}{\sqrt{\left(\sum_{k=1}^{S} x_i^k + x_i^b\right)^2 + 1} - \left(\sum_{k=1}^{S} x_i^k + x_i^b - 1\right)} C_i + \gamma_i^v, \quad \begin{array}{l} v = 1,\ldots,S. \\ i = 2,\ldots,M. \end{array} \qquad (4.\text{II}.10.\text{b})$$

$$\dot{x}_i^b = -\frac{2x_i^b\left[\sqrt{\left(\sum_{k=1}^{S} x_i^k + x_i^b\right)^2 + 1} - \sum_{k=1}^{S} x_i^k + x_i^b\right]}{\sqrt{\left(\sum_{k=1}^{S} x_i^k + x_i^b\right)^2 + 1} - \left(\sum_{k=1}^{S} x_i^k + x_i^b - 1\right)} C_i + \lambda_i^b, \quad i = 2,\ldots,M. \qquad (4.\text{II}.10.\text{c})$$

where

$$\gamma_i^v = \frac{2x_{i-1}^v\left[\sqrt{\left(\sum_{v=1}^{S} x_{i-1}^v + x_{i-1}^b\right)^2 + 1} - \sum_{v=1}^{S} x_{i-1}^v + x_{i-1}^b\right]}{\sqrt{\left(\sum_{v=1}^{S} x_{i-1}^v + x_{i-1}^b\right)^2 + 1} - \left(\sum_{v=1}^{S} x_{i-1}^v + x_{i-1}^b - 1\right)} C_{i-1}, \quad \begin{array}{l} v = 1,\ldots,S. \\ i = 2,\ldots,M. \end{array} \qquad (4.\text{II}.11)$$

note that all the variables in γ_i^v are delayed by τ_{i-1} (not shown for notational convenience).

The dynamics of a single path can be described by

$$\dot{x}(t) = G(t,\tau,x)\left(C(t,\tau)\otimes e^{s}\right) + h(t,x) = f(x,C,t,\tau) = \left[\dot{x}_1, \quad . \quad . \quad . \quad , \dot{x}_M\right]^T \qquad (4.\text{II}.12)$$

This model (4.II.10) and (4.II.11), or (4.II.12) can be used to represent all possible paths for any OD pair.

To write in terms of the unified fluid flow equation, follow example 4:II.a.iii).

4.II.c) General models

As discussed earlier more general models can be derived by using system identification techniques. If on-line identifiers are used coupled with an appropriate on-line controller design, then we obtain an adaptive system. See chapter **3** for a description of adaptive control systems. This is an interesting line of research to follow.

Appendix 4.III: Proofs of the service-rate control algorithms

The optimisation methods of mathematical programming, dynamic programming, or variational methods (the classical variational calculus or the minimum principle of Pontryagin) can be employed to solve the problem of the optimal service-rate control.

We make use of ***Pontryagin's minimum principle*** [162]. Basically Pontryagin's minimum (or maximum) principle is a set of necessary conditions [162], [202], that must be satisfied by a control variable and trajectory of the state, which minimise the objective function in the general, nonlinear, optimal control problem.

Proof of algorithm 4.1

In this section, we provide the proof of the solution for the optimal service-rate control problem (implementable in a multilevel form).

The proof is based on the simple heuristic described in section 4.5.1.1, page 136. It extends the non time-delay nonlinear case to the time-delayed nonlinear case. For simplicity of exposition, we show the derivation in two phases.

Phase i) At the lower levels of the multilevel solution we ignore the constraints on the delayed states and controls (constraints iv & v). On the assumption that the predictions of the delayed terms equal their true values the optimal solution is easily derived. However the constraints iv & v are not satisfied if the predictions of the delayed terms do not equal their true values. Therefore,

Phase ii) satisfy these constraints (iv & v) but only at the highest level of the hierarchy by forcing the predictions of the delayed terms to equal their true values. Successive iterations (assuming they converge[#1]) ensures that the predicted trajectories are the same as the true trajectories.

Phase 1

By using the objective function and the constraints, we firstly form the Hamiltonian $\mathcal{H}$ (the predictions of the time delayed terms are treated as accurate and known)

$$\begin{aligned}\mathcal{H} = {}^1\!/_2 \left\{ (x-x^d)^T Q (x-x^d) + (C-C^d)^T R (C-C^d) \right\} + \\ \nu^T(t)\left[Ax + BC + D(\hat{x},\hat{C},\hat{x}(t-\tau),\hat{C}(t-\tau),t)\right] + \pi^T(x-\hat{x}) + \beta^T\left(C-\hat{C}\right)\end{aligned} \quad (4.\text{III}.1)$$

where

[#1] A proof of convergence is beyond the scope of this thesis. Our simulative experience shows that there is an optimisation interval over which this algorithm converges. See comments on page 146.

ν is the $nx1$ costate vector i.e. $\nu = [\nu_1, \ldots, \nu_n]^T$,

π, β are $nx1$ and $mx1$ Lagrange multipliers.

Note that, with the exception of the time delayed dependencies, for clarity we have omitted all the other variable's dependencies.

The necessary conditions of the above performance index being optimum [202] are:

$$\frac{\partial \mathcal{H}}{\partial \pi} = 0, \qquad \frac{\partial \mathcal{H}}{\partial \beta} = 0, \qquad \frac{\partial \mathcal{H}}{\partial \gamma} = 0,$$

$$\frac{\partial \mathcal{H}}{\partial \alpha} = 0 \qquad \frac{\partial \mathcal{H}}{\partial \hat{x}} = 0, \qquad \frac{\partial \mathcal{H}}{\partial \hat{C}} = 0,$$

$$\frac{\partial \mathcal{H}}{\partial C} = 0 \qquad \frac{\partial \mathcal{H}}{\partial \nu} = \dot{x}, \qquad \frac{\partial \mathcal{H}}{\partial x} = -\dot{\nu}$$

with split boundary conditions $\nu(t_f) = 0$ and $x(t_0) = x_o$

Solving the above necessary conditions, decomposing (noting that Q, R, A and B are all block diagonal and the coordination variables x^d, C^d, $\hat{x}$, $\hat{C}$, $\hat{C}(t-\tau)$, $\hat{x}(t-\tau)$, ν and β are fixed by higher levels) and organising into 3 levels (lowest level consists of distributed LUs and the two higher levels consist of SUs) we obtain:

At each Local Unit, i (LU i)

$$\frac{\partial \mathcal{H}}{\partial C} = 0 \qquad \Rightarrow \qquad C_i = -R_i^{-1} B_i^T \nu_i - R_i^{-1} \beta_i + C_i^d$$

$$\frac{\partial \mathcal{H}}{\partial \nu} = \dot{x} \qquad \Rightarrow \qquad \dot{x}_i = A_i x_i + B_i C_i + D_i \qquad x_i(t_o) = x_{i,o}$$

and, at the higher levels:

Supremal Level 1 (SUL1):

$$\frac{\partial \mathcal{H}}{\partial x} = -\dot{\nu} \qquad \Rightarrow \qquad \dot{\nu} = -Q(x - x^d) - A^T \nu - \pi \qquad \nu(t_f) = 0$$

$$\frac{\partial \mathcal{H}}{\partial \hat{x}} = 0 \qquad \Rightarrow \qquad \left[\frac{\partial[Ax]}{\partial \hat{x}} + \frac{\partial[BC]}{\partial \hat{x}} + \frac{\partial\left[D^T\right]}{\partial \hat{x}}\right]\nu - \pi = 0$$

$$\frac{\partial \mathcal{H}}{\partial \hat{C}} = 0 \qquad \Rightarrow \qquad \left[\frac{\partial[Ax]}{\partial \hat{C}} + \frac{\partial[BC]}{\partial \hat{C}} + \frac{\partial\left[D^T\right]}{\partial \hat{C}}\right]\nu - \beta = 0$$

Supremal Level 2 (SUL2):

$$\frac{\partial \mathcal{H}}{\partial \pi} = 0 \qquad \Rightarrow \qquad \hat{x} = x$$

$$\frac{\partial \mathcal{H}}{\partial \beta} = 0 \qquad \Rightarrow \qquad \hat{C} = C.$$

Phase 2

The constraints on the delayed terms are satisfied on the third level. Therefore the following updates are also performed at SUL2

$$\frac{\partial \mathcal{H}}{\partial \gamma} = 0 \qquad \Rightarrow \qquad \hat{x}(t-\tau) = x(t-\tau)$$

$$\frac{\partial \mathcal{H}}{\partial \alpha} = 0 \qquad \Rightarrow \qquad \hat{C}(t-\tau) = C(t-\tau).$$

Note that for general, nondelayed, nonlinear systems the optimal solution is derived (in phase 1), and extended to deal with the time delayed terms (in phase 2). Combining phase 1 and phase 2 we obtain equations 4.9-4.17, i.e. algorithm 4.1.

VVV

Proof of algorithm 4.2

We firstly form the Hamiltonian $\mathcal{H}$ (by using the objective function and the constraints)

$$\mathcal{H}(t,x,C) = w^x(t)x(t) + w^c(t)C(t) + \nu^T(t)\left[G(t,x)\left(C(t)\otimes e^s\right) + h(t,x)\right] \qquad (4.\text{III}.2)$$

where

$\nu(t)$ is the $n\times 1$ costate vector i.e. $\nu(t) = \left[\nu_1(t),\ldots,\nu_n(t)\right]^T$.

The necessary first order conditions for optimality of the above performance index are:

$$\frac{\partial \mathcal{H}}{\partial x_i} = -\dot{v}_i \qquad \frac{\partial \mathcal{H}}{\partial v_i} = \dot{x}_i, \qquad \text{and} \qquad \frac{\partial \mathcal{H}}{\partial C} = 0$$

with split boundary conditions $v(t_f) = 0$ and $x(t_0) = x_o$.

Direct calculation gives

$$\dot{v}(t) = -w^x(t) - \left\{ \frac{\partial h^T(t,x,\lambda)}{\partial x(t)} + \left(\frac{\partial \left[G(x,t)\left(C(t) \otimes e^s\right)\right]^T}{\partial x(t)} \right) \right\} v(t) \qquad v(t_f) = 0 \quad (4.\text{III}.3)$$

$$\dot{x}(t) = G(t,x)\left(C(t) \otimes e^s\right) + h(t,x,\lambda) \qquad x(t_0) = x_o \qquad (4.\text{III}.4)$$

$$0 = w^C(t) + \left(\frac{\partial \left[G(x,t)\left(C(t) \otimes e^s\right)\right]^T}{\partial C(t)} \right) v(t) \qquad (4.\text{III}.5)$$

where

$v(t)$ is the n x 1 costate vector i.e. $v(t) = [v_1(t), \ldots, v_n(t)]^T$,

$v(t_f)$ is the costate variable at the terminal time t_f,

$x(t_0) = x_o^T$ is the initial state of the system.

Solving *(4.11.3)*- (4.II.5) will yield the necessary optimal solution for $C(t)$.

∇∇∇

<u>Proof of algorithm 4.3</u>

It follows from algorithm *4.2* by setting the costate derivative to zero, i.e. $\dot{v}(t) = 0$. Solving for the costate variable and substituting back into the Hamiltonian we obtain

$$\mathcal{H}_{mod} = w^x x + w^c C - w^x \left[\frac{\partial h}{\partial x} + \frac{\partial G\left(C \otimes e^s\right)}{\partial x} \right]^{-1} \left[G\left(C \otimes e^s\right) + h \right]$$

For simplicity, we do not show the variable's dependencies.

Next we minimise the modified Hamiltonian $\mathcal{H}_{mod}$ with respect to $C(t)$, i.e.

$$\frac{\partial\left\{w^x(t)\left[\frac{\partial h(t,\lambda)}{\partial x(t)}+\frac{\partial G(t,x)\left(C(t)\otimes e^s\right)}{\partial x(t)}\right]^{-1}\left[G(x,t)\left(C(t)\otimes e^s\right)+h(t,\lambda)\right]\right\}}{\partial C(t)}=w^c(t)$$

Solving for $C(t)$ provides us with the costate equilibrium service-rate allocations.

VVV

Note that the conditions derived above, by using Pontryagin's minimum principle, are necessary conditions for an optimal solution, but they may not be sufficient to ensure that the derived solutions are the globally optimal, i.e. that the global minimum has been obtained. However as the sufficient conditions are almost impossible to derive (one can derive the sufficient conditions only for specific situations, as for example by restricting the control space), in general, one is content with the derivation of the necessary conditions only and then use other means (e.g. simulation) to investigate whether these are also sufficient.

Appendix 4.IV: Simulations A-G

case A) Simulation run to demonstrate the validity of the state model

Two simulations were performed. One using DENS and the other using CSTS.

i) DENS: **An M/M/1** model was simulated using DENS.

Two cases were considered for a single node VP simulation:

run **i)** VP traffic of $\lambda_1^v = 0.2$, and stationary background **traffic** of $\lambda_1^b = 1$.

run **ii)** VP traffic of $\lambda_1^v = 0.2$, and nonstationary background traffic of $\lambda_1^b = 0.5 + 0.4\sin(t)$.

The allocated service-rate was calculated using the optimal costate equilibrium equation (4.23), page 152. For each run two schemes are used to dynamically allocate service-rate: a) using the ensemble average of the state variables; b) using the running average of the state variables, T_{ra} set equal to 2 time units for this simulation study [208]. **A** total of 5000 simulations are used for averaging.

ii) CSTS: An M/M/1 model is simulated with the allocated service-rate calculated from the optimal costate equilibrium equation, (4.23), as for the DENS simulation.

The same two runs, as in the DENS are considered for a single node VP simulation:

run **i)** VP traffic of $\lambda_1^v = 0.2$, and stationary background traffic of $\lambda_1^b = 1$.

run ii) VP traffic of $\lambda_1^v = 0.2$, and nonstationary background traffic of $\lambda_1^b = 0.5 + 0.4\sin(t)$.

The results are plotted in the same figures used for the DENS simulations. Figure 4.6 compares simulation results for stationary background traffic, and figure 4.7 for nonstationary background traffic. As can be seen from these figures the behaviour of the DENS and CSTS have similar trends (the worst observed deviation is 18% and for most cases the deviation is well within 5%). This suggests that the CSTS is useful with its major computational advantages.

Run i) **DENS** and **CSTS** for stationary background traffic

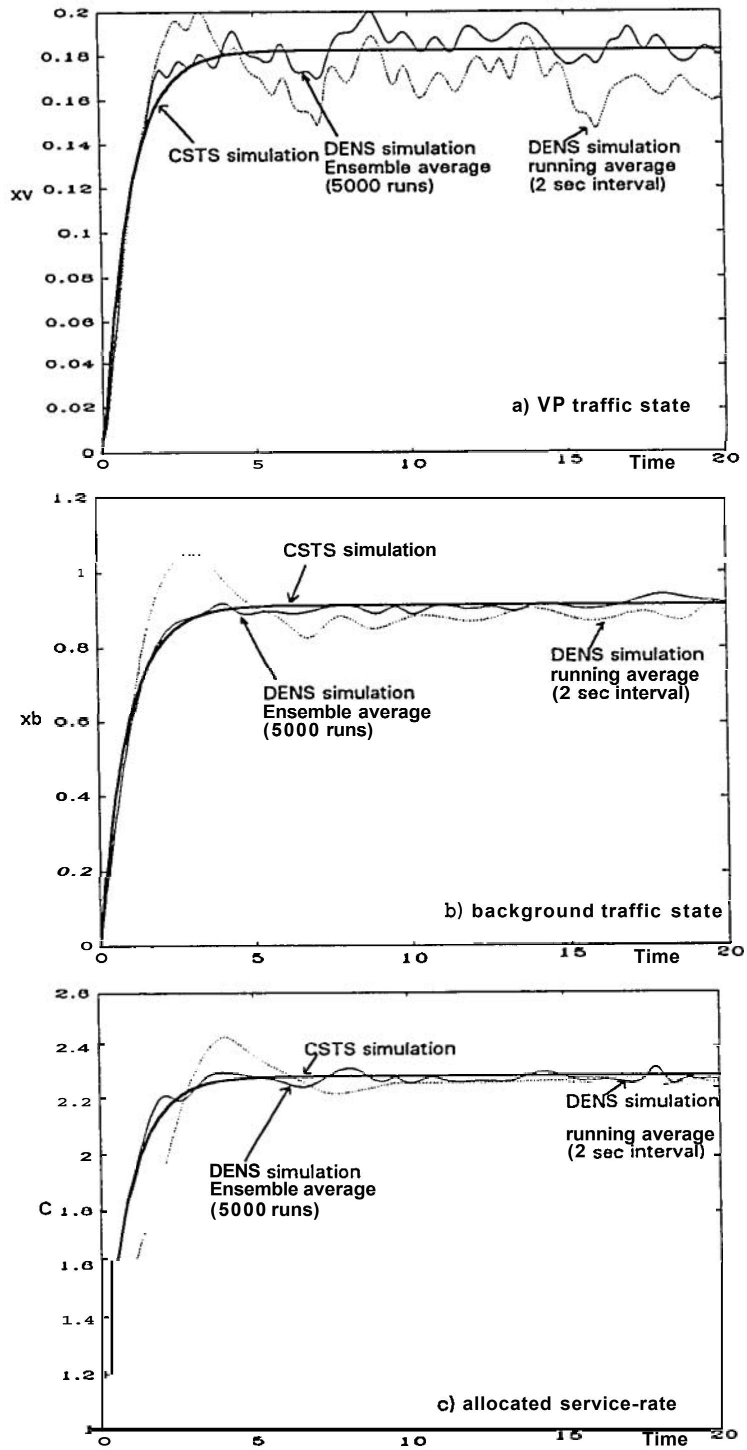


Figure 4.6. **DENS** and **CSTS** for stationary background traffic: a) VP traffic state; b) background traffic state; c) service-rate allocation.

Run ii) DENS and CSTS for nonstationary background traffic

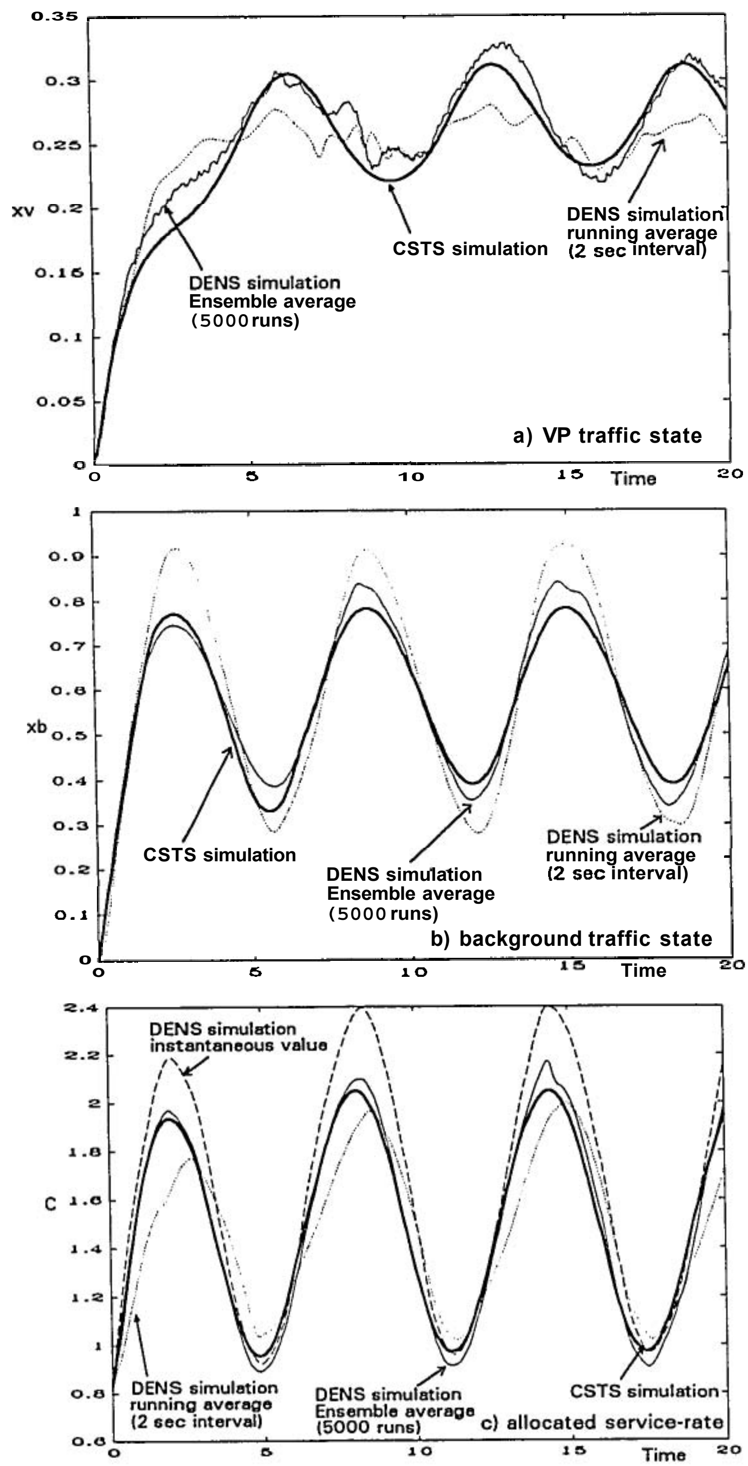


Figure 4.7. DENS and CSTS for nonstationary background traffic: a) VP traffic state; b) background traffic state; c) service-rate allocation.

case B) Simulation run to demonstrate the ability of a higher level to influence local behaviour

This set of simulation runs has been designed to show that the behaviour of the LUs can be influenced by changes in the weighting coefficients and the reference values. These are the main coordination tools, via which a higher level can (possibly) influence local behaviour.

changing the weights:

run i) VP traffic of $A_1^v = 0.2$, and background traffic of $A_1^b = 0.5 + 0.4\sin(t)$, $A_2^b = 0.5$, and $A_3^b = 0.6$. The costate equilibrium strategy, for a FIFO buffer discipline, was implemented. The weights were set to:

1) $w_x = 5$, $w_c = 1$, and

2) $w_x = 1$, $w_c = 5$.

The ability to significantly influence local behaviour is demonstrated in figures 4.8-4.11 (e.g. in figures 4.8 and 4.10 the VP traffic at node 1 ranges over 0.11 to 0.55 –a 400% difference; for the chosen changes in the weights). The tradeoff between buffer-space and service-rate (substitute resources) can also be observed from these figures.

run ii) This example uses a FIFO buffer server discipline. The multilevel algorithm described in section 4.5.1.2 (algorithm 4.1) is implemented. Note that since the buffer discipline is of a FIFO type, the controller cannot discriminate between the VP traffic and the background traffic at a node along the VP, but it can discriminate between nodes along a VP. (Note that a cyclic server discipline with separate logical queues will enable one to also discriminate between the VP traffic and the background traffic by appropriate choice of the weights and references). The weighting coefficients for the queue state, for VP traffic at the node 1 queue, is varied to demonstrate the ability of the higher units to influence local behaviour (figures 4.12-4.15). Note that the allocation of service-rate for VP traffic is significantly affected by the weighting coefficient for queue state (for this simulation run the service-rate changes by about 25% over the chosen weights). As the weight on the state (i.e. Q value; R, the weight on service-rate can also be used) is increased, the queue state is kept at a lower value, but at the cost of extra server capacity. In this example the references for the state and the service-rate at all nodes are kept fixed. Observe that node 2 VP traffic is hardly affected by the change of behaviour at node 1

Above simulations demonstrate that the allocation of service-rate and the state of the traffic are significantly affected by the weighting coefficients. Also note that the tradeoff between buffer-space and service-rate is also demonstrated. The weighting coefficients are only one of the means by which the behaviour of the local units can be influenced. An additional means is via changes in the reference values.

changing references:

For this set of simulation runs the multilevel algorithm (modified algorithm 4.1) described in section 4.5.1.2 is used with a cyclic queue (CQ) server discipline.

run iii) Local units must be able to discriminate between traffic types with different quality of service (QoS) requirements, for example service type 1 may be sensitive to cell-delay while type 2 is not. This discrimination may be (possibly) accomplished by appropriate settings of the reference values for service-rate C^d, and the buffer state x^d (e.g. voice traffic is sensitive to delay, so a state reference of zero could be used, whereas a delay tolerant data traffic type can have its reference set at a higher value). The references (provided as ratios of the link server rate, e.g. for node 1 the desired service-rate for VP traffic is set at $\frac{1}{4}$ of the total link service rate of node 1) are set to:

1) $C^d = [\frac{1}{4} \;\; \frac{3}{4} \;\; \frac{2}{7} \;\; \frac{5}{7} \;\; \frac{2}{8} \;\; \frac{6}{8}]^T$ with $C^{link} = [3 \quad 3 \quad 3]$,

2) $C^d = [\frac{1}{2} \;\; \frac{1}{2} \;\; \frac{2}{7} \;\; \frac{5}{7} \;\; \frac{2}{8} \;\; \frac{6}{8}]^T$ with $C^{link} = [3 \quad 3 \quad 3]$, and

3) $C^d = [\frac{3}{4} \;\; \frac{1}{4} \;\; \frac{2}{7} \;\; \frac{5}{7} \;\; \frac{2}{8} \;\; \frac{6}{8}]^T$ with $C^{link} = [3 \quad 3 \quad 3]$.

Note that for the above three cases (1,2,3) the link server rate, C_i^{link}, $i = 1,2,3$ is equal to 3 cellsltime unit (e.g. for case 1 at node 1 the desired service-rate is $\frac{1}{4} \times 3$ cellsltime unit and for the background traffic the desired service-rate is $\frac{3}{4} \times 3$ cells/time unit). The reference values for VP traffic and background traffic sum to 1 in all cases (e.g. for case 1 at node 1 these rates are $\frac{1}{4}$ and $\frac{3}{4}$). For case 4 the link server rate C_i^{link}, $i = 1,2,3$ is increased to 10 cells/time unit, to simulate the case of a lightly loaded network, i.e. capacity is plentiful.

4) $C^d = [\frac{1}{4} \;\; \frac{3}{4} \;\; \frac{2}{7} \;\; \frac{5}{7} \;\; \frac{2}{8} \;\; \frac{6}{8}]^T$ with $C^{link} = [10 \quad 10 \quad 10]$.

The reference value for the buffer state (x^d) for both VP traffic and background traffic is set to zero. The VP traffic and background traffic rates are as stated in the simulation test bed section. The results from these simulations are shown in the figures 4.16-4.21, where the ability to significantly (by several factors) influence the local behaviour is again demonstrated [e.g. the state of the path traffic (figure 4.16) shows a significant improvement over the state of the background traffic (figure 4.17) as the reference ratio for the VP traffic service rate increases from $\frac{1}{4}$ to $\frac{3}{4}$ of the link rate (figure 4.18), at the expense of the background traffic service-rate (figure 4.19)]. Also note that for the case of a lightly loaded network (case 4) constant allocations of service-rate, over the length of the simulation run, have been made (see figures 4.20-4.21) since the service-rate of the server is plentiful and the interactions between the VPs are insignificant (this is shown in case *C* of the simulation runs).

run i.1) Non stationary input: optimal service-rate allocations; $w_x = 5$, $w_c = 1$. Note that the initial conditions are set to nonzero values.

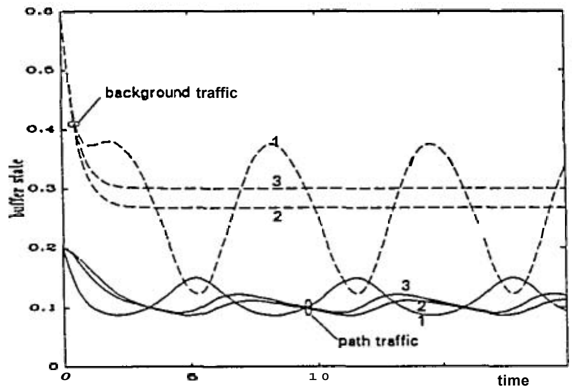


Figure 4.8. VP buffer state: optimal service-rate allocation. $w_x = 5$, $w_c = 1$.

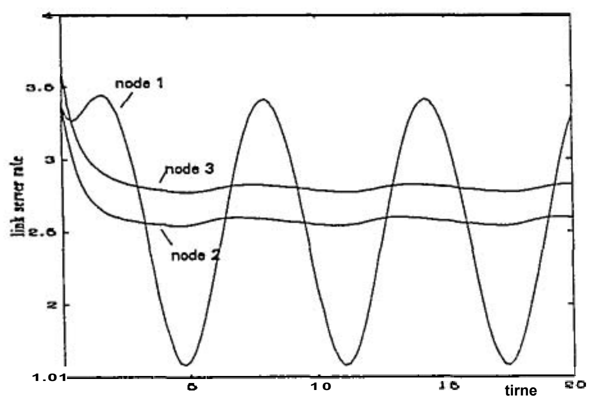


Figure 4.9. VP link server service-rate: optimal service-rate allocation. $w_x = 5$, $w_c = 1$.

run i.2) Non stationary input: optimal service-rate allocations; $w_x = 1$, $w_c = 5$. Note that the initial conditions are set to same nonzero values as in run i.1

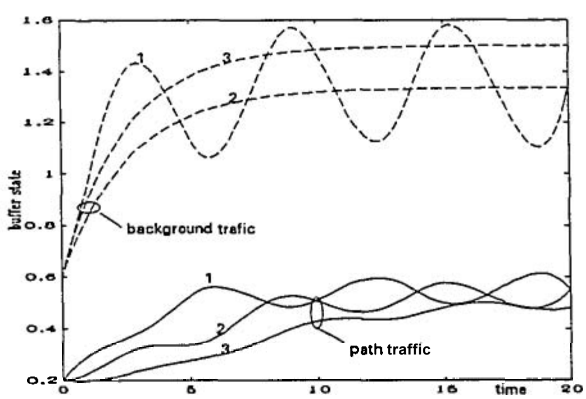


Figure 4.10. VP buffer state: optimal service-rate allocation. $w_x = 1$, $w_c = 5$.

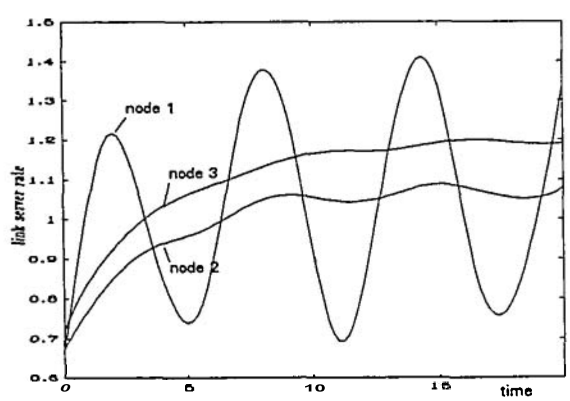


Figure 4.11. VP link server service-rate: optimal service-rate allocation $w_x = 1$, $w_c = 5$.

run ii) This example uses the multilevel implementation with a **FIFO** buffer server discipline.

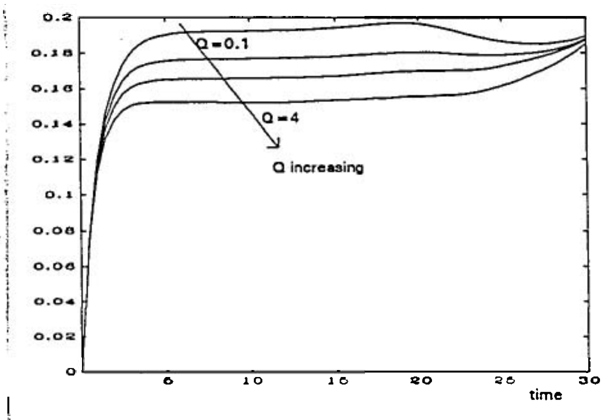


Figure 4.12. Node 1 **VP** traffic <u>buffer state.</u>

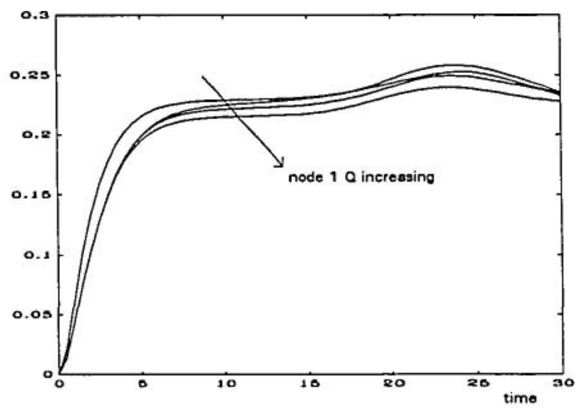


Figure 4.13. Node 2 **VP** traffic <u>buffer state.</u> Note that it is hardly affected by the change of behaviour at node 1.

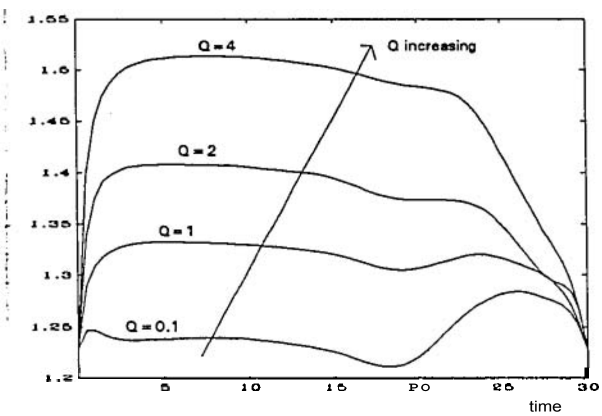


Figure 4.14. Node 1 allocated <u>service-rate.</u>

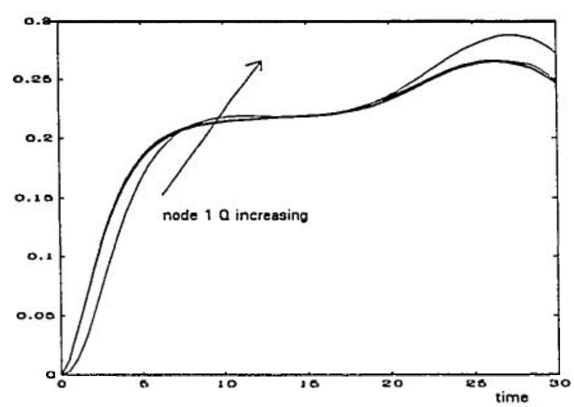


Figure 4.15. Node 2 allocated <u>service-rate.</u> Note that it is hardly affected by the change of behaviour at node 1.

run iii) This example uses the multilevel implementation with a CQ server discipline and with a separate queue for each VP.

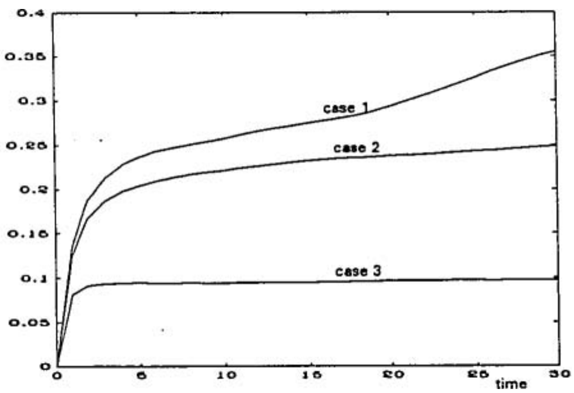


Figure 4.16. VP traffic buffer state at node 1 for changes in the service-rate reference values.

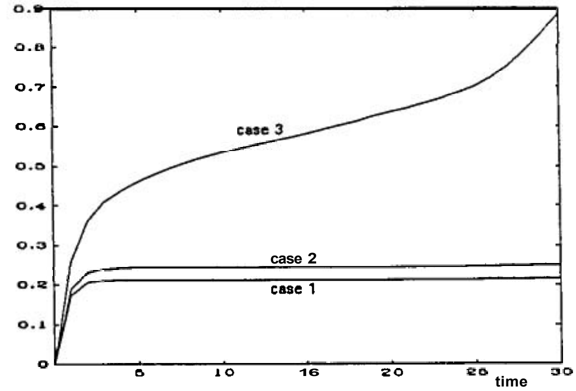


Figure 4.17. Backmound traffic buffer state at node 1 for changes in the service-rate reference values.

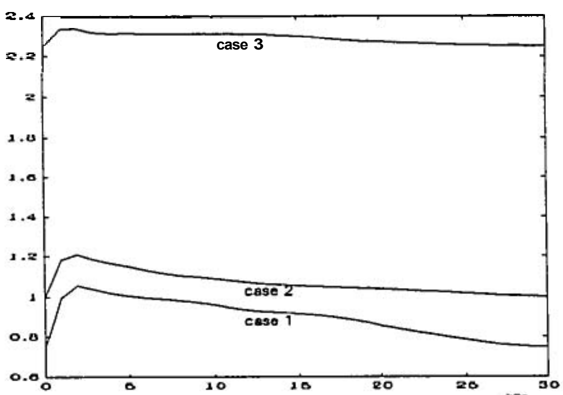


Figure 4.18. Service-rate allocated to VP traffic at node 1 as the references on the service-rate are changed.

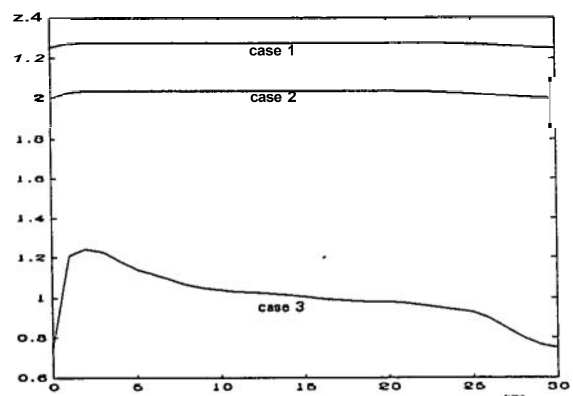


Figure 4.19. Service-rate allocated to backmound traffic at node 1 as the references on the service-rate are changed.

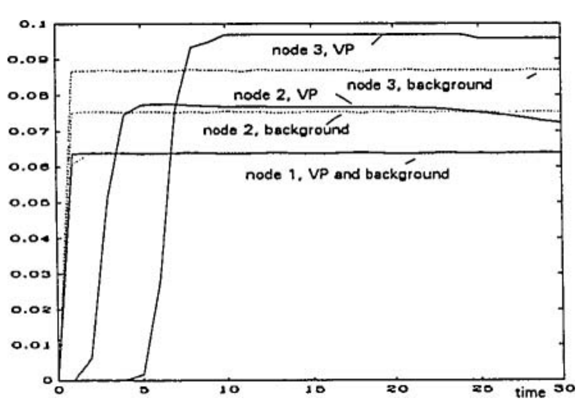


Figure 4.20. Buffer state for plentiful link capacity $C_i^{link} = 10$, $i = 1,2,3$.

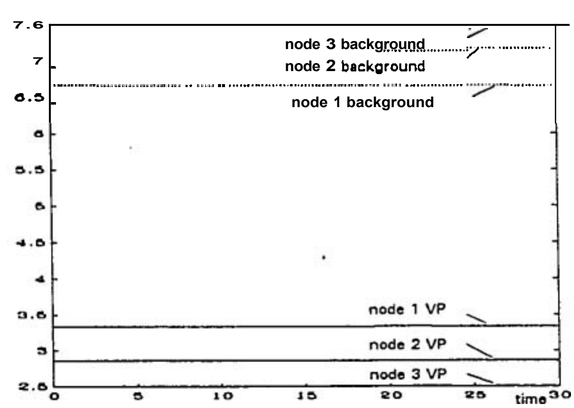


Figure 4.21. Service-rate allocations for plentiful link capacity $C_i^{link} = 10$, $i = 1,2,3$.

case C) Simulation run to demonstrate that the interaction terms become more pronounced as the link service-rate becomes scarce

For this case the same simulation model and control algorithm as in case B) run iii) is used. Two runs were performed:

run i) $C^d = [\tfrac{1}{4} \;\; \tfrac{3}{4} \;\; \tfrac{2}{7} \;\; \tfrac{5}{7} \;\; \tfrac{3}{8} \;\; \tfrac{5}{8}]^T$, with $C^{link} = [3 \;\; 3 \;\; 3]$, and

run ii) $C^d = [\tfrac{1}{4} \;\; \tfrac{3}{4} \;\; \tfrac{2}{7} \;\; \tfrac{5}{7} \;\; \tfrac{3}{8} \;\; \tfrac{5}{8}]^T$, with $C^{link} = [10 \;\; 10 \;\; 10]$.

The interaction terms between the nodes spanned by the VP become less important, by several orders of magnitude, (figures 4.23 and 4.25) as the service-rate becomes plentiful (i.e. for a lightly loaded network, see figures 4.20 and 4.21). But as the network load increases and the available service-rate becomes scarce (i.e. for a loaded network, see figures 4.16 - 4.19), then the interaction terms become more important (figures 4.22 and 4.24). This is indicated by an increase in their numerical values (e.g. for node 1 the costate variable increases from 0.005 to 0.15).

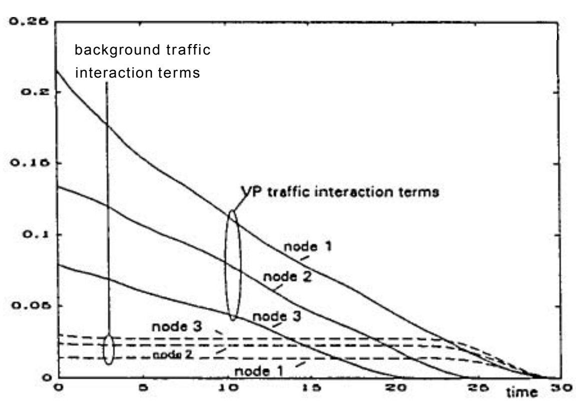


Figure 4.22. The costate variable $\nu(t)$ interaction term for a loaded network: link server capacity is equal to 3 cells/time unit.

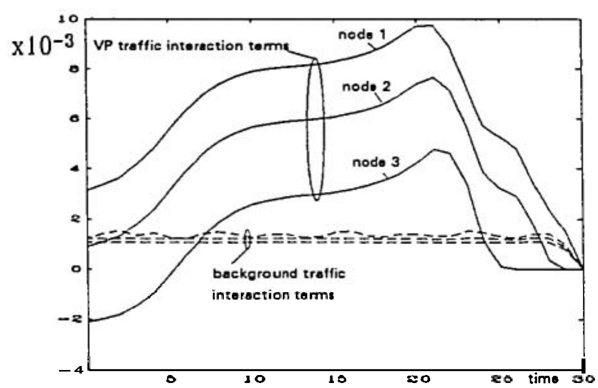


Figure 4.23. The costate variable $\nu(t)$ interaction term for a lightly loaded network: the link server capacity is 10 cells/time unit.

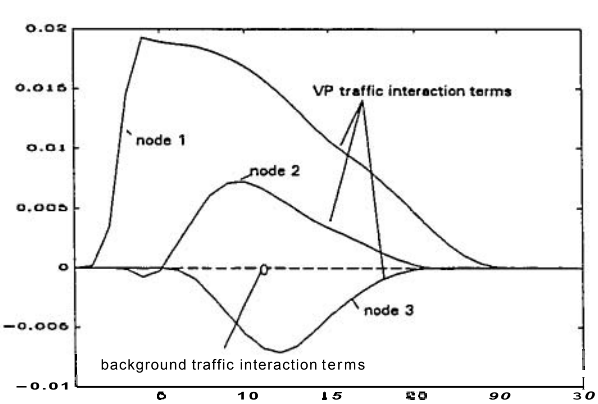


Figure 4.24. The interaction variable $\beta(t)$ for a loaded network: link server capacity is equal to 3 cells/time unit.

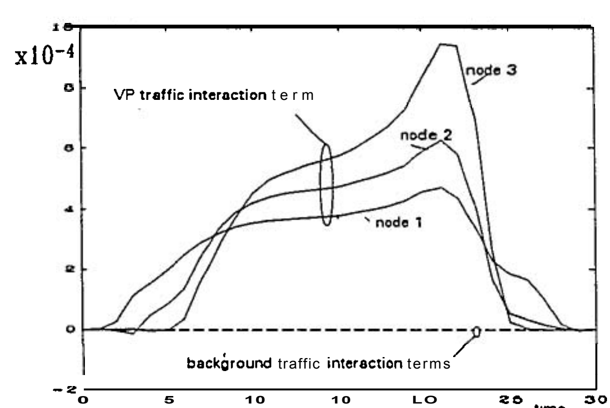


Figure 4.25. The interaction variable $\beta(t)$ for a lightly loaded network: link server capacity is equal to 10 cells/time unit.

case D) Simulation run to demonstrate the abilitv to minimise the effect of the interactions between the nodes alone a VP

Again the simulation model and algorithm of simulation case B) run iii) case 1 is used. For this simulation the VP traffic at node 1 is changed as follows:

. Run i) λ_1^1 the VP traffic is 0.1 cells per time unit.

Run ii) λ_1^1 the VP traffic is 0.2 cells per time unit.

Run iii) λ_1^1 the VP traffic is 0.4 cells per time unit.

The behaviour of the multilevel controller[#1] and its effect on the state of the system can be observed in figures 4.26, 4.28, 4.30 and 4.31. Figure 4.30 shows that the LU at node 1 allocates service-rate as required so that the influence of an unexpected change in VP traffic (simulated by run i-iii) on the state of the buffers (shown in figure 4.28 for node 3) and service-rate allocations (shown in figure 4.31 for node 3) downstream is minimised. Figures 4.27 and 4.29 show the case of static service-rate allocations (static allocations are set equal to the service-rate reference values of the dynamic scheme, i.e $C^{static} = [0.85 \;\; 1.7 \;\; 0.8 \;\; 2.0 \;\; 0.75 \;\; 2.25]^T$, shown as dashed lines in figures 4.30-4.31). Comparing the dynamic scheme to the static service-rate allocation scheme we can observe a significant improvement on the buffer state of node 3 [e.g. there is about 80 % improvement in the buffer occupancy for the dynamic case (figure 4.28) over the static case (figure 4.29) at λ_1^1=0.4 cells per time unit] because of the increase in the service-rate allocation at node 1 (without an increase in service-rate allocations at node 3). This demonstrates that the effect of the interactions between the nodes along a VP has been minimised by the use of coordinated decentralised LUs.

[#1]Note that the propagation delays have been included in this simulation run, and that these delays were included in the derivation of the optimal multilevel controller.

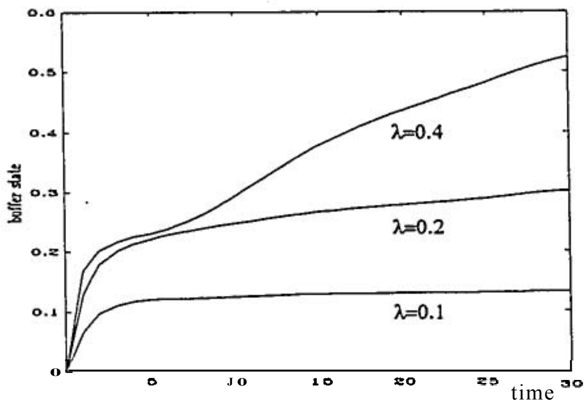


Figure 4.26. Node 1 buffer state: dvnamic service-rate allocation.

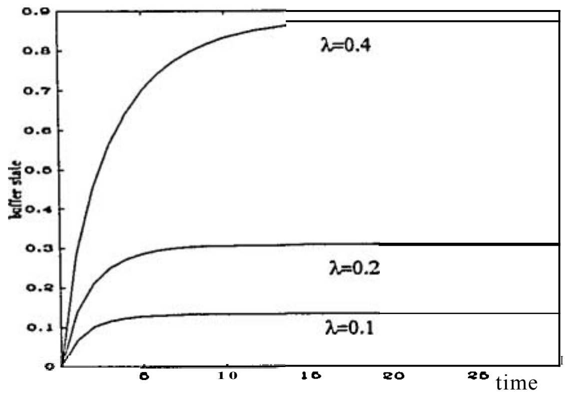


Figure 4.27. Node 1 buffer state: static service-rate allocation.

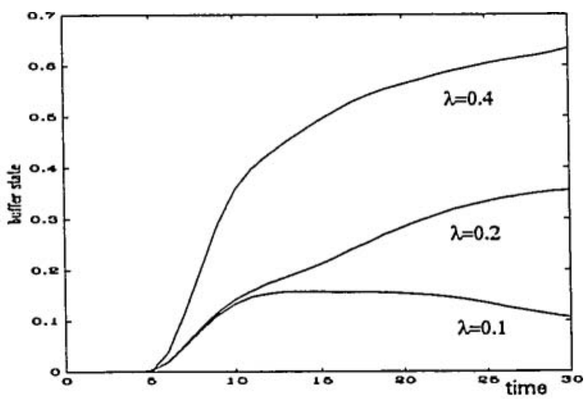


Figure 4.28. Node 3 buffer state: dynamic service-rate allocation.

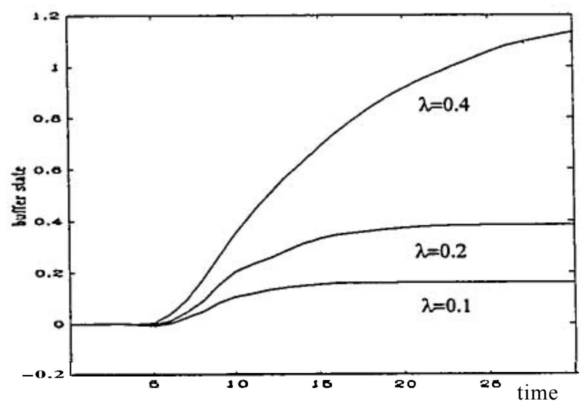


Figure 4.29. Node 3 buffer state: static service-rate allocation.

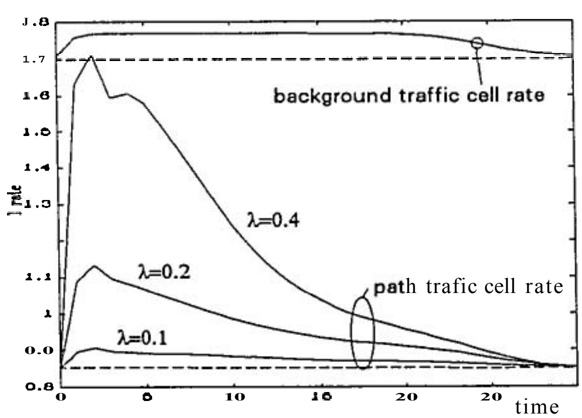


Figure 4.30. Node 1 dynamically allocated service-rate.

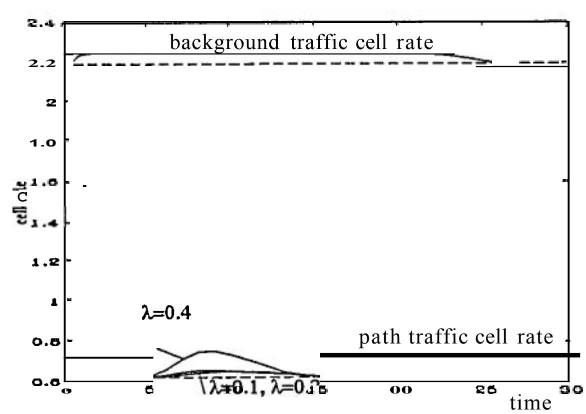


Figure 4.31. Node 3 dynamically allocated service-rate.

case E) Simulation run to compare the costate eauilibrium solution with the full costate solution

The costate equilibrium solution is compared with the full costate solution, obtained using the multilevel approach of [178]. Figures 4.32 and 4.33 show the costate equilibrium solution. Figures 4.34 and 4.35 show the full costate solution. It can be seen, for this specific example, that the equilibrium solution does provide comparable control to the full costate solution. The allocated service-rate is within a few percent for a service-rate weight of 0.1, but increasing to about 17 percent for a service-rate weight of 0.25. Full assessment of the suboptimality of the costate equilibrium solution remains a matter requiring further investigation.

run i) The equilibrium costate scheme is implemented for the standard VP parameters.

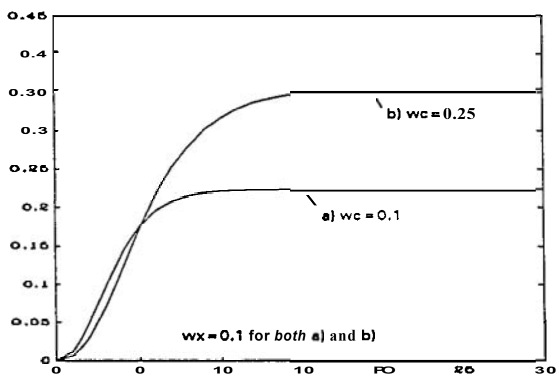


Figure 4.32. Node 3 VP traffic buffer state; service-rate allocated dynamically using the eauilibrium costate solution.

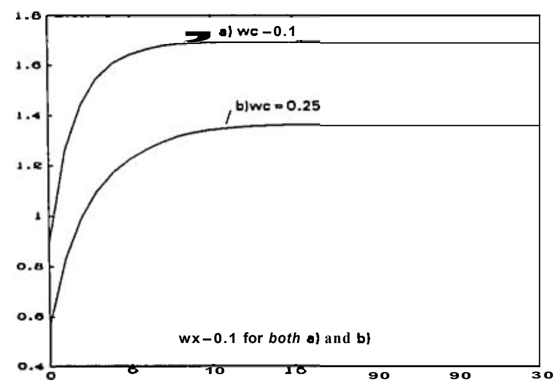


Figure 4.33. Node 3 Service-rate allocation using the equilibrium costate solution.

Run ii) The optimal full costate scheme is implemented for the standard VP parameters.

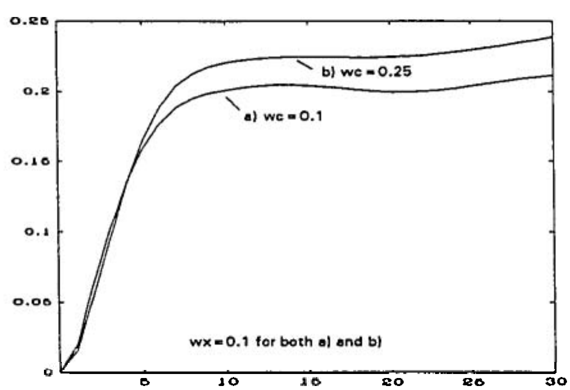


Figure 4.34. Node 3 VP traffic buffer state; service-rate allocated dynamically using full costate solution.

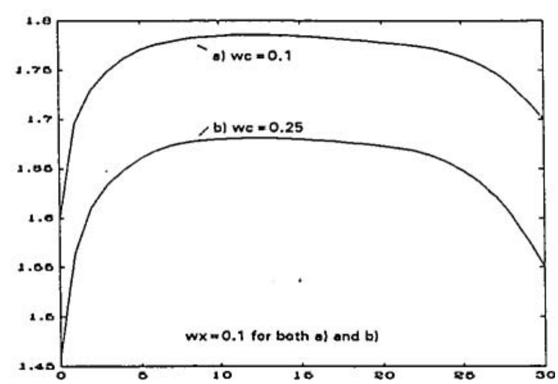


Figure 4.35. Node 3 service-rate allocated dynamically using full costate solution.

case F) Simulation run to investigate whether the costates reach equilibrium as time tends to infinity

Figure **4.36** shows the full costate solutions versus time. The full costate solution is again obtained by using the multilevel approach of [178]. Figure 4.36 suggests that the costates will reach equilibrium values at about 10 time units (i.e. $\dot{v}(t) \to 0$ for $t \to \infty$).

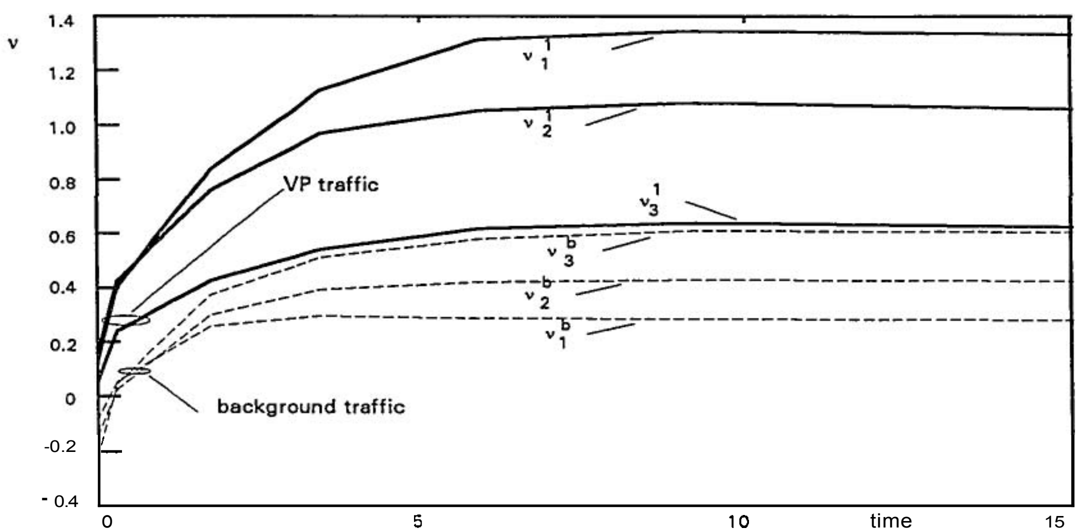


Figure 4.36. Demonstration of the existence of the costate equilibrium (i.e. $\dot{v}(t) \to 0$) for $t \to \infty$.

CHAPTER 5

AN ILLUSTRATIVE HIERARCHICAL STRUCTURE FOR THE CONTROL OF SERVICE-RATE

5.1. Introduction

Till now we have focused on three generic functions for control in BISDN: in Chapter 3 the integrated adaptive CAC and flow control (ACFC); and in Chapter 4 the VP service-rate control (VPC) at the VP level. Even though these schemes can operate independently (pursue their own goals), in isolation these schemes cannot take into account network wide objectives.

In this section, we illustrate the hierarchical approach (see discussion in Chapter 2) for the solution of a particular control problem—that of service-rate control. A schematic of the proposed hierarchical structure can be seen in figure 5.11, page 206. It features four vertical levels: The VP Allocation and Management (VPAM); the VP Overall Supremal Unit (VPOSU); the VP Control (VPC); and the Link Service Protocol (LSP).

In section 5.2, we formulate VPAM; the highest level[#1] of the proposed illustrative hierarchy. VPAM is formulated to operate at a slow time scale (much slower than VPOSU). Its objectives are global (network wide), can be multiple, possibly conflicting and noncommensurate. The solution of VPAM, (being global and operating on a very slow time scale) minimises the interactions between the VPs sharing a link, but only in the longer term. On its own, the solution offered by VPAM is not adequate. It cannot handle the very short to long term traffic fluctuations. Therefore we use a vertical decomposition (see section 5.3) to formulate VPOSU. This level is coordinated by the output of VPAM. It operates on medium to long term time scales. Due to the geographic distribution of the network a horizontal decomposition is also essential, with the LUs of VPOSU located at the origin nodes of the VPs. Now, since the solution of VPOSU is for the medium to long term, a further vertical decomposition is necessary to form the VPC level (see section 5.4). The VPC level is coordinated by the output of VPOSU. Within the VPC level further vertical and horizontal decompositions are carried out in order to obtain a coordinated decentralised solution. Decentralisation is by the LUs of VPC. Coordination is by the VPC SUs. These are implemented at the

[#1] **Higher levels may be formulated, as for example the creation and deletion of VPs. In this illustrative example we will assume that VPAM is the highest level of the hierarchy.**

outgoing links of each node spanned by the VP. Note that VPC is derived in Chapter 4. As the solutions offered by the VPC are for short to medium term, they cannot minimise the very short variations (on cell-time scale). Hence for the particular case of a Cyclic Queue (CQ) server discipline with a separate queue for each VP, a heuristic LSP is proposed in section 5.5 to make the service-rate allocation more dynamic (and efficient) at the cell-time scale[#1]. The LSP allocates the time of the link server among the VPs that use it, under the direction of VPC.

It is worth pointing out that the proposed decomposition (designed mainly for illustrative purposes) is only one of many possible. Of course further decompositions can be formulated and integrated with the proposed hierarchical structure. Overlapping decompositions are reported to offer advantages over nonoverlapping decompositions [222], [223], [218], [33].

5.2 VPAM the higher level service-rate allocation and management scheme

VPAM is located at the highest level (level 4) of the proposed hierarchy. It is associated with a "slow" time scale in terms of hours or tens of minutes. It aims to supply optimal service-rate assignment, taking into account global network considerations. VPAM allows frequent dynamic reconfigurations of VPs across the network. Gerla et al [148] developed a M/M/1 queuing model (assuming independence between the queues) for VPAM, aimed at minimising total expected delays. Hui et al [157] formulate VPAM as a Non-Linear Programming model which minimises the total usage cost. In [161] Herzberg and Pitsillides propose an alternative model for VPAM which uses a network carrier viewpoint and maximises total network throughput. It modifies the model of Herzberg [224], developed originally for SDWSONET networks, and enhances the model developed by Herzberg in [146] for the BISDN. Of course other criteria of optimisation can be incorporated (as for example minimisation of delay, minimisation of total network loss, minimisation of congestion, maximisation of total network revenue etc...) to formulate a multiobjective optimisation problem [225], [226], that can also be hierarchically organised [218]. Also game theoretic concepts may be used to formulate optimisation problems that can deal with other issues, such as conflicting objectives, or introducing measures of fairness into the VP allocations.

In this thesis, we present an extension of the objective function used by Herzberg and Pitsillides [161] to provide the service-rate allocation problem with fairness among the VPs. The VPOSU supremal units supply to VPAM the statistical data required about

[#1] The case of other server disciplines will not be discussed further in this thesis, however for the case of a FIFO discipline, a modified VP ACFC can be used; see discussion in Chapter 3, page 46.

the expected Origin-Destination (OD) pair traffic loads based on fill knowledge of recent accepted and rejected calls.

5.2.1 Multiobiective VPAM model

Consider a single (virtual) network consisting of N nodes, representing ATM switches/Cross-Connect sites, and L transmission links connecting the nodes. For a given: network topology; expected OD traffic loads; and link capacities, we try to find an optimal VP service-rate assignment which maximises the total expected network throughput. We seek to provide for "fair" allocations of service-rate among all VPs (note that different VPs may have different performance objectives, which will have to be taken into account). The measure of fairness employed here[#1] is based on the concept of Pareto optimality from game theoretic [229], [230] considerations (also known as efficient, noninferior and nondominated optimality). Two main concepts of fairness are often discussed in the game theoretic literature: the concept of Pareto optimality [225]; and the concept of Nash optimality [23 1]. Pareto optimality generally can describe cooperative game situations. For the Pareto solution there does not exist another solution which is better for all users, i.e. the value of any objective function cannot be improved without degrading at least one of the other objective functions. Nash optimality generally describes noncooperative game situations. For the Nash solution, if all users use their Nash strategies then a single user deviating from his Nash strategy cannot improve his cost function. It thus safeguards against a single user deviating from the equilibrium strategy. In our problem the Pareto optimality concept is applicable, since the cooperation of the local units is guaranteed (i.e. they do not have their own local goals which can be in conflict with the global coordinator; if they did then the concept of Nash equilibrium or the concept of Stackelberg equilibrium [30] which allows the leader, or coordinator, to influence local behaviour for the overall benefit will be more suitable; see also [232] where Markovian Stackelberg strategies are discussed).

We define:

C_i^{link} - Available service-rate of link i, $i = 1,\dots,L$ for VP assignment.

N_p - Number of network unidirectional OD pairs,'indexed $j = 1,\dots,N_p \leq N(N-1)$.

P_j- Number of predetermined possible paths connecting OD pairj. Note that this formulation allows for multiple VPs between an OD pair.

$U_{j,p}$ - Service-rate assigned to OD pair j through path p, $j = 1,\dots,N_p$, $p = 1,\dots,P_j$.

[#1] Other measures of fairness have appeared in the literature, as for example [227] [228], however most are based on (fairly) ad hock concepts of fairness.

U_j - Service-rate assigned to OD pair j. Clearly $U_j = \sum_{p=1}^{P_j} U_{j,p}$, j = 1,...,$N_p$.

U^* is a Pareto optimal solution, $U^* = [U_1^*,...,U_{j_p}^*]$.

$D_j(U_j)$ - Expected throughput of OD pair j when it utilises service-rate assignment of size U,. Typically, $D_j(U_j)$ is a concave non-decreasing function that is monotonically increasing toward the asymptotic value $\overline{D}_j$, e.g. it can be obtained from equation (5.2) by allowing for an infinite allocation of service-rate, i.e.

$$D_j(\infty) = \int_0^\infty u f_j(u) du = \overline{D}_j .$$

$T_j^{\min}$ ($T_j^{\max}$) - Minimal (maximal) service-rate assigned by the user to OD pair j (e.g. $T_j^{\min}$ can be set to meet minimum performance objectives and $T_j^{\max}$ for fairness).

$\delta_{j,p}^i$ - Takes the value of 1 if path p of OD pair j uses link i, and 0 otherwise.

$F_j(U_j)$ - probability function for service-rate demand.

$f_j(u)$ - probability density function for service-rate demand.

Observe that the $U_{j,p}$ are the references to be provided to the lower levels. If the lower levels are the VPCs then $U_{j,p} \equiv C^d$, that is the desired references, as described in section 4.5.1, page 132. Note that if the references are required as ratios of the link service-rate (as for example in the modified algorithm 4.1), then an appropriate transformation can be carried out at this level.

The mathematical formulation for such a model is:

$$\max_U \left\{ D_1(U_1),...,D_{j_p}(U_{j_p}) \right\} \tag{5.1}$$

subject to the constraints

$$\sum_{j=1}^{N_p} \sum_{p=1}^{P_j} \delta^i_{j,p} U_{j,p} \le C_i^{link}, \qquad i = 1,...,L$$

$$T_j^{\min} \ge U_j = \sum_{p=1}^{P_j} U_{j,p} \ge T_j^{\max} \quad j = 1,...,N_p$$

$U_{j,p} \geq 0 \qquad j = 1,..,N_p \qquad p = 1,..,P_j.$

Note that $U^* \in \mathcal{U}$ the set of all Pareto optimal solutions if and only if $D_j(U_j) \leq D_j(U_j^*)$, $j = 1,..,N_p$ with strict inequality for at least one **j**.

To solve the above model, statistical characteristics of the functions $D_j(U_j)$, $j = 1,..,N_p$ should be known. We assume that each function $D_j(U_j)$ is derived from an appropriate probability fbnction $F_j(U_j)$ for service-rate demand and a corresponding probability density function f_j(u). By considering the throughput as a "fluid flow", the fbnction $D_j(U_j)$ can be obtained [161]:

$$D_j(U_j) = \int_0^{U_j} u f_j(u)du + U_j \int_{U_j}^{\infty} f_j(u)du = \int_0^{U_j} u f_j(u)du + U_j[1 - F_j(U_j)] \qquad (5.2)$$

The first term in equation *(5.2)* is the expected throughput for service-rate demand below the assigned service-rate of U_j, and the second term is for demand above the assigned service-rate of U_j.

Figure *5.1* presents a family of $D_j(U_j)$ functions derived from Normal Probability Functions having an average service-rate demand of *150* Mbit/s and different variance values σ. For illustrative purposes an assigned value of U_j = *200* Mbit/sec is shown by the dotted vertical line. Observe the expected throughput decrease as the variance of the service-rate demand function increases.

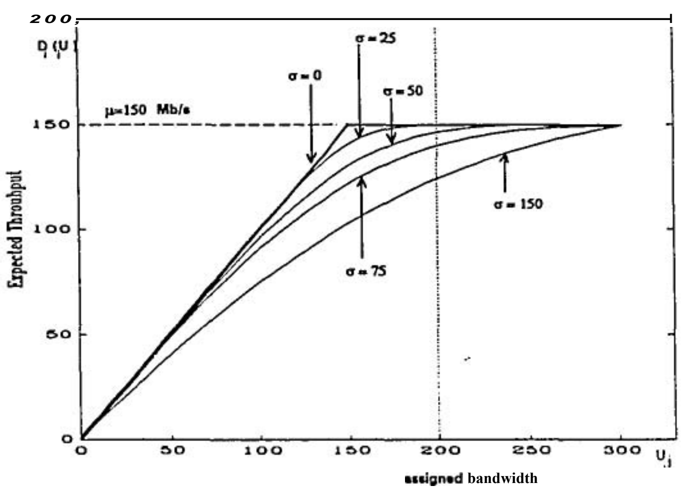


Figure *5.1*. Typical throughput functions $D_j(U_j)$.

The above problem belongs to the general class of multiobjective non-linear constrained optimisation problems. We want to find the set of the Pareto optimal solutions, and from this set select the optimum solution (or preferred solution), which is defined as any preferred Pareto optimal solution that belongs to the indifference band (the indifference band is defined [33] to be a subset of the Pareto optimal set where the improvement of one objective function is equivalent—in the mind of the decision maker—to the degradation of another).

solution approaches

Among the many methods that generate the set of feasible solutions [233], [33], [234], are:

- the weighted sum of the objective functions [226] (weighting method, parametric method). Herzberg [224], [146], [161] converts the multiobjective non linear problem to a single objective LP problem. He uses an equal weight of one for all functions (hence only generates one solution among the infinitely many). Note though that this approach offers all the computational advantages of the LP formulation. However the fairness issue cannot be addressed directly; but it can be enforced by the use of the constraints on the minimal (T_j^{min}) and maximal (T_j^{max}) service-rate allocations.
- the ε-constrained method [235] can be used to generate the set of noninferior solutions. It is worth noting that the Surrogate Worth Tradeoff (SWT) [236], [33] method can be used in conjunction with the ε-constrained method to generate the relative tradeoffs between the objective functions, and hence allow a quantitatively comparison of the objective (even for noncornrnensurate) functions.
- Hierarchical multiobjective analysis that exploits the general concept of decomposition-coordination, to provide for computational tractability, and possibly decentralisation of the computations [33], [234].

Once the set of Pareto optimal solutions is generated then the decision maker's preference can be used to choose the best-compromise solution from the generated set. Alternatively the Nash arbitration scheme [237][#1], or another scheme that gives a measure of fairness [238], can be used to select a unique operating point.

[#1] Note that the Nash arbitration scheme is different than the Nash equilibrium; the former refers to the cooperative situation, whilst the latter refers to noncooperative or competitive situations.

Example 5.1

Using a 3-node network, a comparison between the optimal bandwidth solution for two single objective formulations (sum and product of individual objective functions) and the Pareto optimum set is performed.

We consider a 3-node network ($N = 3$) with 2 OD pairs, both destined for node 3. Two VPs are established for each OD pair (N, $= 4$, $P_j = 2$, $j = 1,2$). The link capacities are set equal to 100 Mbit/sec ($C_i^{link} = 100$, $i = 1,2,3$).

The network topology is shown in figure 5.2, and the traffic characteristics (assuming a normally distributed probability function for service-rate demand) are tabulated below.

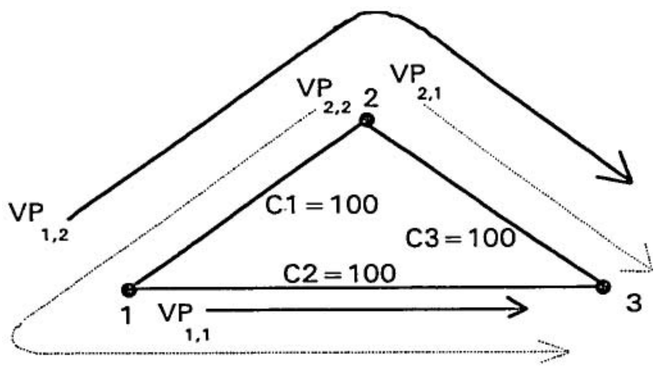


Figure 5.2. Three node network topology used for example 5.1.

		case i)	case ii)
Origin-destination pair	mean value of traffic demand	variance of traffic demand	variance of traffic demand
OD 1-3 (2 VPs) ($VP_{1,1}$ link 1-3 $VP_{1,2}$ link 1-2-3)	110	55	55
OD 2-3 (2 VPs) ($VP_{2,1}$ link 2-3 $VP_{2,2}$ link 2-1-3)	50	25	100

Table 5.1. Traffic characteristics for example 5.1.

Two cases are considered. The first depicts traffic with low variance for both OD pairs ("smooth" traffic) and in the second case OD pair 2-3 has a high variance ("bursty" traffic).

The mathematical formulation, as given earlier, is:

$$Max\{D_1(U_1), D_2(U_2)\}$$

such that.

$$U_{1,2} + U_{2,2} = C_1^{link} \qquad U_{1,1} + U_{2,2} = C_2^{link} \qquad U_{1,2} + U_{2,1} = C_3^{link}$$

$$U_1 = U_{1,1} + U_{1,2} \qquad U_2 = U_{2,1} + U_{2,2}$$

$$U_{i,j} \geq 0, \quad i = 1,2. \quad j = 1,2.$$

The ε-constraint method (see appendix 5.1) is used in this example to generate the set of Pareto optimal solutions. The Pareto optimum set is compared with two particular solutions obtained by using single objective methods, that of: the sum of the objective functions; and the product of the objective functions. It is shown, as expected, that the single objective formulation solutions are subsets of the Pareto optimum set.

For easy comparison the allocations for the sum and the product forms of the objective function for the two cases are tabulated below:

	sum of the objective functions, i.e. $Max\{D_1(U_1)+D_2(U_2)\}$		product of the objective functions, i.e. $Max\{D_1(U_1)\times D_2(U_2)\}$	
	service-rate allocations U_j	objective function value $D_j(U_j)$	service-rate allocations U_j	objective function value $D_j(U_j)$
case i	U_1=133 U_2=67	$D_1(U_1)$=99.2 $D_2(U_2)$=49.8	U_1=125 U_2=75	$D_1(U_1)$=96.4 $D_2(U_2)$=51.9
case ii	U_1=121 U_2=79	$D_1(U_1)$=95 $D_2(U_2)$=44	U_1=102 U_2=98	$D_1(U_1)$=85.9 $D_2(U_2)$=51

Table 5.2. Service-rate allocations for example 5.1 using the sum and the product forms of the objective function.

A comparison, as seen in table 5.2 above, of the single objective function formulations of the sum of the objective functions ($\sum_j D_j(U_j)$) and the product of the objective functions ($\prod_j D_j(U_j)$), shows that there are pronounced differences (about 20% for case ii) in the service-rate allocations of the two schemes, especially for the cases of more bursty traffic (indicated by a high variance in the demand function). The Pareto optimum set of the service-rate allocations and the objective functions (generated using the ε-constraint method, see appendix 5.1 for the details) are shown in figures 5.3 -5.6. As it can be seen from figures 5.3 and 5.4, both single objective formulations are particular solutions of the Pareto optimum set. The choice of the optimum solution based on either of the two single optimisation objectives, is not clear cut. However equipped with the Pareto optimum set one can select the "best" solution (e.g. using a method such as the SWT function), in the eyes of the decision maker.

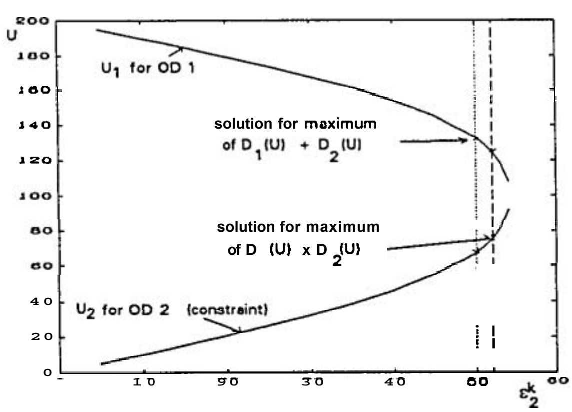


Figure 5.3. Pareto set of the optimum service-rate allocation for case i.

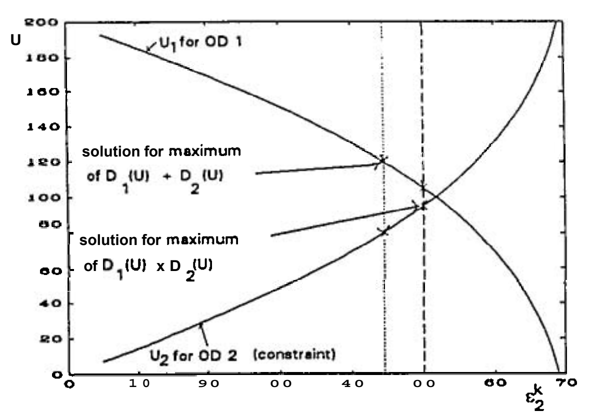


Figure 5.4. Pareto set of the optimum service-rate allocation for case ii.

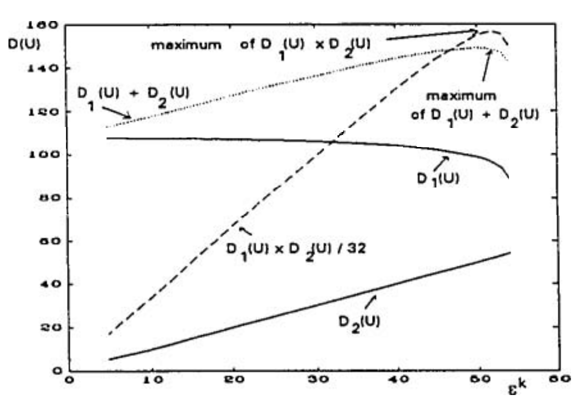


Figure 5.5. Functional values for case i.

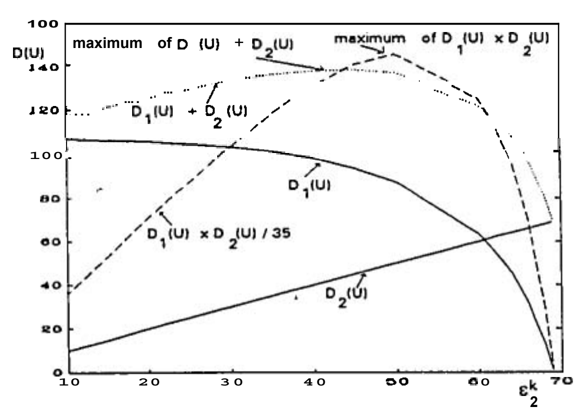


Figure 5.6. Functional values for case ii.

We will not pursue any of the proposed solution approaches further, but only recommend that a comparative study may be useful. The papers by Mason, Douligeris and colleagues are worthy of perusal: in [239] they show the Pareto optimality of the product of powers form for the optimal flow control; and in [240] Douligeris shows the formulation of a multiobjective problem, that of maximising the throughput (formulated using an M/M/1 model), for users sharing the network on a processor sharing basis, under a delay constraint. Hsia and Lazar [241] also maximise throughput with a constraint on the delay, for a FCFS exponential server, to solve the flow control problem based on the noncooperative concept of Nash.

5.3. The virtual path overall supremal unit (VPOSU) level

The VP overall supremal unit (VPOSU) is located at the third level of the proposed hierarchy. It is responsible for minimising the medium to long term interactions between the VPs (see figure 5.7).

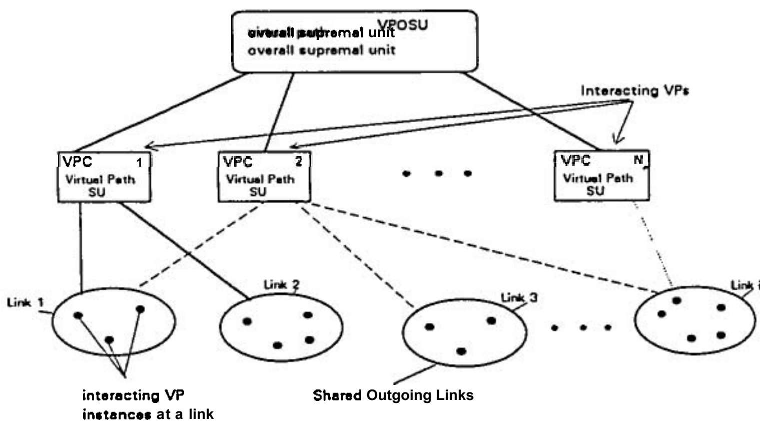


Figure 5.7. The interactions between VPs sharing a link and the proposed solution to neutralise them via a higher level supervisor, the VPOSU.

In this section, we firstly formulate the global constrained optimisation problem and then make use of decomposition-coordination techniques to decompose the system into a number of subsystems located at the origin nodes of the VPs. We use a natural physical decomposition along the VPs (horizontal decomposition), and aim at coordinating the decomposed systems in such a way as to minimise the medium to longer term interactions among the VPs (note that the derived algorithm belongs to the general class of decomposition-coordination algorithms that make use of the interaction prediction principle [27]).

Problem statement:

Assume that the overall state of all the VPs can be described by a state equation of the form

$$\dot{X} = f(X, U, \Lambda, t) \tag{5.3}$$

where

X is a vector of the state of the system of VPs, i.e. $X = [x_1, ..., x_{N_p}]$ where x_i is an n_i x 1 dimensional vector of the state of the ith VP, where $i = 1, ..., N_p$

U is a vector of service-rate allocations of the VPs, and

Λ is a vector of the cell input rates to the VPs.

We consider a cost function of the form

$$J_{VPOSU} = \int_{t_o}^{t_f} \{X^T(t)Q(t)X(t) + U^T(t)R(t)U(t) - w_{\Lambda}(t)\Lambda(t)\} dt. \tag{5.4}$$

The problem formulation then becomes

$$\underset{X,U,\Lambda}{Min}(J_{VPOSU}) \tag{5.5}$$

such that the following constraints are satisfied

i) $\dot{X} = f(X, U, \Lambda, t)$

ii) $\sum_{j=1}^{N_p} \delta_j^i U_j \le C_i^{link}, \quad i = 1, ..., L$

iii) $U_j \ge 0, \quad j = 1, ..., N_p.$

Decomposing into N_p subsystems (N_p is the number of the VPs; for notational simplicity we assume only one VP path for each O-D pair, i.e. $P_j = 1, \ \forall j$), we can describe each subsystem j by

$$\dot{x}_j = f_j(x_j, U_j, \lambda_j, t) + D_j z_j(t) \tag{5.6}$$

where

D_j is a matrix of the interconnections between the subsystems (i.e. between the VPs),

$z_j(t)=\sum_{i=1}^{N_p} D_{ji}x_i(t)$ is the interactions from other subsystems (assuming it has a linear structure, and that it is a function of the state of the overall system).

Recasting the problem in its decomposed form we have

$$\underset{X,U,\Lambda}{Min}(J_{VPOSU})=\underset{x_j,U_j,\lambda_j}{Min}\left(\sum_{j=1}^{N_p} J^j_{VPOSU}\right) \tag{5.7}$$

such that the modified constraints are satisfied

i) $\dot{x}_j=f_j(x_j,U_j,\lambda_j,t)+D_jz_j(t) \quad j=1,...,N_p$

ii) $\sum_{j=1}^{N_p}\delta^i_jU_j\le C_i^{link}, \qquad \mathrm{i}=1,...,\mathrm{L}$

iii) $U_j\ge 0, \qquad j=1,...,N_p$

where

$$J^j_{VPOSU}=\int_{t_o}^{t_f}\left\{x_j^T(t)Q_j(t)x_j(t)+U_j^T(t)R_j(t)U_j(t)-w_{\lambda,j}(t)\lambda_j(t)\right\}dt.$$

Forming the Lagrangian function, we can optimally solve for the minimum of the objective function, and at the same time force the interactions to be minimised

$$\begin{aligned}\mathcal{L}=\sum_{j=1}^{N_p}&\left\{x_j^T(t)Q_j(t)x_j(t)+U_j^T(t)R_j(t)U_j(t)-w_{\lambda,j}(t)\lambda_j(t)\right\}+\\&\nu_j^T(t)\left\{\dot{x}_j-f_j(x_j,U_j,\lambda_j,t)+D_jz_j(t)\right\}+\pi_j^T(t)\left\{z_j(t)-\sum_{i\ne j}D_ix_i\right\}\end{aligned} \tag{5.8}$$

where

ν_j is the costate variable, and

π_j is a Lagrange multiplier

Solving for the following necessary conditions for optimality [162], $\frac{\partial G}{\partial z_j}$ and $\frac{\partial L}{\partial \pi_j}$, we obtain the coordination update from iteration k to iteration k+1 as

$$\begin{bmatrix} \pi_j^{k+1} \\ z_j^{k+1} \end{bmatrix} = \begin{bmatrix} -D_j^T v_j^k \\ \sum_{i \neq j}^{N} D_{ij} x_i^{k+1} \end{bmatrix} \tag{5.9}$$

This task is accomplished at the **VPOSU.** At the local units local subproblems are solved

$$U_j(t) = f_j(x_j, \xi_j) \tag{5.10}$$

where

ξ_j is the coordination variable provided by the higher level to the local units.

For the special case of a linear (or linearised) VP system a local control law of the following form can be derived (using the conditions of optimality; as demonstrated in Chapter 4, appendix 4.III)

$$U_j(t) = -R_j^{-1} B_j^T K_j x_j(t) - R_j^{-1} B_j^T \xi_j(t) + U_j^{equil}(t) \tag{5.11}$$

where

A_j and B_j are obtained by linearising the control system around its equilibrium point (if not already in its linearised form).

K_j and ξ_j are obtained by letting the costate variable equal $v_j = K_j x_j + \xi_j$, thus transforming the TPBV[#1] problem to one in K, with a single point boundary, in addition to obtaining the solutions in a closed loop form.

K_j is obtained by solving the matrix Ricatti equation

$$\dot{K}_j = K_j A_j + A_j^T K_j - K_j B_j R_j^{-1} B_j R_j + Q_j, \qquad K(t_f) = 0$$

ξ_j the compensating vector, is obtained by solving

[#1] Defied in Chapter 4, page 150.

$$\dot{\xi}_j(t) = (-A_j^T + K_j B_j R_j^{-1} B_j)\xi(t) - K_j C_j z_j + \pi_j, \quad \xi(t_f) = 0.$$

Note that the first term of equation (5.11) is a local feedback part, the second term compensates for the interactions between the VPs (in an open loop fashion, since it is dependant on the initial conditions of $x_i(t_o)$, the local state) and the last term is the equilibrium service-rate. The information flow between the VPOSU and its local units is shown below

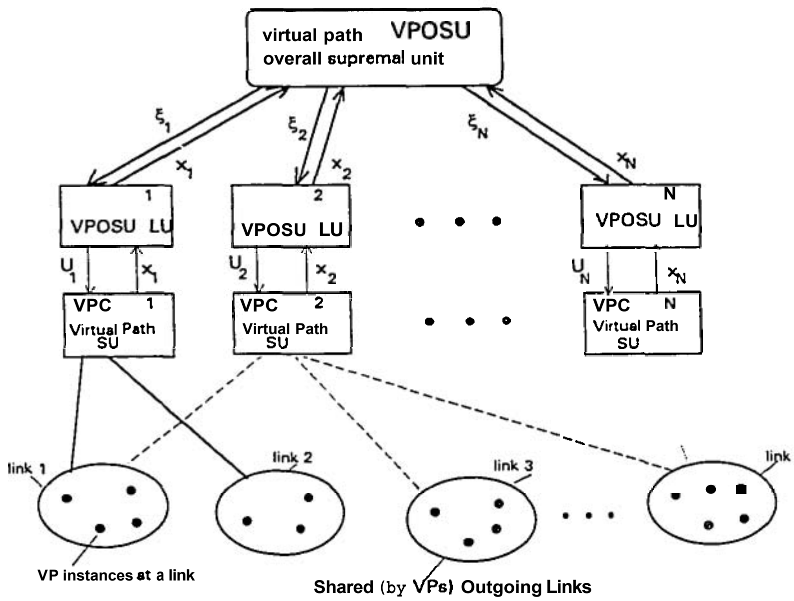


Figure 5.8. The information flow between the VPOSU and the VPC levels.

A single decomposition is treated above to minimise the medium to long term interactions between all the VPs in the network. Of course further decompositions are possible by clustering the VPs into geographical zones. Overlapping decompositions [33] may also prove useful.

5.4 The VP Control (VPC) level

The next level of the hierarchy (level 2) is for short to medium term time scales. The VPC is described in Chapter 4. Based on the state of the network, as seen at the network queues, and under the direction of the VPOSU it controls the service-rate allocated to VPs. The direction is provided by the setting of the references in the objective function of the VPC. Note that within the VPC level an additional vertical and horizontal decomposition, along the nodes spanned by the VP, is carried out so that a coordinated decentralised solution can be obtained (the LUs are located at the link controller). The VPC solution can deal with short to medium term traffic fluctuations. It cannot however deal with cell scale fluctuations. If the VPC output is provided as the reference to the link server (see figure 5.9), without any ability at the link server to modify it (e.g. by using feedback information from the instantaneous server occupancy), then an inefficient service-rate allocation would be the result.

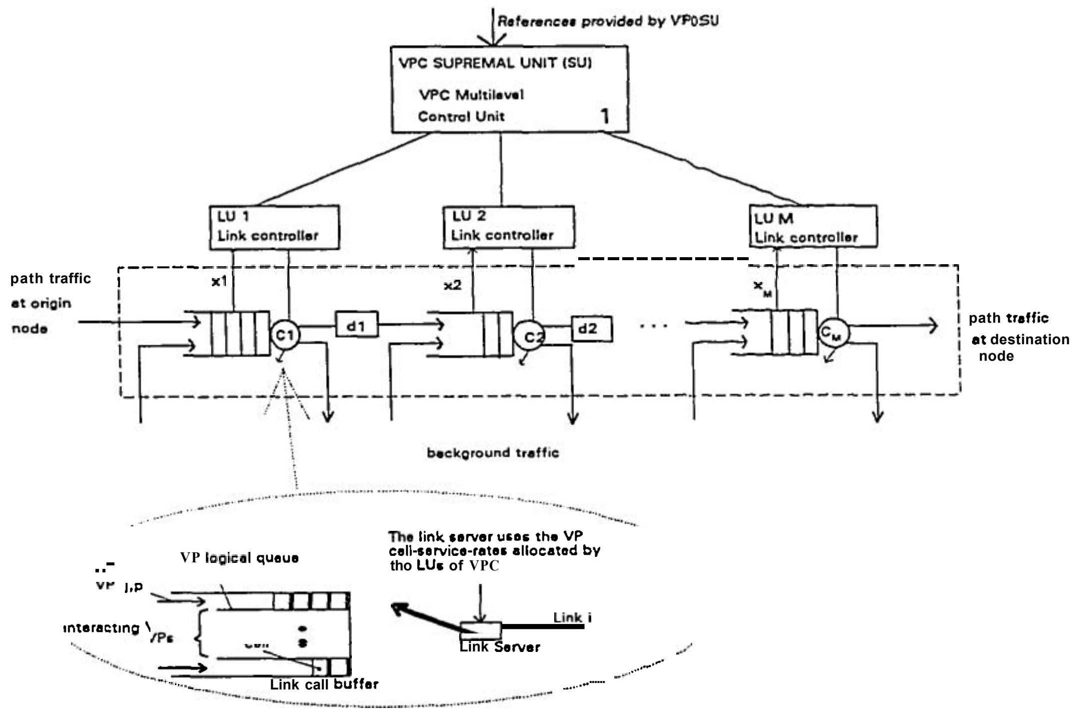


Figure 5.9. A single VP highlighting the link level.

It is not reasonable to remain idle, waiting for cells to amve from a particular VP while another VP cannot cope with its allocation and hence its queue builds up. Therefore, in order to take the cell scale fluctuations into account yet deal with higher level objectives, a more flexible link server protocol is required.

5.5. The lower level link service protocol (LSP)

At the lowest level of the proposed hierarchy we propose the LSP[#1]. It is aimed at reducing the short term congestion, in a fair and efficient way. Fairness is ensured by providing service rate which is not below the service-rate requested for each VP by VPC. Efficiency is provided by serving also during periods for which certain VP queues do not momentarily have any cells to serve ("moving on" policy). This is formulated in the next section.

[#1] As discussed in the introduction to this chapter, we assume a CQ server discipline (with a separate queue for each VP) at each node of the network used by the VPs.

Link Service Protocol (LSP)

At any ATM switching node we assume that the cell buffers, for each outgoing link, are organised with a logical queue for each VP as shown in figure 5.10. The link i serves all VPs j,p. Each VP j,p has service-rate $C^i_{j,p}$ assigned to it by the LUs of the VPC. $C^i_{j,p}$ is assigned as a ratio of the link server capacity C^{link}_i. We suggest, based on heuristics, a simple **LSP** discipline that uses a variable length cyclic cell server to serve the individual queues in accordance with the "move on" **LSP** described below.

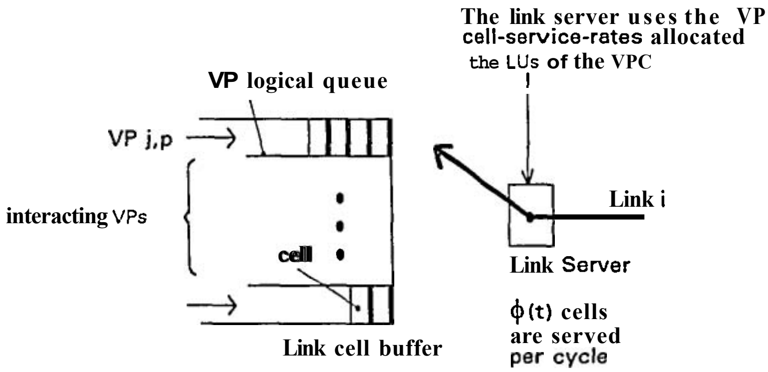


Figure 5.10. The link server outline.

At each link i the "move on" **LSP** discipline serves $u^i_{j,p}$ cells from each VP j,p using the following rule:

$$u^i_{j,p} = \begin{cases} x^i_{j,p} & \text{if } x^i_{j,p} < m^i_{j,p} \\ m^i_{j,p} & \textit{otherwise} \end{cases} \tag{5.34}$$

where

$x^i_{j,p}$ - Number of cells in the queue of VP j,p at the instance the link server accesses it.

$m^i_{j,p}$ - Maximum number of cells that can be served during a link server visit to queue j,p. It is derived fiom the optimal service-rate allocations $C^i_{j,p}$ (which are given as a fraction of the link rate C^{link}_i) allocated by $VP_{j,p}$ VPC, as follows.

$$m_{j,p} = round\left[\Phi^i \times C^i_{j,p}\right]$$

where

$round[x]$ rounds the number x to the nearest integer

Φ^i is a fixed length cycle. For example let $\Phi^i = 100$ cells in the cycle, and $C^i = [\frac{1}{4} \quad \frac{1}{4} \quad 0.52]$ where each entry represents the allocated (by VPC) service-rate ratio of one VP. (In this example **3 VPs** share the link i . Note that for this example the sum of the service-rate allocations exceeds 1). Then $m^i = [25 \quad 25 \quad 52]$.

Thus, this discipline provides a VP with a maximum service-rate of $C^i_{j,p}$ allocated to it by VPC. However if there are no cells to serve, at the instant the server accesses a particular VP queue, then the server does not remain "idle" (i.e. wait until cells for the VP in service arrive and the allocated quota served, or until its allocated cell-service time is exhausted). It "moves on" to the next VPs queue.

The number of cells served, at each link i , within a cycle is $\varphi^i(t) = \sum_{j=1}^{N_p} \sum_{p=1}^{P_j} \delta^i_{j,p} u^i_{j,p}$

where $\varphi^i(t) \le \sum_{j=1}^{N_p} \sum_{p=1}^{P_j} \delta^i_{j,p} m^i_{j,p}$ (using the example above the maximum $\varphi(t)$ in a cycle is equal to 102 cells). Note that the length of the cycle m^i may exceed the fixed length cycle Φ^i. For example, this can arise for a fairly heavily loaded link since the sum of allocated service-rates at any link i (i.e. $\sum_{j=1}^{N_p} \sum_{p=1}^{P_j} \delta^i_{j,p} C^i_{j,p}$) can exceed 1. Also note that for a fairly empty network the sum of allocated service-rates at any link i can be below 1. The VPC does not use hard constraints; it uses feedback from the queues to allocate the service-rate to the VPs, therefore the total allocated service-rate, at any link i , can be above or below the physical link service-rate. The degree of deviation from 1 will depend on the values of the weights on the VPC algorithm. Note that the LSP can accommodate service-rates that are above or below the value of 1; it merely changes the length of the cycle m^i.

The challenge is to keep the cycle length short (for efficiency), whilst allocating server time fairly (which requires a longer cycle due to the rounding to the nearest integer). This is a difficult problem, worthy of further investigation.

5.6 Summary

In this chapter we have presented an illustrative example of a hierarchically organised scheme for the control of service-rate. Four levels are formulated[#1] and their integration discussed. **A** short summary follows.

[#1] One of the levels (VPC) was formulated in chapter 4.

- Level 4 (multiobjective VPAM) is responsible for minimising interactions between the competing VPs on long term time scale. VPAM is located centrally and it operates at a slow time scale. This level uses long term predictions of the service-rate demands from the VPs to allocate service-rate to the VPs, taking into account global network considerations. The outputs of VPAM form the coordinating (reference) inputs to VPOSU (level 3).
- Level 3 (VPOSU) is responsible for minimising interactions between the competing VPs on the medium to long term time scale. VPOSU can also be located centrally, however for practical reasons (computational, spacial separation) a vertical decomposition along the VPs is used here, with one local unit located at each originating node of a VP. VPOSU uses feedback from the (aggregate) state of the VPs and the output of VPAM as its reference to allocate service-rate to the VPs. The outputs from the VPOSU local units form the coordinating (reference) inputs to VPC (level 2).
- Level 2 (VPC) is responsible for minimising interactions between the competing VPs on the short to medium term. Again, for practical reasons (computational, spacial separation) this level is vertically decomposed, featuring one local unit at each link along the VP. VPC uses feedback from the state of the local link queues and the output of VPOSU as its reference to allocate service-rate to the local queues of the VP. The outputs from the local units of the VPC form the coordinating (reference) inputs to LSP (level 1).
- Level 1 (LSP) is responsible for minimising interactions between the competing VPs on the short term (cell time-scale). LSP is situated at the links. It uses feedback from the (instantaneous) state of the link queues and the outputs of VPC as its reference to allocate service-rate to the VPs sharing a link in a fair and efficient way.

A schematic diagram of the proposed scheme can be seen in figure 5.11. Note that not all levels of the presented scheme are required or necessary to implement the service-rate allocation policy. Depending on computational complexity and time scales, different levels can be formulated and incorporated into the overall hierarchical structure. However, for illustrative purposes the proposed scheme is complete.

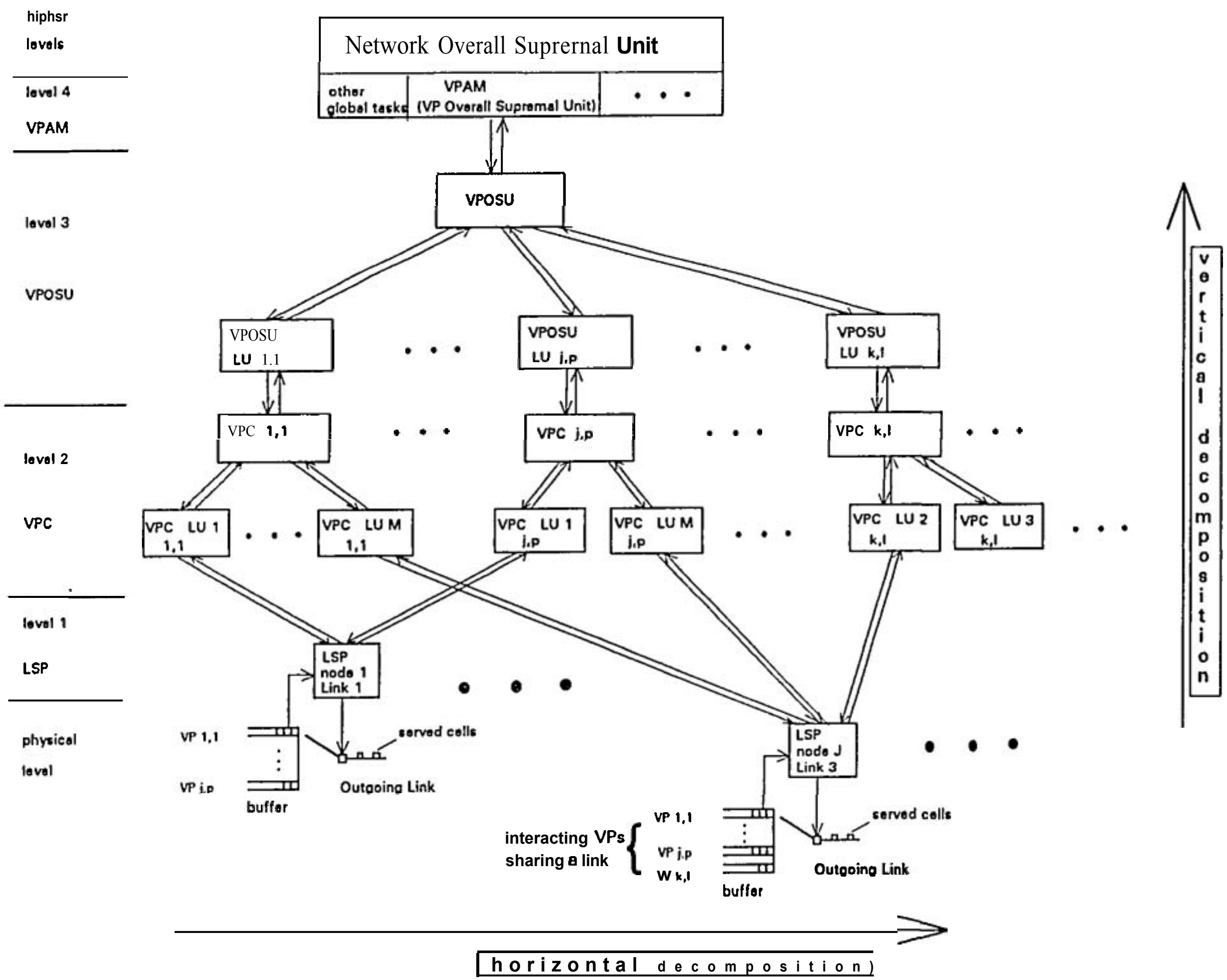


Figure 5.11. Schematic of the proposed hierarchical structure for the dynamic service-rate allocation.

Appendix 5.1: ε-constraint method

Using the ε-constraint method the multiobjective problem of example 5.1 is converted to a single objective problem, with the second objective function added to the constraints (if more objective functions were present these will also be included as constraints), i.e.

$$Max\{D_1(U)\}$$

such that

$$D_2(U) \geq \varepsilon_2^k, \quad k = 1,2,3,...$$

$$U_{1,2} + U_{2,2} = C_1^{link}$$

$$U_{1,1} + U_{2,2} = C_2^{link}$$

$$U_{1,2} + U_{2,1} = C_3^{link}$$

$$U_{i,j} \geq 0, \quad i = 1,2. \quad j = 1,2.$$

Where ε_2^k is varied, in this example, from 5 to about 70. Using the ε-constraint method the Pareto optimum set is generated; shown in figures 5.3 and 5.4. Additionally, the tradeoff function (figures 5.12 and 5.13), the solution in the decision space (figure 5.14), and the solution in the function space (figure 5.15) have been generated (shown for case i only; case ii is similar). Using the tradeoff function, the decision maker can select his indifference band, from which the preferred Pareto optimum is generated.

The preferred solution in the decision space, with an arbitrarily selected indifference band, is also shown (assuming that an interaction with a decision maker has taken place to select the indifference band and from it the preferred Pareto optimal solution).

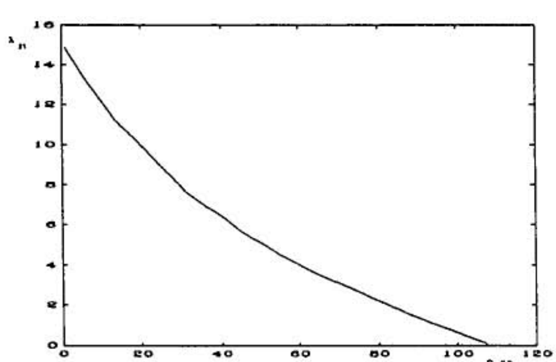


Figure 5.12. Tradeoff function A,, plotted against the fbnction D_1(U).

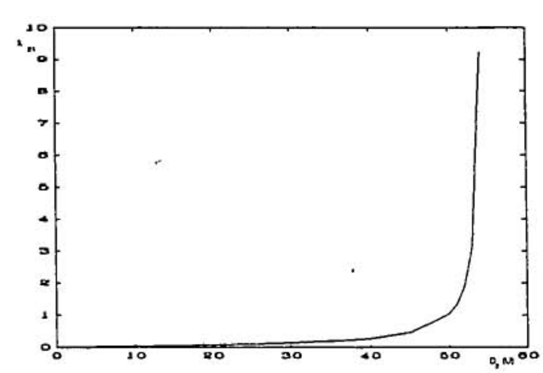


Figure 5.13. Tradeoff function A_{21} plotted against the function D_2(U).

Observe that only those $\lambda_{ij}^{k} > 0$ that correspond to the active constraints $D_j^k(\mathrm{U}) = \varepsilon_j^k$ are of interest since they indicate the marginal benefit of the objective function $D_i(\mathrm{U})$ due to a decrease of ε_j by one unit (they belong to the Pareto optimum solution). One way for the decision maker to define his indifference band [33] (a matter of "personal taste") is by using these **tradeoff** functions.

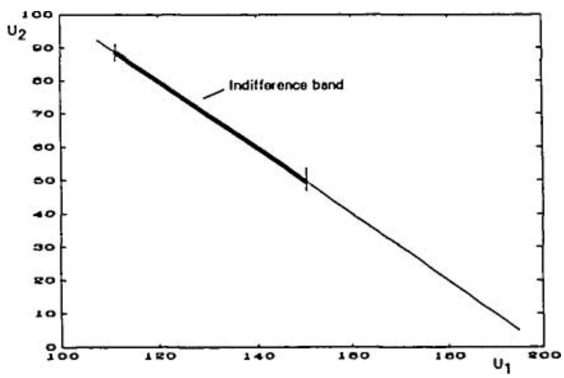


Figure 5.14. Pareto optimum solution in the decision space (U_1 versus U_2). An example of the chosen (in the decision space) indifference band is also shown.

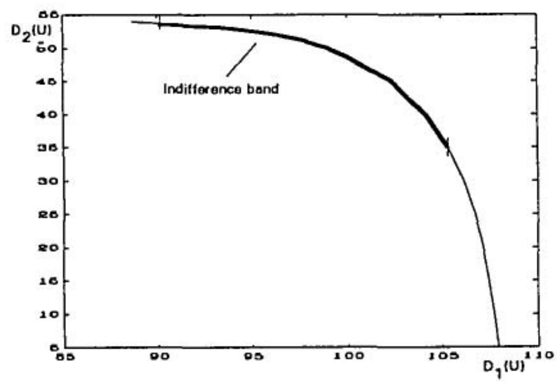


Figure 5.15. Pareto optimum solution in the functional space ($D_1(\mathrm{U})$ versus $D_2(\mathrm{U})$). **An** example of the chosen (in the functional space) indifference band is also shown.

CHAPTER 6

CONCLUSION

6.1 Summary

As noted in Chapter 1, CAC, flow control and sewice-rate control are three principal technical problems in BISDN. In Chapter 2, due to the complexity of the control problems in BISDN we proposed a hierarchically organised control structure. Additionally, due to the nonstationary behaviour of the network, different (dynamic) modelling techniques are proposed as being essential. Also for the efficient control of BISDN it is proposed that the (dynamic) tradeoff between service-rate, buffer-space, cell-delay and cell-loss cannot be ignored.

The primary focus of this thesis has been in the development of new dynamic CAC, flow control and service-rate control structures and methods, which can (possibly) form a part of an overall hierarchically organised BISDN control solution. In this thesis we offer new design and analysis techniques, which are applicable to both the stationary and the nonstationary behaviour of BISDN systems. The tradeoff between service-rate, buffer-space, cell-delay and cell-loss has been incorporated in the problem formulation.

In chapter 3 we have demonstrated that using the general tools of adaptive control, featuring on-line system identification techniques, robust, effective and efficient control can be implemented. It offers guaranteed QoS together with high utilisation of link capacity. In particular, we integrate the formulation of the CAC and flow control problems. Due to our novel control formulation the network efficiency can be maintained at high levels (theoretically at unity utilisation) yet the offered QoS can be regulated to defined target values. Since ACFC is implemented locally it is insensitive to propagation delays between nodes along its path. Using analysis and simulation its performance has been investigated. Bounds on the operating conditions are derived and using simulation we have shown the adaptability, robustness and increased efficiency of the scheme. Additionally, this scheme is dependant on only two broad traffic classifications: the uncontrollable and controllable groups of traffic. It is independent of the arrival process model, and it does not require any user declared parameters other than the peak rate for connections identified as uncontrollable.

In chapter 4 using multilevel optimal control theory, decentralised coordinated solutions are derived. These are based on a novel service-rate control scheme that uses feedback from the network queues. The derived solution is not sensitive to the propagation delays. Its performance has been investigated using simulation techniques.

Finally in chapter 5 the demonstration of a hierarchically structured overall solution is shown for the control of service-rate, in which the solutions at the various levels of the hierarchy are integrated. Four levels are formulated: multiobjective VPAM; VPOSU; VPC; and LSP.

6.2 Contributions of this work

In this section we list the primary contributions presented in this thesis in the order of their appearance.

- In Chapter 2
  - we present arguments to support the assertions that:
    - feedback control is feasible, despite the propagation delays, as long as it is constrained to lie within the control horizon (for example by appropriate vertical and horizontal decompositions);
    - for the overall control of BISDN, a hierarchically organised control structure featuring a vertical as well as a horizontal decomposition is essential;
    - that a number of different dynamic modelling techniques must be employed;
    - and that the dynamic tradeoff between service-rate, buffer-space, cell-delay and cell-loss must be exploited.
  - We also present a new view of hierarchical structure based on the system behaviour in both time and space.
- In chapter **3**
  - we make use of adaptive feedback and adaptive feedforward control methodologies, to solve the combined connection admission control and flow control problem.
  - we introduce a novel control concept, based on only two groups of traffic:
    - the controllable traffic, and

- the uncontrollable traffic.

Based on these two groups of traffic we achieve:

- network controllability by appropriate formulation of the CAC policy.

- we achieve high utilisation of resources and yet maintain the quality of service at prescribed (reference) values. Thus we achieve:
  - low cell-loss and (for the group of uncontrollable traffic) low cell-delay, and low cell-delay-variation (maintained at all times by regulating the QoS);
  - high efficiency (theoretically unity utilisation).
- We have proven analytically, using certain assumptions, that in the long term:
  - the regulator is stable and that it converges to zero regulation error;
  - the utilisation is equal to unity;
  - the controlled network is stable (as long as the uncontrollable traffic remains bounded, by the connection admission control scheme, below the link service-rate);
  - that guaranteed (worst case) bounds on the quality of service can be offered by the network to the user (worst case delay only applies to the uncontrollable traffic).
- Using simulation we have supported the above mentioned proofs and demonstrated the following features:
  - efficiency (a utilisation of 0.89 was demonstrated for a particular set of values for the reference and the expected overflow constant, a 48% improvement over a peak rate allocation scheme. Note that this can be improved further by setting the reference higher or the expected overflow constant at a lower value);
  - robustness to unforeseen traffic;
  - maintenance of the QoS close to the reference values;
  - ability to influence local behaviour.
- the only traffic descriptor required from the user is that of the peak rate of the uncontrollable traffic. (This can be compared with some of the schemes

proposed in the literature that require a full and accurate traffic descriptor in advance of the call connection to the network; even with an assumed accurate model these schemes mainly aim at cell-loss minimisation, without the simultaneous achievement of high network efficiency).

- only two simple traffic classifications (the controllable and uncontrollable groups) are required, irrespective of the detailed behaviour of the connections.

- In Chapter 4:
  - a novel scheme for the dynamic control of service-rate based on feedback from the network queues is proposed.
  - a unified dynamic fluid flow equation to describe the VP is presented.
  - two illustrative examples for the feedback control of service-rate at the VP level are formulated:
    - a nonlinear optimal multilevel implementation, that features a coordinated decentralised solution;
    - a single level implementation that turns out to be computationally complex. Therefore for the single level,
      - the costate equilibrium solution was also derived (using simulation we demonstrate that the costates do attain equilibrium and that this is close to the optimal solution).
  - a discussion of the implementation complexity of the derived optimal policies is included.
  - implementable solutions for the derived optimal policies are considered.
  - extensions are given for previous published works, on the optimal control of nonlinear systems, to the general case of large-scale nonlinear time-delayed systems.
  - we present simulative performance evaluation of the schemes. We have shown that:
    - a Control System Type Simulation (CSTS) closely resembles a Discrete Event Nonstationary Simulation (DENS);
    - the local behaviour of the LUs can be influenced via changes in the references and/or the weighting coefficients that appear in the objective function;

- the interaction between the nodes spanned by a VP becomes more pronounced as the service-rate rate available at a link becomes scarce (as for example during a high load situation);
- the effect of the interactions of the nodes along a VP can be minimised by the use of coordinated decentralised LUs;
- there is a tradeoff between buffer-space and service-rate.

- In Chapter 5:
  - we demonstrate the derivation of a particular solution for the control of service-rate using a hierarchical structure. In particular, we:
    - decomposed the system both vertically and horizontally to provide local coordinated decentralised solutions at the lower levels of the hierarchy and more global solutions operating at slower time scales at the higher levels.
      - For the case of VPAM, the highest level of the presented hierarchy, we extend published results to the case of a multiobjective formulation.
      - At the lowest levels of the hierarchy, we proposed LSP, a novel link server protocol derived from heuristic arguments.
      - At the intermediate level we use VPC, the dynamic service-rate control scheme described above.
      - In order to demonstrate the flexibility and adaptability of the hierarchical structure an additional level VPOSU is also formulated.
  - we extend VPAM to the case of multiple objectives, so that it can deal with issues of fairness, and multiple (possibly) conflicting objectives. We have shown through a simple example that
    - the solution set of the multiobjective formulation contains the particular solutions of the sum and product forms of single objective formulations.

6.3. Future extensions

The control structures and techniques proposed in the thesis appear very attractive for telecommunication problems, and they generate a large number of open questions. This thesis suggests these open questions are worth pursuing. We firstly present some general extensions and then specific extensions for the works presented in Chapters 3, 4 and 5.

General future extensions

- The use of the general tools of adaptive control and non-linear optimal multilevel control have been demonstrated in this thesis for the CAC, flow control and service-rate control problems. Extension to other telecommunication problems is strongly recommended.
- The transmission-rate (Chapter 3) and the service-rate (Chapter 4) control schemes have been investigated independently (their integration is briefly discussed in section 3.3.1.4). A formal investigation of the interactions and possible integration of these two schemes is recommended. Alternatively these two schemes can be seen as competing approaches for the control of BISDN and thus a comparative study may be useful.

  The provision of an overall coordination scheme that uses a combination of changing (by the higher level supervisor) the reference values as well as the weights in the objective functions of the local units. This combination may provide a more adaptable as well as flexible overall system. The changes on the weights can be made adaptively based on the overall state of the network. For example in cases of low network loading the weights on capacity may be relaxed, thus giving local units more freedom to deviate from the higher level directives (the reference values). Additionally, more selectivity may be introduced for different classes of traffic, depending on their tolerance of cell-delay and/or cell-loss. Some options for the updating of the weights that are worthy of further investigation are:

  - updating of the weights based on a heuristic (possibly semiautomatic, or even purely manual) updating of the coordination variables, say from knowledge gathered from statistical data.
  - introduction of a market based mechanism (see for example [242]), and letting the internal prices set the coordination variables (the weights in the objective function of the local units). Shadow prices may provide a basis for the price adjustment mechanism, however we do not recommend sole reliance on shadow prices [243].

  These options must be integrated with the overall scheme that updates the reference values.

- The relationship between time scales and decompositions has not been explored. Their formal study is recommended.
- A formal study of the timing requirements and information flows of the different levels of the hierarchy, as well as their cost is an open question well worthy of investigation.
- Due to the high speed of the network, dynamic routing is a higher level (than the cell level) function. It is of interest to investigate possible timescales and the interaction of routing with the other schemes described in this thesis.

Future extensions specifically for the CAC and flow control problem

- A comparison of the proposed (SISO-Single Input Single Output) implemenation with a multivariable (SIMO-Single Input Multiple Output) (for example option 1 proposed in section 3.2.2, page 35) or MIMO (Multiple Input Multiple Output) (for example an additional feedback signal from the queue length can prove useful) implementation is worthy of investigation.
- An adaptive multilevel [244] implementation. For example multiple local control units (one for each controllable source) can be used to individually control the transmission-rate of the controllable connections (for example option 1 proposed in section 3.2.2, page 35). These local units can be located at the customer premises.
- Appropriate jacketing software to prevent the regulator from reacting to unforeseen circumstances is an essential ingredient in any practical implementation. Use of fuzzy logic, or expert systems methodologies, to assist in this area, is worthy of investigation.

Future extensions specifically for the service-rate control problem

- The estimation of the ensemble averages may prove challenging. As already pointed out, Warfield et al [98] have presented a simple, recursive, real time estimator of the arrival rate. Its incorporation with the service-rate control algorithms is worthy of further investigation.
- An optimal multilevel implementation for the control of service-rate is proposed in chapter 4. It is of interest to investigate the use of adaptive (multilevel) control theory. In particular the use of continuous time methods (for example continuous time GPC [245], [246], or continuous time robust adaptive LQ control [247]), suitably extended to a multilevel implementation, may prove of value.

- The adaptive solution of the nonlinear optimal control problem, including the TPBV problem, may provide computationally attractive (implementable) solutions.
- Fuzzy optimal control [248] appears promising, and its incorporation into this scheme to allow for easier algorithmic solutions, is also worthy of investigation.
- The dynamic creation and deletion of VPs creating a reconfigurable network, see for example [249], and the interaction with the allocation of service-rate to VPs may be of interest.
- The solution of the combined service-rate and buffer-space control problem may provide for more efficient use of the resources.

<u>Future extensions specifically for the hierarchicallv organised control of cell-service-rate</u>

- The use of feedback from lower levels to the higher levels of the hierarchy (VPAM is currently open loop based) can (possibly) increase the overall robustness of the scheme; see for example [31].
- The extension of the multiobjective VPAM to include other objectives, as for example the maximisation of revenue, minimisation of total network call loss and the minimisation of total network delay, will offer better compromise solutions (say between the customers and the network operator).
- The derivation of an optimal Link Server Protocol, and its comparison with the proposed heuristic LSP strategy is worthy of further investigation.

6.4 Concluding remark

In this thesis adaptive control theory and multilevel optimal control theory have been successfully employed. Their suitability for BISDN control has been demonstrated through specific problem formulations. We offer an integrated structured approach to the control of BISDN that has the essential features of implementability, efficiency, effectiveness and robustness.

APPENDIX A

Most of the results and simulations presented in this thesis have been prepared by custom written software in MATLAB[#1]. In this appendix we list the three main suites of custom programs:

i) ACFC and ATM switch.

This simulation suite mainly consists of:

- A cell-level simulation of an ATM switch with multiple input connections, no internal blocking, and one output buffer of 100 cell places at the outgoing port.
- The traffic load which comprises a mix of voice, data and variable rate video, with random connections and disconnections of sources.
- The outgoing link buffer monitor (includes the calculation for the feedback signal and other QoS figures).
- The ACFC adaptive controller.

ii) Single level and Multilevel control of VPs.

This simulation suite mainly consists of

- CSTS of a VP spanning M nodes.
- Single level implementation, based on the costate solution.
- Multilevel (using quadratic and linear objective function formulation) implementation of the solution.

iii) Multiobjective formulation of VPAM.

This simulation suite mainly consists of

- ε-constraint optimisation

Note: the code can be made available by contacting the author at the School of Electrical Engineering, Swinburne University of Technology.

[#1] MATLAB is the trademark of The Mathworks Inc.

BIBLIOGRAPHY

[1] CCITT: Recommendation I.121, "Broadband Aspects of ISDN", Geneva, 1991.

[2] R. Handel, M. N. Huber, "Integrated Broadband Networks: An Introduction to ATM-Based Networks", Addison-Wesley, 1991.

[3] CCITT: Recommendation 1.211, "B-ISDN Service Aspects", Geneva, 1991.

[4] CCITT: Recommendation I.150, "B-ISDN ATM functional Characteristics", Geneva, 1991.

[5] Special issue: "Broadband network performance and congestion control", International Journal of Digital and analog communication systems, Vol. 3, No. 2, April-June 1990.

[6] Special issue: "Congestion control in ATM networks", IEEE Network, Vol. 6, No. 5, September 1992.

[7] Special issue: "Congestion control in high speed packet switched networks", IEEE Journal of Selected Areas in Communications, Vol. 9, No. 7, September 1991.

[8] Special issue: "Teletraffic Analysis of ATM systems", IEEE Journal of Selected Areas in Communications, Vol. 9, No. 3, April 1991.

[9] Cost 224, "Performance evaluation and design of multiservice networks", Edited by J. W. Roberts, EUR14152EN, Commission of the European Communities, 1992.

[10] J. Filipiak, "Real Time Network Management", Elsevier Science Publishers, North Holland, 1991.

[11] M. de Prycker, "Asynchronous Transfer Mode - Solution to Broadband ISDN" Ellis Horwood Ltd., 1991.

[12] M. Decina, V. Trecordi, "Traffic management and Congestion control in ATM networks", Guest editorial, IEEE Network, Vol. 6, No. 5, September 1992.

[13] CCITT: Recommendation 1.371 (temporary document), "Traffic Control and Congestion Control in B-ISDN", Geneva, June 1992.

[14] R. Jain, "Myths about congestion control in high-speed networks", ITC Specialist Seminar, New Jersey, 1990.

[15] P. Boyer, "A congestion control for the ATM", ITC Specialist Seminar, New Jersey, 1990.

[16] K. Sohraby, L. Fratta, I. Gopal, A. A. Lazar, "Congestion Control in High-Speed Packet Switched Networks", Guest Editorial, IEEE Journal of Selected Areas in Communications, Vol. 9, No. 7, September 1991.

[17] H. Van As, "Transient analysis of Markovian queuing systems and its application to congestion control modelling", IEEE Journal on Selected Areas in Communication, Vol. 4, No. 6, pp. 891-904, September 1986.

[18] D. Tipper, M. K. Sundareshan, "Numerical Methods for modelling Computer Networks Under Nonstationary Conditions", IEEE Journal on Selected Areas in Communication, Vol. 8, No. 9, pp. 1682-1695, December 1990.

[19] W. Lovegrove, J. L. Hammond, D. Tipper, "Simulation Methods for Studying Nonstationary Behaviour of Computer Networks", IEEE Journal on Selected Areas in Communication, Vol. 8, No. 9, pp. 1696-1708, December 1990.

[20] J-C. Bolot, A. Udaya Shankar, "Analysis of a fluid approximation to flow control dynamics", IEEE INFOCOM'92, pp. 2398-2407, Florence, May 1992.

[21] D. P. Heyman, A. Tabatabai, T. V. Lakshman, "Statistical Analysis and simulation study of video teleconferencing traffic in ATM networks", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 2, No. 1, pp. 49-58, March 1992.

[22] V. Anantharam, "The optimal buffer allocation problem", IEEE Transactions on Information theory, Vol. 35, July 1989.

[23] S. Low, P. Varaiya, "A simple theory of traffic and resource allocation in ATM", IEEE GLOBECOM'91, December 1991.

[24] J. Y. Hui, "Resource allocation for broadband networks", IEEE Journal on Selected Areas in Communication, Vol. 6, No. 9, pp. 1598-1608, December 1988.

[25] J. Filipiak, "M-architecture: A Structural Model of Traffic Management and Control in Broadband ISDNs,", IEEE Communication Magazine Vol. 27, No. 5, pp. 25-32, 1989.

[26] H. Gilbert, O. Aboul-Magd, V. Phung, "Developing a cohesive traffic management strategy for ATM networks", IEEE Communications Magazine, pp. 36-45, October 1991.

[27] M. D. Mesarovic, D. Macko, Y. Takahara, "Theory of Hierarchical, Multilevel, Systems", Academic Press, New York, 1970.

[28] M. G. Singh, A. Titli, "Systems: Decomposition, Optimisation and Control", Pergamon Press, Oxford, 1978.

[29] D. D. Siljak, M. K. Sundareshan, "A Multilevel Optimisation of Large-Scale Dynamic Systems", IEEE Transactions on Automatic Control, AC-21, 79-84, 1976.

[30] J. B. Cruz, Jr., "Leader-Follower Strategies for Multilevel systems", IEEE Transactions on Automatic Control, Vol. AC-23, No. 2, April 1978.

[31] W. Findeisen, F. N. Bailey, M. Brdys, K. Malinowski, P. Tatjewski, A. Wozniak, "Control and Coordination in Hierarchical Systems", John Wiley, 1980.

[32] M. S. Mahmoud, M. F. Hassan, M. G. Darwish, "Large-Scale Control Systems", Marcel Dekker, New York, 1985.

[33] Y. Y. Haimes, K. Tarvainen, T. Shima, J. Thadathil, "Hierarchical Multiobjective Analysis of Large-Scale Systems", Hemisphere Publishing Corporation, 1990.

[34] D. D. Siljak, "Large-Scale Dynamic Systems :Stability and Structure", North-Holland, NY, 1978.

[35] X-R. Cao, "A comparison of the dynamics of continuous and Discrete Event Systems", Proceedings of the IEEE, Vol. 77, No. 1, pp. 7-13, January 1989.

[36] A. Pitsillides, J. Lambert, B. Warfield, "A structure for the control of BISDN", 7th Australian Teletraffic Research Seminar, Mannum, Australia, November, 1992.

[37] A. Pitsillides, J. Lambert, B. Warfield, "A hierarchical control approach for Broadband-ISDN communication networks", 12th Triennial IFAC World Congress, Sydney, July 1993.

[38] G. Ramamurthy, R. S. Dighe, "A Multidimensional framework for congestion control in B-ISDN", IEEE Journal of Selected Areas in Communications, Vol. 9, No. 9, pp. 1440-1451, December 1991.

[39] R. Warfield, R. Harris, S. Mischnowicz, "Some problems of network management in a fast packet switching network", in Proceedings 3rd Fast Packet Workshop, Melbourne, May 1988.

[40] L. H. Campbell, H. J. Everitt, "A layered approach to network management and control", NOMS'92, September 1992.

[41] A. E. Eckberg, B. T. Doshi, R. Zoccolino, "Controlling congestion in B-ISDN/ATM: Issues and strategies", IEEE Communications Magazine, pp. 64-70, September 1991.

[42] K. H. Muralidhar, M. K. Sundareshan, "Adaptive routing and flow control in large communication networks: a hierarchical scheme for multiobjective optirnisation", IEEE INFOCOM '84, San Fransisco, 1984.

[43] R. Bolla, F. Davoli, A. Lombardo, S. Palazzo, D. Panno, "Adaptive bandwidth allocation by hierarchical control of multiple ATM traffic classes", IEEE INFOCOM'92, Florence, May 1992.

[44] Z. Ren, J. S. Meditch, "A two layer congestion control protocol for Broadband ISDN", IEEE INFOCOM'92, Florence, May 1992.

[45] D. A. Hughes, H. S. Bradlow, G. Anido, "Congestion control in an ATM network", ITC-13, Copenhagen, pp. 835-840, June 1991.

[46] CCITT: Draft Recommendation 1.311, "B-ISDN General Network Aspects", Melbourne, 1991.

[47] R. G. Addie, J. L. Burgin, S. L. Sutherland, "B-ISDN Protocol Architecture", IEEE GLOBECOM'88, Florida, USA, December 1988.

[48] J. Burgin, D. Dorman, "Broadband ISDN Resource Management: The Role of Virtual Paths", IEEE Communications Magazine, pp. 44-48, September 1991.

[49] T. Kanada, K. Sato, T. Tsuboi, "An ATM based transport network architecture", IEEE COMSOC workshop, Osaka, November 1987.

[50] S. Ohta, K. Sato, I. Tokizawa, "A dynamically controllable ATM transport network based on the virtual path concept", IEEE GLOBECOM'88, FT. Lauderdale, FL, pp. 1272-1276, November 1988.

[51] CCITT: Recommendation G114, "Mean one-way propagation delay", Blue Book, Fascicle 111.1, Geneva, 1989.

[52] A. Pitsillides, "Connection admission control (CAC)" Report LTR0492, Laboratory for Telecommunication Research, Swinburne University of Technology, July 1992.

[53] T. Murase, H. Suzuki, S. Sato, T. Takeuchi, "A call admission control scheme for ATM networks using a simple quality estimate", IEEE Journal of Selected Areas in Communications, Vol. 9, No. 9, pp. 1461-1470, December 1991.

[54] G. M. Woodruff, R. G. M. Rogers, P. S. Richards, "A congestion control framework for high speed integrated packetized transport", IEEE GLOBECOM'88, 7.1, 1988.

[55] K. J. Astrom, B. Wittenmark, "Adaptive control", Addison- Wesley, 1989.

[56] D.W. Clarke, "Introduction to self-tuning control", Implementation of self-tuning controllers, edited by K. Warwick, Peter Peregrinus, 1988.

[57] F. C. Schoute, "Simple decision rules for acceptance of mixed traffic streams", ITC12, 1988.

[58] P. Joos, W. Verbiest, "A statistical bandwidth allocation and usage monitoring algorithm for ATM networks", ICC'89, 13.5, 1989.

[59] K. Mase, S. Shioda, "Real-time network management for ATM networks", ITC-13, pp. 133-140, 1991.

[60] S. Akhtar, "Congestion control in fast packet switching network", Master Thesis, Washington University, 1987.

[61] G. Gallassi, G. Rigolio, L. Fratta, "ATM: Bandwidth assignment and bandwidth enforcement policies", IEEE GLOBECOM'89, 49.6, pp. 1788-1793, 1989.

[62] C. Rasmussen, J. H. Sorenson, K. S. Kvols, S. B. Jacobsen, "Source independent call acceptance in ATM networks", IEEE Journal of Selected Areas in Communications, Vol. 9, No. 3, pp. 351-358, April 1991.

[63] R. Guerin, H. Ahmadi, M. Naghshineh, "Equivalent capacity and its application to bandwidth allocation in high-speed networks", IEEE Journal of Selected Areas in Communications, Vol. 9, No. 7, September 1991.

[64] W. P. Yeh, J. L. Hammond, D. Tipper, "Design and evaluation of a method for admission control for broadband packet networks", submitted to Computer Networks and ISDN, also CCCS Clemson University working paper, 1992.

[65] H. Saito, K. Shiomoto, "Dynamic call admission control in ATM networks", IEEE Journal of Selected Areas in Communications, Vol. 9, No. 7, pp. 982-989, September 1991.

[66] H. Saito, "Call admission control in an ATM network using upper bound of cell loss probability", IEEE Transactions on Communications, Vol. COM-40, No. 9, pp. 1512-1521, September 1992.

[67] K. Tutufor, "On admission control and policing in an ATM based network", 7th ITC Specialist Seminar, New Jersey, October 1990.

[68] M. Butto, E. Cavallero, A. Tonietti, "Effectiveness of the "leaky bucket" policing mechanism in ATM networks", IEEE Journal of Selected Areas in Communications, Vol. 9, No. 3, pp. 335-342, April 1991.

[69] B. Lague, C. Rosenberg, F. Guillemin, "A generalisation of some policing mechanisms", IEEE INFOCOM'92, pp. 767-775, Florence, May 1992.

[70] F. Guillernin, P. Boyer, A. Dupuis, L. Romoeuf, "Peak Rate Enforcement in ATM", IEEE INFOCOM'92, pp. 753-758, Florence, May 1992.

[71] L. Kleinrock, M. Gerla, "Flow Control: A comparative survey", IEEE Transactions on Communications, Vol. COM-28, No. 4, pp. 553-574, April 1980.

[72] D. Tipper, "Adaptive Routing, Flow Control, and Buffer Management in Computer Communication Networks, Ph.D. Thesis, The University of Arizona, 1988.

[73] D. Mitra, I. Mitrani, K. *G.* Ramakrishnana, J. B. Seery, A. Weiss, "A unified set of proposals for control and design of high speed data networks", 7th ITC Specialist Seminar, Moristown, October 1990.

[74] D. Mitra, "Asymptotically optimal design of congestion control for high speed data networks", IEEE Transactions on Communications, Vol. COM-40, No. 2, pp. 301-311, February 1992.

[75] D. Mitra, J. B. Seery, "Dynamic adaptive windows for high speed networks with multiple paths and propagation delays", IEEE INFOCOM' 91, 1991.

[76] K. K. Ramakrishnan, R. Jain, "A binary feedback scheme for congestion avoidance in computer networks with a connectionless network layer", ACM SIGCOMM, 1988.

[77] S. Pingali, D. Tipper, J. Hammond, "The performance of adaptive window flow controls in a dynamic load environment", IEEE INFOCOM '90, San Fransisco, June, 1990.

[78] A. Mukherjee, L. H. Landweber, T. Faber, "Dynamic time windows and generalised virtual clock: combined closed-looplopen-loop congestion control", IEEE INFOCOM' 92, Florence, May 1992.

[79] N. Yin, M. G. Hluchyj, "A dynamic rate control mechanism for source coded traffic in a fast packet network", IEEE Journal of Selected Areas in Communications, Vol. 9, No. 7, pp. 1003-1012, September 1991.

[80] C. Agnew, "Dynamic modelling and control of congestion-prone systems", Operations Research, Vol. 24, No. 3, pp. 400-419, 1976.

[81] S. Keshav, "A control theoretic approach to flow control", ACM SIGCOMM'91, Zurich, Switzerland, September 1991.

[82] D. W. Clarke, "Introduction to self-tuning control", in Implementation of self-tuning controllers, edited by K. Warwick, Peter Peregrinus, England, 1988.

[83] R. E. Kalman, "Design of a self-optimising control system" Transactions ASME, 80, pp. 468-478, 1958.

[84] E. Mishkin, L. Brown, "Adaptive Control Systems", McGraw-Hill, 1961.

[85] V. Peterka, "Adaptive Digital Regulation of Noisy Systems" 2nd IFAC Symposium on Identification and Process Parameter Estimation, Prague, paper 6.2, 1970.

[86] K. J. Astrom, B. Wittenrnark, "On Self Tuning regulators" Automatica, Vol. 9, pp. 185-199, 1973.

[87] D. W. Clarke, P. J. Gawthrop, "Self-Tuning Controller" Proceedings IEE Pt. D, 122, pp. 929-934, 1975.

[88] G. C. Goodwin, K. S. Sin, "Adaptive Filtering, prediction, and Control", Prentice Hall, Englewood Cliffs, NJ, 1984.

[89] B. E. Ydstie, A. H. Kemma, L. K. Liu, "Multistep predictive control and adaptation", in Implementation of self-tuning controllers, edited by K. Warwick, Peter Peregrinus, England, 1988.

[90] J. L. Burgin, C. J. O'Neil, "Traffic Control and Resource Management for the Broadband-ISDN", Internal Report 8080, Telecom Australia Research Laboratories, December 1991.

[91] P. B. Luh, Y-C. Ho, R. Muralidharan, "Load adaptive pricing: An emerging tool for electric utilities", IEEE Transactions on Automatic Control, Vol. AC-27, No. 2, April 1982.

[92] L. Ljung, T. Soderstrom, "Theory and Practice of Recursive Identification", MIT press, 1983.

[93] P. S. Tuffs, D. W. Clarke, "Self-tuning control of offset: a unified approach", Proceedings IEE, Vol. 132, Pt.D, pp. 100-110, 1985.

[94] R. Addie, M. Zukerman, "An asymptotically accurate method for performance evaluation of semi-Markov queues with large state space", 6th Australian Teletraffic Research Seminar, Wollongong, Australia, November 1991.

[95] R. Addie, M. Zukerman, "An approximation for performance evaluation of stationary single server queues" to appear in the IEEE Transactions on Communications, 1993. (Also in Proceedings INFOCOM'93, San Fransisco, April 1993).

[96] R. Addie, M. Zukerman, "Performance evaluation of Gaussian queue with a finite buffer", 7th Australian Teletraffic Research Seminar, Mannum, Australia, November 1992.

[97] R. Addie, M. Zukerman, "A Gaussian traffic model for a B-ISDN statistical multiplexer," IEEE GLOBECOM'92, Florida, USA, December 1992.

[98] B. Warfield, S. Chan, " Real-time traffic estimation in B-ISDN", 7th Australian Teletraffic Research Seminar, Mannum, Australia, pp. 141-150, November 1992.

[99] H. Yamada, S. Sumita, "A traffic measurement method and its application for cell loss probability estimation in ATM networks", IEEE Journal of Selected Areas in Communications, Vol. 9, No. 3, pp. 315-323, April 1991.

[100] H. Ahmadi, P. Kermani, "Real time network load estimation in packet switched networks", IFIP, 1991.

[101] R. D. Strum, D. E. Kirk, "First principles of discrete systems and Digital Signal Processing", Addison-Wesley, 1988.

[102] T. Soderstrom, P. Stoica, "System Identification", Prentice Hall (UK), 1989.

[103] T. R. Fortescue, L. S. Kershenbaum, B. E. Ydstie, "Implementation of self-tuning regulators with variable forgetting factors", Automatica, Vol. 17, pp. 831-835, 1981.

[104] L. Chisci, G. Zappa, "A systolic architecture for iterative LQ optirnisation", Automatica, Vol. 27, No. 5, pp. 799-810, 1991.

[105] K.J.Astrom, "Theory and applications of adaptive control-A survey" Automatica, vol. 19, pp. 471-486, 1983.

[106] M. Luoni et al, "Source models and applications for video", 2nd RACE workshop on traffic performance aspects in IBCN, Aveiro, Portugal, January, 1992.

[107] B. Maglaris, D. Anastassiou, P. Sen, G. Karlsson, J. Robbins, "Performance models of statistical multiplexing in packet video communications", IEEE Transactions on Communications, Vol. 36, No. 7, pp. 834-844, July 1988.

[108] R. Grunenfelder, J. P. Cosmas, S. Manthorpe, A. Odinrna-Okafor, "Characterisation of video codecs as Autoregressive Moving Average Processes and related queueing system performance", IEEE Journal of Selected Areas in Communications, Vol. 9, No. 3, pp. 284-293, Vol. 9, No. 3, April 1991.

[109] ISO/IEC, "Coded representation of picture, audio and multimedialhypermedia information", JTC 1/SC29, December 1991.

[110] M. Zukerman, M. Leditschke, M. Biggar, "Traffic studies of variable bit rate conferencing video", Packet Video'91, Kyoto, Japan, 1991.

[111] J. Appleton, "Modelling a connection acceptance strategy for asynchronous transfer mode", 7th ITC Specialist Seminar, paper 5.1, New Jersey, October 1990.

[112] T. R. Griffiths, "Analysis of connection acceptance strategy for ATM networks", IEEE GLOBECOM'90, San Diego, 1990.

[113] H. Esaki, K. Iwamura, T. Kodama, "Simple and effective admission control method for an ATM network", IEEE GLOBECOM'90, 300.4, Vol. 1, pp 28-33, December 1990.

[114] C. J. Weinstein, "Fractional speech loss and talker activity model for TASI and for packet switched speech", IEEE Transactions on Communications, Vol. 26, No. 8, pp. 1253-1257, August 1978.

[115] H. Suzuki, T. Murase, S. Sato, T. Takeuchi, "A simple and burst-variation independent measure of service quality for ATM traffic control", 7th ITC Specialist Seminar, 13.1, New Jersey, October 1990.

[116] K. Warwick (editor), "Implementation of self-tuning controllers", Peter Peregrinus, England, 1988.

[117] C. J. Hams, S. A. Billings (editors), "Self-tuning and adaptive control: theory and applications", Peter Peregrinus, 1985.

[118] K. J. Astrom, B. Wittenmark, "Computer controlled systems: theory and design", Prentice Hall , Englewood Cliffs, NJ, 1990.

[119] K. J. Astrom, U. Borisson, L. Ljung, B. Wittenmark, "Theory and applications of self-tuning regulators" Automatica, Vol. 13, pp. 457-476, 1977.

[120] Special issue: "Adaptive control", Automatica, Vol. 20, No. 5, 1984.

[121] 0 . J. M. Smith, "A controller to overcome dead-time", ISA. journal, ***6,*** pp. 28-33, 1959.

[122] K. J. Astrom, "Introduction to stochastic control theory", Academic Press, 1970.

[123] G. Monga, E. Mosca, "MUSMAR: multistep multivariable regulators base on multistep cost functionals", in Advances in Control, edited by D. G. Lainiotis and N. S. Tzannes, pp. 334-341, D. Reidel, Dordercht, 1979.

[124] V. Peterka, "Predictor-based self tuning control", Automatica, Vol. 20, pp. 39-50, 1984.

[125] D. W. Clarke, C. Mohtadi, P. S. Tuffs, "Generalised predictive control", Automatica, Vol. 23, (2 parts) pp. 137-160, 1987.

[126] T. T. C. Tsang, D. W. Clarke, ""Generalised predictive control with input constraints", Proceedings IEE Pt. D, Vol. 135, pp. 451-460, 1988.

[127] E. Mosca, J. Zhang, "Stable redesign of predictive control", Automatica, Vol. 28, No. 6, pp. 1229-1233, 1992.

[128] C. Mohtadi, "Numerical algorithms in self-tuning control", in Implementation of self-tuning controllers, edited by K. Warwick, Peter Peregrinus, England, 1988.

[129] G. Casalino, F. Davoli, R. Minciardi, G. Zappa, "On implicit modelling theory: Basic concepts and application to adaptive control", Automatica, Vol. 23, No. 2, pp. 189-201, 1987.

[130] G. Casalino, R. Minciardi "Further results on implicit models with application to LQ adaptive optimisation", Automatica, Vol. 28, No. 2, pp. 407-410, 1992.

[131] D. W. Clarke, P. J. Gawthrop, "Self-Tuning Control" Proceedings IEE Pt. D, Vol. 126, No. 6, pp. 633-640, 1979.

[132] D. W. Clarke, P. J. Gawthrop, "Implementation and application of microprocessor-based self-tuners" Automatica, Vol. 17, pp. 233-244, 1981.

[133] D. W. Clarke, "Introduction to self-tuning controllers", in Self-tuning and adaptive control: theory and applications, edited by C. J. Harris and S. A. Billings, Peter Peregrinus, 1985.

[134] K. J. Astrom, P. Eykhoff, "System Identification-A survey", Automatica, Vol. 7, pp. 123-162, 1971.

[135] L. Ljung, "System Identification: Theory for the user", Prentice Hall, Englewood Cliffs, NJ, 1987.

[136] S. L. Shah, W. R. Cluett, "RLS based estimation schemes for self-tuning control", in Implementation of self-tuning controllers, edited by K. Warwick, Peter Peregrinus, England, 1988.

[137] K. S. Narendra, A. M. Annaswamy, "Stable Adaptive Systems", Prentice Hall, Englewood Cliffs, NJ, 1989.

[138] G. C. Goodwin, D. J. Hill, M. Palaniswarni, "A perspective of convergence of adaptive control algorithm", Automatica, Vol. 20, No. 5, pp. 519-531, 1984.

[139] A. Pitsillides, R. Luppa, "Practical experiences with the Asea Novatune" 3rd Conference on Control Engineering, Sydney, May 1986.

[140] D. Peel, M. Tham, "Self-tuning control: case studies" in Implementation of self-tuning controllers, edited by K. Warwick, Peter Peregrinus, England, 1988.

[141] A. J. Moms, Y. Nazer, R. K. Wood, "Single and multivariable application of self-tuning controllers", in Self-tuning and adaptive control: theory and applications, edited by C. J. Hams and S. A. Billings, Peter Peregrinus, 1985.

[142] D. W. Clarke, "Implementation of self-tuning controllers", in Self-tuning and adaptive control: theory and applications, edited by C. J. Hams and S. A. Billings, Peter Peregrinus, 1985.

[143] B. Wittenmark, K. J. Astrom, "Practical issues in the implementation of self-tuning control", Automatica, Vol. 20, No. 5, pp. 595-605, 1984.

[144] G. C. Goodwin, P. J. Ramadge, P. E. Caines, "Discrete time multi-variable adaptive control", IEEE Transactions on Automatic Control, Vol AC-25, pp. 449-456, 1980.

[145] L. Kleinrock, "Communication Nets: Stochastic message flow and delay", McGraw-Hill, New York, 1964.

[146] M.Herzberg, "A linear programming model for virtual path allocation and management in B-ISDN", ABSSS' 92, Melbourne, Australia, July 1992.

[147] S. P. Evans, "Algorithms and principles for optimal call admission and capacity management in B-ISDN", Report TRC 16/90, Teletraffic Research Centre, Adelaide 1990.

[148] M. Gerla, J. A. S. Monteiro, R. Pazos, "Topology Design and Bandwidth Allocation in ATM nets", IEEE Journal on Selected Areas in Communication, Vol. 7, No. 8, pp. 1253-1262, October 1989.

[149] R. J. Harris, M. H. Rossiter, "A Multi-class, Multi-Commodity Flow Model of a Fast Packet Switch", Second Australian Teletraffic Research Seminar, Adelaide, November 1987.

[150] Z. Dziong, J. Choquette, K. Liao, L. Mason, "Admission control and routing in ATM networks", ITC specialist seminar, Paper No. 15.3, Adelaide 1989.

[151] F. P. Kelly, "Routing and capacity allocation in networks with trunk reservation", Mathematics of Operations research, Vol. 15, No. 4, pp. 771-793, 1990.

[152] P. Boyer, J. Louvion, D. Tranchier, "Intelligent multiplexing in an ATM network", 3rd IEEE COMSOC Multimedia '90, Bordeaux, France, November 1990.

[153] J. M. Hyman, A. Lazar, G. Pacifici, "Real-time scheduling with quality of service constraints", IEEE Journal on Selected Areas in Communication, Vol. 9, No. 7, pp. 1052-1063, September 1991.

[154] Y. Takagi, S. Hino, T. Takahashi, "Priority assignment control of ATM line buffers with multiple QoS classes", IEEE Journal on Selected Areas in Communication, Vol. 9, No. 7, pp. 1078-1092, September 1991.

[155] R. Dighe, C. J. May, G. Ramamurthy, "Congestion avoidance strategies in broadband packet networks", IEEE INFOCOM'91, 1991.

[156] J. Y. Hui, "Layered required bandwidth for heterogeneous traffic", IEEE INFOCOM'92, Florence, May 1992.

[157] J. Y. Hui, M. B. Gursoy, N. Moayeri, R. D. Yates, "A layered broadband switching architecture with physical or virtual path configurations", IEEE Journal of Selected Areas in Communications, Vol. 9, No. 9, pp. 1416-1426, December 1991.

[158] S. Ohta, K. Sato, "Dynamic bandwidth control of the virtual path in an Asynchronous Transfer Mode network", IEEE Transactions on Communications, Vol. 40, No. 7, July 1992.

[159] K. Sato, S. Ohta, I. Tokizawa, "Broad-Band ATM Network Architecture Based on Virtual Paths", IEEE Transactions on Communications, Vol. 38, No. 8, August 1990.

[160] J. Burgin, "Dynamic Capacity Management in the BISDN", International Journal of Digital and Analog Communications Systems, Vol. 3, pp. 161-165, 1990.

[161] M. Herzberg, A. Pitsillides, "A hierarchical approach for the bandwidth allocation, management and control in B-ISDN", IEEE International Conference on Communications, ICC'93, Geneva, May 1993.

[162] L. S. Pontryagin, V. G. Boltyanski, R. V. Garnkrelidze, F. Mishchenko, "The mathematical theory of optimal processes", Wiley, New York, 1962.

[163] A. Segall, "The modelling of adaptive routing in data communication networks", IEEE Transactions on Communications, Vol. 25, No. 1, pp. 85-95, January 1977.

[164] F. H. Moss, A. Segall, "An optimal control approach to dynamic routing in networks", IEEE Transactions on Automatic Control, Vol. AC-27, No. 2, April 1982.

[165] J. Filipiak, "Optimal control of store-and-forward networks", Optimal Control Applications and Methods, Vol. 3, pp. 155-176, 1982.

[166] J. Filipiak, "Modelling and Control of Dynamic Flows in Communication Networks", New York: Springer-Verlag, 1988.

[167] D. Tipper, M. K. Sundareshan, "Optimal threshold routing of virtual circuits in data networks", ORSA/TIMS Joint National Meeting, 1988.

[168] D. Tipper, M. K. Sundareshan, "An optimal control approach to decentralised dynamic virtual circuit routing in computer networks", IEEE INFOCOM '90, pp. 926-933, 1990.

[169] A. A. Economides, P. A. Ioannou, J. A. Silvester, "Adaptive routing and congestion control for window flow controlled virtual circuit networks", 27th Annual Allerton Conference on Communication, Control, and Computing, pp. 849-857 September 1989.

[170] A. A. Economides, P. A. Ioannou, J. A. Silvester, "Dynamic routing and congestion control for multi-class virtual circuit networks", Technical report # 90-06-01, Electrical Engineering, University of Southern California, June 1990.

[171] G. I. Stassinopoulos, P. Kostantopoulos, "Optimal'congestion control in single destination networks", IEEE Transactions on Communications, COM-33, pp. 792-800, 1985.

[172] A. Iftar, E. J. Davison, "A decentralised control strategy with discrete-time updating for dynamic routing", IFAC 11th Triennial World Congress, pp. 111-116, August 1990.

[173] P. E. Sarachik, U. Ozguner, "On decentralised dynamic routing for congested traffic networks", IEEE Transactions on Automatic Control, AC-27, pp. 1233-1238, 1982.

[174] G. Casalino, F. Davoli, R. Minciardi, R. Zoppoli, "On the structure of decentralised dynamic routing strategies", IEEE 22nd CDC, WP 9-3:15, Vol. 1, pp. 472-476, December 1983.

[175] J. M. Garcia, J. C. Hennet, "Models and control structure for a large telephone network", Large Scale Systems, 1980.

[176] K. H. Muralidhar, M. K. Sundareshan, "Combined routing and flow control in computer communication networks: A two-level adaptive scheme ", IEEE Transactions on Automatic Control, Vol. AC-32, pp. 15-25, January 1987.

[177] A. Pitsillides, J. Lambert, "An Optimal Control Approach to Dynamic Bandwidth Allocation in B-ISDN", Communications'92, Sydney, October 1992.

[178] A. Pitsillides, J. Lambert, "A Multilevel Control Theoretic Approach to Dynamic Bandwidth Allocation in Broadband-ISDN", TENCON'92, Melbourne, November, 1992.

[179] A. Pitsillides, J. Lambert, B. Warfield, "Multilevel optimal control of virtual paths in Broadband-ISDN", internal working paper, Laboratory for Telecommunication research, Swinburne University of Technology, March 1992.

[180] K. L. Rider, "A simple approximation to the average queue size in the time-dependent M/M/1 queue", Journal of the Association for Computing Machinery, Vol. 23, No. 2, pp. 361-367, 1976.

[181] A. Pitsillides, J. Lambert, N. Li, J. Steiner, "Dynamic Bandwidth Allocation in Communication Systems: An Optimal Control Approach", IEEE International Conference on Systems Engineering, Kobe, Japan, September 1992.

[182] S. Sharma, D. Tipper, "Approximate models for the study of nonstationary queues and their application to communication networks", IEEE International Conference on Communications, ICC'93, Geneva, May 1993.

[183] M. F. Neuts, "Matrix-Geometric solutions in Stochastic Models," John Hopkins Univ. Press, 1981.

[184] T. G. Robertazzi, "Computer networks and systems: Queuing theory and performance evaluation", Springer-Verlag, 1990.

[185] A. Reibman, K. Trivedi, "Numerical analysis of markov models" Comput. Operations Research, Vol. 15, No. 1, 1988.

[186] M. S. Mahmoud, "Optimal control of constrained problems by the costate coordination structure", Automatica, Vol. 14, pp. 31-40, 1978.

[187] M. Jamshidi, C. M. Wang, "A computational Algorithm for Large-Scale Nonlinear Time-Delay Systems", IEEE Transactions of Systems, Man, and Cybernetics, Vol. 14, No. 1, pp. 2-9, January 1984.

[188] K. Permar, V. S. Vaidya, "A multilevel control algorithm for nonlinear time-delay systems", International Conference on Systems, Man, and Cybernetics, pp. 700-703, Vol. 2, 1987.

[189] W. Xu, "A multilevel control of large scale time delay systems with serial", SICE'90, Tokyo, 1990.

[190] M. F. Hassan, M. G. Singh, "A two level costate prediction algorithm for non-linear systems", Automatica, Vol. 13, pp 629-634, 1977.

[191] M. F. Hassan R. Hurteau, M. G. Singh, A. Titli, "A three level costate prediction algorithm for continuous dynamical systems", Automatica, Vol. 14, pp 177-181, 1978.

[192] Y. Takahara, "A multilevel structure for a class of dynamic optimisation problems", in Optimisation Methods for Large Scale Systems, edited by D. A. Wismer, Mac-Millan, New York, 1971.

[193] M. F. Hassan, M. G. Singh, "Hierarchical successive approximation algorithms for non-linear systems", two parts, Large Scale Systems, Vol. 2, pp. 65-95, 1981.

[194] A. Pitsillides, "A multilevel control theoretic approach to B-ISDN based on the virtual path", internal report LTR0192, Laboratory for Telecommunications Research, Swinburne University of Technology, January 1992.

[195] B. D. O. Anderson, J. B. Moore, "Linear optimal control", Prentice Hall, Englewood Cliffs, New Jersey, 1971.

[196] J. N. Tsitsiklis, M. Athans, "Guaranteed robustness properties of multivariable, nonlinear, stochastic optimal regulators, IEEE 22nd CDC, WP 6 - 4:00, pp. 389-394, San Antonio, Tx, December 1983.

[197] D. D. Siljak, R. C. Dorf, "On the minimisation of sensitivity in optimal control systems", 3rd Allerton Conf. on Circuit and System theory, University of Illinois, pp. 225-229, October 1965.

[198] R. I. Badr, M. F. Hassan, J. Bernussou, A. Y. Bilal, "Time-domain robustness and performance optimisation of linear quadratic control problems", IEE Proceedings Vol. 135, Pt. D, No. 4, July 1988.

[199] D. P. Bertsekas, "Dynamic programming and stochastic control", Academic Press, New York, 1976.

[200] M. G. Singh, A. Titli, K. Malinowski, "Decentralised control design: An overview", Plenary lecture, IFAC Large Scale Systems, Warsaw, 1983.

[201] B. Warfield, P. Sember, "Prospects for the use of artificial intelligence in real-time network traffic management", Computer Networks and ISDN Systems, Vol. 20, pp. 163-169, 1990.

[202] A. P. Sage, C. White, "Optimum Systems Control", Englewood Cliffs, NJ, Prentice Hall, 1977.

[203] F. Lewis, "Optimal Control", John Wiley, New York, 1986.

[204] A. Konheim, Private communique, October 1992.

[205] L. Kleinrock, "Queuing systems, Volume II: Computer applications", John Wiley, 1976.

[206] J. S. Meditch, "Optimality properties of proportional capacity assignment in message and packet-switched networks", IEEE 22nd CDC, WP 9-3:30, pp. 477-482, San Antonio, Tx, December 1983.

[207] W. P. Lovegrove, "Methodology for studying transient behaviour in computer networks using simulation", Ph.D. thesis, Clemson University, 1990.

[208] D. Tipper, "Simulation results on the dynamic bandwidth allocation algorithm, using a discrete event non-stationary simulation", Private communique, 1992.

[209] S. G. Tzafestas, K. Watanabe (editors), "Stochastic large-scale engineering systems", Marcel Dekker, New York, 1992.

[210] M. Aoki, "Control of large-scale dynamic systems by aggregation", IEEE Transactions of Automatic control, Vol. AC-13, pp. 246-253, June 1968.

[211] P. V. Kokotovic, R. E. O'Malley, P. Sannuti, "Singular perturbations and order reductions in control theory-An overview" Automatica, Vol. 12, pp. 123-132, 1976.

[212] L. S. Lasdon, "Optimisation theory for Large-Scale systems" MacMillan, New York, 1970.

[213] N. J. Smith, A. P. Sage, "An introduction to hierarchical systems theory", Comput. &Elect. Engng. Vol. 1, pp. 55-71, 1973.

[214] W. Findeisen, "Decentralised and hierarchical control under consistency or disagreement of interests", Automatica, Vol. 18, No. 6, pp. 647-664, 1982.

[215] M. S. Mahmoud, "Multilevel systems control and applications: A survey", .IEEE Transactions System, Man, Cybernetics, Vol. SMC-7, pp. 125-143, 1977.

[216] M. Javdan, "On the use of Lagrange duality in multi-level optimal control", IFAC Symposium on Large Scale System Theory and Applications, Udine, Italy, 1976.

[217] M. F. Hassan, M. G. Singh, "The optimisation of Non-Linear systems using a new two-level method", Automatica, Vol. 12, pp. 359-363, 1976.

[218] Y. Y. Haimes, D. Li, "Hierarchical Multiobjective Analysis for Large-Scale Systems: Review and current status", Automatica, Vol. 24, No. 1, pp. 53-69, January 1988.

[219] B. H. Soong, "Telecommunication traffic estimation and control", Ph.D. Thesis, Electrical Engineering Department, The University of Newcastle, Australia, 1988.

[220] D. Tipper, J. L. Hammond, S. Sharma, A. Khetan, K. Balakrishnan, S. Menon, "An Analysis of the congestion effects of link failures in wide are networks", internal working paper, Clemson University, Clemson, USA, 1992.

[221] K. S. Watson, "Performance evaluation of cyclic service strategies- A survey", Performance'84, pp. 521-533, Elsevier Science Publishers B.V., North Holland, 1984.

[222] D. Macko, Y. Y. Haimes, "Overlapping coordination of hierarchical structures", IEEE Transactions System, Man, Cybernetics, Vol. SMC-8, pp. 745-751, October 1978.

[223] T. Shima, K. Tarvainen, Y. Y. Haimes, "Hierarchical overlapping coordination for large scale systems", 8th Triennial IFAC World Congress, Kyoto, Japan, 1981.

[224] M. Herzberg, "Network Bandwidth Management-A new direction in network management", 6th Australian Teletraffic Research Seminar, Wollongong, NSW, November 1991.

[225] H. W. Kuhn, A. W. Tucker, "Nonlinear programming", 2nd Berkeley Symposium on Math. Stat. and Prob., University of California press, pp. 481-492, 1951.

[226] L. A. Zadeh, "Optimality and non-scalar-valued performance criteria", IEEE Automatic Control, Vol. AC-8, pp. 59-60, 1963.

[227] M. Zukerman, S. Chan, "Fairness in Broadband ISDN" IEEE INFOCOM '92, Florence, Italy, May 1992.

[228] M. Gerla, M. Staskauskas, "Fairness in flow controlled networks", Journal of Telecommunication Networks, Computer Science Press, pp. 29-38, 1982.

[229] J. Von Neumann, O. Morgenstern, "Theory of games and economic behaviour", Princeton Uni. press, 3rd ed., 1953.

[230] J. Eatwell, M. Milgate, P. Newman (editors), "Game theory: The new palgrave" The MacMillan Press, 1989.

[231] T. Basar, J. Olsder, "Dynamic Noncooperative Game Theory", Academic Press, London, 1982.

[232] V. R. Saksena, J. B. Cruz, "Optimal and near-optimal incentive strategies in the hierarchical control of Markov Chains", Automatica, Vol. 21, No. 2, 181-191, 1985.

[233] J. G. Lin, "Three methods for determining Pareto-optimal solutions of multiple-objective problems", in Directions in Large-Scale Systems, edited by Y-C. Ho and S. K. Mitter, Plenum Press, 1975.

[234] K. Tarvainen, "On the generating of Pareto optimal alternatives in large scale systems", IFAC Large Scale Systems: Theory and applications, Zurich, 1986.

[235] Y. Y. Haimes L. Lasdon, D. Wismer, "On a bicriterion formulation of integrated systems identification and system optimisation", IEEE Transactions System, Man, Cybernetics, Vol. SMC-1, pp. 296-297, 1971.

[236] Y. Y. Haimes, W. A. Hall, H. T. Freedman, "Multiobjective optirnisation in water resources systems: The surrogate worth-tradeoff method", Elsevier, Amsterdam, 1975.

[237] J. Nash, "The bargaining problem", Econometrica, Vol. 18, pp. 155-162, 1950.

[238] X. Cao, "Preference functions and bargaining solutions", 21st CDC, Orlando, FL, pp. 164-171, December 1982.

[239] R. Mazumdar, L. G. Mason, C. Douligeris, "Fairness in network optimal flow control: optimality of product form", IEEE Transactions on Communications, Vol. 39, No. 5, May 1991.

[240] C. Douligeris, "Multiobjective Flow Control in Telecommunication Networks", IEEE INFOCOM'92, Florence, May 1992.

[241] M. Hsiao, A. A. Lazar, "Bottleneck modelling and decentralised flow control-I Global objectives and II Individual objectives", Proceedings on Information Sciences and Systems, pp. 169-173, pp. 558-563, Princeton, NJ, USA, March 1984.

[242] K. Arrow, F. Hahn, "General competitive analysis", Holden Day Publishers, 1971.

[243] B. Rustem, "The diagonalizability of quadratic functions and the arbitrariness of shadow prices", Automatica, Vol. 27, No. 3, pp. 573-574, 1991.

[244] Y. Ito, "Adaptive control of econometric models with integrated and decentralised policymakers", in Stochastic Large-Scale Engineering Systems, edited by S. G. Tzafestas and K. Watanabe, Marcel Dekker, New York, 1992.

[245] P. Gawthrop, H. Demircioglu, "Continuous-time generalised predictive control", IFAC symposium on adaptive systems in control and signal processing, pp. 123-128, Glasgow, UK. 1989.

[246] H. Dernircioglu, P. Gawthrop, "Multivariable continuous-time generalised predictive control (MCGPC)", Automatica, Vol. 28, No. 4, pp. 697-713, 1992.

[247] J. Sun, P. Ioannou, "Robust adaptive LQ control schemes", IEEE Transactions on Automatic control, Vol. AC-37, No. 1, pp. 100-105, January 1992.

[248] A. Ishigame, S. Kawamoto, T. Taniguchi, "Fuzzy optimal controller based on neural network for nonlinear systems", IEEE International Conference on Systems Engineering, Kobe, Japan, September 1992.

[249] J. L. Hammond, J. D. Spragins, "Rapidly reconfiguring computer communication networks: Definition and major issues", IEEE INFOCOM 87, pp. 202-206, San Fransisco, USA, 1987.