

**BANDWIDTH ALLOCATION IN BROADBAND INTEGRATED  
SERVICES DIGITAL NETWORKS**

**Jose Augusto Suruagy Monteiro**

**July 1990  
CSD-900018**



UNIVERSITY OF CALIFORNIA

Los Angeles

**Bandwidth Allocation in Broadband  
Integrated Services Digital Networks**

A dissertation

submitted in partial satisfaction

of the requirements for the degree

Doctor of Philosophy in Computer Science

by

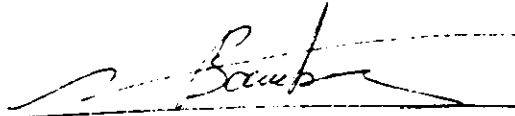
**José Augusto Suruagy Monteiro**

1990

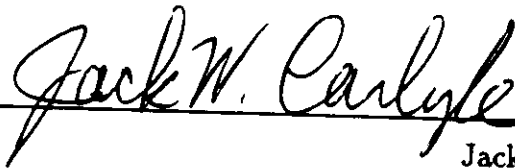
Copyright © 1990 by José Augusto Suruagy Monteiro

All Rights Reserved

The dissertation of José Augusto Suruagy Monteiro is approved.



Nicholas Bambos



Jack W. Carlyle



Leonard Kleinrock



Siu Tang



Mario Gerla, Committee Chair

University of California, Los Angeles

1990



This dissertation is dedicated to my children:  
Estêvão, Leticia, Mateus, and Renat(o|a)<sup>1</sup>;  
and to my wife, Maristelma.

---

<sup>1</sup>We are expecting him/her around August 9th.





## Table of Contents

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Introduction</b>  | <b>1</b>  |
| 1.1      | Integrated Services Digital Networks . . . . .               | 1         |
| 1.2      | Broadband Services . . . . .                                 | 3         |
| 1.3      | Fast Packet Switching . . . . .                              | 5         |
| 1.4      | Digital Cross-Connect Systems . . . . .                      | 7         |
| 1.5      | New Design Paradigms . . . . .                               | 8         |
| 1.6      | Dissertation Overview . . . . .                              | 9         |
| <b>2</b> | <b>Broadband ISDN</b>  | <b>11</b> |
| 2.1      | Broadband Services . . . . .                                 | 11        |
| 2.1.1    | Service Parameters . . . . .                                 | 12        |
| 2.1.2    | Traffic Types . . . . .                                      | 13        |
| 2.2      | Broadband Transport . . . . .                                | 15        |
| 2.2.1    | Introduction . . . . .                                       | 15        |
| 2.2.2    | Asynchronous Transfer Mode (ATM) . . . . .                   | 16        |
| 2.3      | Broadband Switching . . . . .                                | 17        |
| 2.3.1    | Switch Architectures . . . . .                               | 17        |
| 2.3.2    | Switches Performance . . . . .                               | 21        |
| 2.4      | Digital Cross-Connect Systems . . . . .                      | 27        |
| 2.5      | B-ISDN Architecture . . . . .                                | 28        |
| 2.6      | Reconfiguration with DCS . . . . .                           | 29        |
| <b>3</b> | <b>Resource Allocation</b>                                   | <b>33</b> |
| 3.1      | Introduction . . . . .                                       | 33        |
| 3.1.1    | Call level . . . . .   | 34        |
| 3.1.2    | Burst level . . . . .  | 35        |
| 3.1.3    | Cell level . . . . .   | 35        |
| 3.1.4    | Road Map . . . . .   | 35        |
| 3.2      | Traffic Models . . . . .                                     | 36        |
| 3.2.1    | Constant Bit Rate Traffic . . . . .                          | 36        |
| 3.2.2    | Bursty traffic model . . . . .                               | 36        |
| 3.2.3    | Variable-Bit-Rate (video) traffic model . . . . .            | 38        |
| 3.3      | Bandwidth Allocation . . . . .                               | 39        |
| 3.3.1    | Introduction . . . . .                                       | 39        |
| 3.3.2    | Bursty Traffic Analysis . . . . .                            | 40        |
| 3.3.3    | Variable-Bit-Rate Traffic Analysis . . . . .                 | 47        |
| 3.3.4    | Comparison of Bursty and Variable Bit Rate Traffic . . . . . | 52        |

|          |  |           |
|----------|--|-----------|
| 3.4      | Admission Control . . . . .  | 53        |
| 3.5      | Traffic Mixing Trade-offs . . . . .  | 55        |
| 3.5.1    | To integrate or not to integrate, that is the question . . . . .           | 55        |
| 3.6      | Multiplexers in Tandem and Bandwidth Allocation . . . . .                  | 58        |
| 3.6.1    | Output Process for a Single Multiplexer . . . . .                          | 59        |
| 3.6.2    | Multiplexers in Series . . . . .   | 62        |
| 3.6.3    | Multiplexers in Series with Merging Sources at Second Stage . . . . .      | 63        |
| 3.6.4    | Multiplexers in Series with Several Destinations and Merging . . . . .     | 65        |
| 3.6.5    | Multiplexers Fed by other Multiplexers with Several Destinations . . . . . | 66        |
| 3.7      | Summary . . . . .  | 67        |
| <b>4</b> | <b>Input Rate Control Mechanisms</b>                                       | <b>69</b> |
| 4.1      | Introduction . . . . .   | 69        |
| 4.2      | The Ideal IRC Mechanism . . . . .  | 71        |
| 4.3      | Input Rate Control Mechanisms . . . . .                                    | 74        |
| 4.3.1    | Credit Counter . . . . .   | 74        |
| 4.3.2    | Jumping and Moving Windows . . . . .                                       | 75        |
| 4.3.3    | Leaky Bucket . . . . .   | 77        |
| 4.3.4    | Delta . . . . .  | 78        |
| 4.3.5    | Delta-2 . . . . .  | 80        |
| 4.3.6    | Delta-3 . . . . .  | 81        |
| 4.4      | Analysis of IRC mechanisms under bursty traffic . . . . .                  | 84        |
| 4.4.1    | Jumping Window analysis under bursty traffic . . . . .                     | 84        |
| 4.4.2    | Leaky Bucket analysis under bursty traffic . . . . .                       | 87        |
| 4.4.3    | Delta mechanism analysis under bursty traffic . . . . .                    | 100       |
| 4.4.4    | Delta-2 mechanism analysis under bursty traffic . . . . .                  | 102       |
| 4.4.5    | Delta-3 mechanism analysis under bursty traffic . . . . .                  | 108       |
| 4.5      | Analysis of IRC mechanisms under VBR traffic . . . . .                     | 112       |
| 4.5.1    | Jumping Window analysis under VBR traffic . . . . .                        | 112       |
| 4.5.2    | Leaky Bucket analysis under VBR traffic . . . . .                          | 113       |
| 4.5.3    | Delta-2 mechanism analysis under VBR traffic . . . . .                     | 120       |
| 4.5.4    | Delta-3 mechanism analysis under VBR traffic . . . . .                     | 122       |
| 4.6      | IRC Mechanisms Comparison . . . . .  | 123       |
| 4.6.1    | Conformity to the ideal marking probabilities . . . . .                    | 124       |
| 4.6.2    | Effect on well-behaved sources . . . . .                                   | 124       |
| 4.6.3    | Reaction time . . . . .  | 126       |
| 4.6.4    | Implementation complexity . . . . .  | 127       |
| 4.6.5    | Summary . . . . .  | 127       |
| 4.7      | Conclusions . . . . .  | 129       |

|          |   |            |
|----------|---|------------|
| <b>5</b> | <b>B-ISDN Design Problem</b>                                    | <b>131</b> |
| 5.1      | Broadband Network Architecture . . . . .                        | 132        |
| 5.2      | Design Problems . . . . .                                       | 134        |
| 5.3      | Reconfiguration . . . . .                                       | 136        |
| 5.3.1    | Reconfiguration Time Scales . . . . .                           | 139        |
| 5.3.2    | Topology Tuning . . . . .                                       | 140        |
| 5.4      | Design as a P/S Network . . . . .                               | 140        |
| 5.4.1    | Embedded Network Design . . . . .                               | 142        |
| 5.4.2    | Topology Tuning . . . . .                                       | 146        |
| 5.4.3    | Example . . . . .   | 147        |
| 5.4.4    | Summary . . . . .   | 151        |
| 5.5      | Design as a C/S Network . . . . .                               | 151        |
| 5.5.1    | Assumptions . . . . .   | 152        |
| 5.5.2    | Integrated Network Routing Strategy . . . . .                   | 155        |
| 5.5.3    | Application of INR to ATM Networks . . . . .                    | 156        |
| 5.5.4    | Multiservice C/S Network Design . . . . .                       | 157        |
| 5.6      | Summary and Future Work . . . . .                               | 158        |
| <b>6</b> | <b>Conclusions</b>  | <b>161</b> |
| 6.1      | Contributions . . . . .   | 161        |
| 6.2      | Future work . . . . .   | 164        |
| <b>A</b> | <b>Uniform Arrival and Service Model</b>                        | <b>167</b> |
| <b>B</b> | <b>Uniform Arrival and Service Model for Finite Buffer Size</b> | <b>171</b> |
| <b>C</b> | <b>Invariance of R with same K/L ratio</b>                      | <b>173</b> |
| <b>D</b> | <b>Variable Bit Rate Traffic Fluid Flow Model</b>               | <b>175</b> |
| <b>E</b> | <b>Bandwidth Allocation and Routing in the Embedded Network</b> | <b>177</b> |
|          | <b>Bibliography</b>   | <b>181</b> |



## List of Figures

|  |    |
|--|----|
| 1.1 Network Communications before ISDN. . . . .  | 2  |
| 1.2 Network Communications with ISDN. . . . .  | 2  |
| 1.3 Channel Bit Rate and Duration of a Session for a Variety of Services. . . . .                                  | 5  |
| 1.4 Banyan Based FPS Fabric. . . . .   | 6  |
| 1.5 Operation of a Digital Cross-Connect System (DCS). . . . .   | 7  |
|  |    |
| 2.1 Constant Bit Rate (Periodic) Traffic. . . . .  | 14 |
| 2.2 Bursty Traffic. . . . .  | 14 |
| 2.3 Variable Bit Rate Traffic. . . . .   | 15 |
| 2.4 Digital Cross-Connect Switch. . . . .  | 27 |
| 2.5 Long-term broadband network architecture. . . . .  | 29 |
| 2.6 Backbone topology. . . . .   | 30 |
| 2.7 Embedded topology A. . . . .   | 31 |
| 2.8 Embedded topology B. . . . .   | 31 |
|  |    |
| 3.1 Terminal State Layers. . . . .   | 34 |
| 3.2 Bursty Source. . . . .   | 37 |
| 3.3 Markov Chain Model for a bursty Source. . . . .  | 37 |
| 3.4 Variable Bit Rate (VBR) Source. . . . .  | 38 |
| 3.5 Simulation vs. UAS comparison ( $P = 10^{-5}$ ). . . . .   | 42 |
| 3.6 Normalized $R$ ( $P = 10^{-9}$ ). . . . .  | 44 |
| 3.7 $R$ sensitivity with respect to $P$ ( $b = 2$ ). . . . .   | 44 |
| 3.8 $R$ sensitivity with respect to $P$ ( $b = 3$ ). . . . .   | 45 |
| 3.9 $R$ sensitivity with respect to $P$ ( $b = 10$ ). . . . .  | 45 |
| 3.10 $R$ sensitivity with respect to $L$ . . . . .   | 46 |
| 3.11 $R$ sensitivity with respect to $\alpha$ . . . . .  | 47 |
| 3.12 Simulation vs. UAS results for VBR sources ( $K=30\text{ms}$ ). . . . .                                       | 50 |
| 3.13 Simulation vs. UAS results for VBR sources ( $K=0.1\text{ ms}$ ). . . . .                                     | 50 |
| 3.14 $R$ sensitivity with respect to the GOS requirement ( $P$ ), and buffer size ( $K$ ) for VBR sources. . . . . | 51 |
| 3.15 Sensitivity of $R$ with respect to the buffer size ( $K$ ) for VBR sources. . . . .                           | 51 |
| 3.16 Comparison of Bursty and Variable Bit Rate Traffic. . . . .   | 52 |
| 3.17 Mixing of Bursty and Variable Bit Rate Traffic. . . . .   | 55 |
| 3.18 Mix of Bursty and CBR Sources ( $\text{GOS}=10^{-5}$ ). . . . .   | 57 |
| 3.19 Mix of VBR and CBR Sources ( $\text{GOS}=10^{-5}$ ). . . . .  | 57 |
| 3.20 Multiplexer Busy Period. . . . .  | 60 |
| 3.21 Multiplexers in Series. . . . .   | 63 |
| 3.22 Multiplexers in Series with Merging Sources at Second Stage. . . . .  | 63 |
| 3.23 Multiplexers in Series with Several Destinations and Merging. . . . .   | 65 |

|      |  |     |
|------|--|-----|
| 3.24 | Multiplexers Fed by other Multiplexers with Several Destinations. . . . .  | 66  |
| 4.1  | Ideal Policing Mechanism Behavior. . . . .   | 71  |
| 4.2  | Simulation system for the study of IRC mechanisms. . . . .   | 73  |
| 4.3  | Example of evolution of a Jumping Window counter. . . . .  | 75  |
| 4.4  | Example of evolution of Moving Window counter. . . . .   | 76  |
| 4.5  | Leaky Bucket Functional Diagram. . . . .   | 77  |
| 4.6  | Example of evolution of the state of a LB pseudo queue. . . . .  | 78  |
| 4.7  | Example of evolution of the state of the Delta counters. . . . .   | 80  |
| 4.8  | Example of evolution of the state of the Delta-2 counters. . . . .   | 82  |
| 4.9  | Example of evolution of the state of the Delta-3 counters. . . . .   | 83  |
| 4.10 | Jumping Window Marking Probabilities for a Bursty Source — 2<br>hours equivalent simulated time. . . . .                   | 85  |
| 4.11 | Effect of Marking on well behaved sources (Jumping Window). . . . .  | 86  |
| 4.12 | Leaky Bucket Behavior for a Bursty Source ( $GOS=10^{-5}$ ). . . . .   | 89  |
| 4.13 | Reaction Time for several counter sizes. . . . .   | 91  |
| 4.14 | Analytical versus Simulation Results for a Bursty Source Leaky<br>Bucket with $Q=200$ . . . . .                            | 94  |
| 4.15 | Analytical versus Simulation Results for a Bursty Source Leaky<br>Bucket with $Q=2,500$ . . . . .                          | 95  |
| 4.16 | Analytical versus Simulation Results for a Bursty Source Leaky<br>Bucket with $Q=10,000$ . . . . .                         | 95  |
| 4.17 | Leaky Bucket Marking Probabilities for a Bursty Source ( $B_e = B_m$ ). . . . .  | 97  |
| 4.18 | Leaky Bucket Marking Probabilities (1,000 sec. observation period). . . . .  | 99  |
| 4.19 | Effect of Marking cells on well behaved sources (Leaky Bucket). . . . .  | 99  |
| 4.20 | Delta Marking Probabilities ( $D_{pk} = 1$ sec.). . . . .  | 102 |
| 4.21 | Delta-2 Marking Probabilities ( $\rho_n = 0.5$ ). . . . .  | 104 |
| 4.22 | Delta-2 Marking Probabilities ( $\rho_n = 0.8$ ). . . . .  | 105 |
| 4.23 | Marking effect on well behaved sources for Delta-2 ( $\rho_n = 0.8$ ). . . . .   | 106 |
| 4.24 | Delta-3 Marking Probabilities — 2 hours equivalent simulated time. . . . .   | 109 |
| 4.25 | Delta-3 Marking Probabilities ( $\rho_n = 0.91$ ) - 1000 sec equivalent<br>simulated time. . . . .                         | 110 |
| 4.26 | Marking effect on well behaved sources for Delta-3 ( $\rho_n = 0.91$ ) . . . . .   | 111 |
| 4.27 | Jumping Window Marking Probabilities for a VBR Source. . . . .   | 112 |
| 4.28 | Effect of Marking cells on well behaved sources (Jumping Window). . . . .  | 113 |
| 4.29 | Leaky Bucket Behavior for a VBR Source ( $GOS = 10^{-5}$ ). . . . .  | 114 |
| 4.30 | Analytical versus Simulation Results for a VBR Source Leaky Bucket<br>with $Q=1K$ . . . . .                                | 117 |
| 4.31 | Analytical versus Simulation Results for a VBR Source Leaky Bucket<br>with $Q=10K$ . . . . .                               | 118 |
| 4.32 | Leaky Bucket Marking Probabilities for a VBR Source ( $B_e = B_m$ ). . . . .   | 118 |
| 4.33 | Leaky Bucket Marking Probabilities for a VBR Source ( $B_e = B_m$ )<br>- 1,000 seconds equivalent simulation time. . . . . | 119 |

|      |  |     |
|------|--|-----|
| 4.34 | Effect of Marking cells on well behaved VBR sources (Leaky Bucket).                          | 120 |
| 4.35 | Delta-2 Marking Probabilities (VBR source; $\rho_n = 0.8$ ).                                 | 121 |
| 4.36 | Delta-3 Marking Probabilities for a VBR source — 2 hours equivalent simulated time . . . . . | 123 |
| 4.37 | Marking probabilities comparison for IRC mechanisms (bursty source).                         | 124 |
| 4.38 | Effect on well behaved sources comparison for IRC mechanisms (bursty source). . . . .        | 125 |
| 4.39 | Reaction times comparison for IRC mechanisms (bursty source). . .                            | 126 |
| 5.1  | Backbone topology. . . . .   | 137 |
| 5.2  | Embedded topology A. . . . .   | 137 |
| 5.3  | Embedded topology B. . . . .   | 138 |
| 5.4  | Best network for original traffic. . . . .   | 148 |
| 5.5  | Best network after traffic change. . . . .   | 149 |
| 5.6  | Topology Tuning: initial solution. . . . .   | 150 |
| 5.7  | Topology Tuning: Discrete solution. . . . .  | 150 |





## List of Tables

|      |   |     |
|------|---|-----|
| 1.1  | CCITT Classification of Broadband Services . . . . .  | 4   |
| 3.1  | Multiplexer Parameters for Bursty Traffic ( $GOS = 10^{-5}$ ). . . . .  | 62  |
| 3.2  | Number of Extra Sources ( $M$ ) for Multiplexers in Series with Merging. . . . .  | 65  |
| 3.3  | Number of Extra Sources ( $M$ ) for Multiplexers in Series with Several<br>Destinations and Merging. . . . .  | 66  |
| 3.4  | Number of First Stage Multiplexers ( $n$ ) and Extra Sources ( $M$ ) for<br>Multiplexers Fed by other Multiplexers with Several Destinations. . . . . | 67  |
| 4.1  | Jumping Window Reaction Time (lower bounds) . . . . .   | 87  |
| 4.2  | Cell loss/marketing probability for $B_e \approx B_m (L = 100)$ . . . . .   | 88  |
| 4.3  | Leaky Bucket Parameters for Bursty Traffic ( $GOS = 10^{-5}$ ) . . . . .  | 90  |
| 4.4  | Leaky Bucket Reaction Time (lower bound) . . . . .  | 93  |
| 4.5  | LB Cell marking probabilities for $B_e \approx B_m (L = 100)$ . . . . .   | 97  |
| 4.6  | Delta-2 Reaction Time (lower bound) . . . . .   | 107 |
| 4.7  | Leaky Bucket Parameters for VBR Traffic ( $GOS = 10^{-5}$ ) . . . . .   | 114 |
| 4.8  | Cell loss/marketing probability for $B_e \approx B_m$ (VBR source). . . . .   | 116 |
| 4.9  | "Basic" Elements for IRC Mechanisms. . . . .  | 128 |
| 4.10 | IRC Mechanisms Comparison Summary. . . . .  | 128 |
| 5.1  | Interconnection topology per level . . . . .  | 133 |
| 5.2  | Original Traffic Matrix . . . . .   | 148 |



## ACKNOWLEDGMENTS

This is probably one of the most difficult pages to write in this dissertation. How can I fully acknowledge all the support and friendship I received from so many people during my tenure here at UCLA in the past 5 years? Let me try...

First of all, I want to thank my advisor, professor Mario Gerla, for his continuous support, encouragement, and friendship, as well as the other members of my committee: Jack Carlyle, Leonard Kleinrock, Chris Tang, and Nicholas Bambos.

Many other faculty members have been supportive in the course of these five years. In particular, I would like to thank Dan Berry, David Martin, and Larry McNamee (who even hooded me in the School of Engineering commencement).

I am also in debted to many people of the CS department staff: Verra Morgan, for being part of my family at UCLA; Doris Sublette, Rosie Murphy, Alexandra Pham, Judy Williams, Roberta Nelson, and Saba Hunt.

I would also like to recognize the following colleagues for their friendship and help: Frank Schaffa, Joe Bannister, Charlie Tai, Beto Avritzer, and Berthier Ribeiro.

Several people outside the UCLA community were very instrumental in the work reported in this dissertation. They are Stefano Rigobello of Telettra, José Roberto Boisson de Marca of the Pontifícia Universidade Católica do Rio de Janeiro, and in particular, Luigi Fratta of the Politecnico di Milano.

I would especially like to thank my family for their support and sacrifices made. Most of the burden was sustained by my wife, Maristelma, who had not only to adjust to a new environment, but also to raise our children with only my partial help. I am in debted to my children: Estêvão, Letícia, and Mateus for the time

taken away from them in these precious years of their lives. Much credit goes also to my parents, for their support back in Brazil, and for the time my father spent here alleviating my family chores. A special thanks to aunt Maria Alice who came to help us both when Mateus was born in '88, and now.

Finally, I would like to express my gratitude for the support of my employer, Universidade Federal de Pernambuco (UFPE), and all the people that took my place during my leave; and to CAPES from the Brazilian Education Secretariat that partially supported me under grant 6886/84-4.

## VITA

- June 23, 1957    Born, Recife, Pernambuco, Brazil
- 1979            B.Sc., Electrical Engineering  
Universidade Federal de Pernambuco  
Recife, Pernambuco, Brazil
- 1981-1982      Member of the Engineering Staff  
Fundação para o Desenvolvimento Tecnológico da Engenharia  
São Paulo, São Paulo, Brazil
- 1982            M.Sc., Electrical Engineering (Digital Systems)  
Universidade de São Paulo  
São Paulo, São Paulo, Brazil
- 1983-present   Assistant Professor (on leave)  
Departamento de Informática  
Universidade Federal de Pernambuco  
Recife, Pernambuco, Brazil

## PUBLICATIONS

- Cunha, P. R. F., Monteiro, J. A. S., Lucena, E. B. (April, 1985). Implementation of a Transport Protocol for the CEPINNE Network (in Portuguese). In: *Anais do 3º Simpósio Brasileiro sobre Redes de Computadores*, Rio de Janeiro, Brazil.
- Gerla, M., Monteiro, J. A. S., Pazos, R. (October, 1989). Configuring a Packet Network in an Integrated Environment. *IEEE Journal on Selected Areas in Communications*, 7(8):1253-1262.
- Monteiro, J. A. S. (November 1982). Description, Validation and Automatic Generation of Protocol Implementations (in Portuguese). Masters' thesis, Universidade de São Paulo, Brazil.
- Monteiro, J. A. S. (May, 1983). Design and Automatic Generation of Protocol Implementations (in Portuguese). In: *Anais do V Congresso Regional de Informática*,

Olinda, Brazil.

Monteiro, J. A. S., Jurema F., M. A., Cunha, P. R. F. (March, 1984). An X.25 Gateway for the Interconnection of a Mainframe to a Packet Switch Data Network (in Portuguese). In: *Anales de la X Conferencia Latino-Americana de Informatica*, Valparaiso, Chile.

Monteiro, J. A. S. (April, 1984). A Technique for the Design and Implementation of Protocols (in Portuguese). In: *Anais do 2º Simpósio Brasileiro sobre Redes de Computadores*, Campina Grande, Brazil.

Monteiro, J. A. S., Jurema F., M. A., Cunha, P. R. F. (April, 1984). A Gateway for the Interconnection of a DEC-10 Computer to a Packet Switch Data Network (in Portuguese). In: *Anais do 2º Simpósio Brasileiro sobre Redes de Computadores*, Campina Grande, Brazil.

Monteiro, J. A. S., Cunha, P. R. F., Rodrigues Filho, J. S. (April, 1985). On the Use of a Start/Stop-X.25 Conversor for Mainframe Access to Public Networks (in Portuguese). In: *Anais do 3º Simpósio Brasileiro sobre Redes de Computadores*, Rio de Janeiro, Brazil.

Monteiro, J. A. S., Gerla, M. (June, 1990). Topological Reconfiguration of ATM Networks. In: *Proc. INFOCOM '90*, pages 207-214, San Francisco, CA.

Monteiro, J. A. S., Gerla, M., Fratta, L. (June, 1990). Statistical Multiplexing in ATM Networks. In: *Proc. Fourth International Conference on Data Communication Systems and their Performance*, pages 148-162, Barcelona, Spain.

Monteiro, J. A. S., Gerla, M., Fratta, L. (September, 1990). Leaky Bucket Analysis for ATM Networks. To be presented at: *SBT/IEEE International Telecommunications Symposium*, Rio de Janeiro, Brazil.

ABSTRACT OF THE DISSERTATION

# Bandwidth Allocation in Broadband Integrated Services Digital Networks

by

**José Augusto Suruagy Monteiro**

Doctor of Philosophy in Computer Science

University of California, Los Angeles, 1990

Professor Mario Gerla, *Chair*

Broadband Integrated Services Digital Network (B-ISDN) is the new generation communication network that will provide transport facilities for a variety of traffic sources such as video, voice and data in an integrated environment. In this dissertation, we study *bandwidth allocation* problems in such networks.

We start by evaluating the statistical multiplexing gain for bursty as well as variable bit rate (VBR) traffic, using a fluid-flow approximate model. We obtain the required bandwidth per source in a finite buffer multiplexer in order to achieve a given Grade Of Service (GOS), expressed by the cell loss probability. For both bursty and VBR traffic sources, we perform a sensitivity analysis of significant parameters. The required bandwidth for bursty sources is shown to depend on burst and buffer length through their ratio. The mix of bursty and VBR traffic is considered. We compare the results obtained through simulation with approximations proposed in the literature. Finally, we study the multiplexing of bursty sources in the network internal buffers.

The knowledge of the bandwidth required by a source can be used by the admission control mechanism in order to decide whether or not a new call can be accepted still guaranteeing the GOS. However, this will be achieved only if the sources comply with the parameters specified at call set-up. Input Rate Control (or Policing) mechanisms have the function of assuring that the sources abide by their initial specification. We use an analytic model introduced elsewhere and simulation, to evaluate the effectiveness of policing mechanisms proposed in the literature as well as 3 new ones.

The design of B-ISDNs differs from the design of conventional data networks also in the possibility of dynamically reconfiguring the logical network through the use of Digital Cross-connect Systems (DCS). This reconfiguration can be performed in order to reduce store and forward delay and nodal processing overhead, as well as for congestion control purposes. We present an algorithm for the topology tuning of B-ISDN using a packet switching design approach, and suggest some improvements on a multiservice circuit switching design strategy to account for the variability of required bandwidth with the traffic mix.



## Chapter 1

### Introduction

#### 1.1 Integrated Services Digital Networks

ISDN stands for *Integrated Services Digital Network*. Nowadays, a corporation or the home user need to rely on several separate communication networks in order to transfer data of different types (Fig. 1.1) [Pan87]. The idea behind ISDN is that of an *Information Outlet* which as the power outlet must be universal and ubiquitous [Roc87]. Therefore, ISDN is supposed to provide a common interface for the transfer of data of all these different types (Fig. 1.2) [Pan87]. Another important aspect of ISDN is its flexibility to accommodate new services.

In a first phase, the now called *narrowband* ISDN will provide end to end digital connectivity to support the transfer of voice, data and low speed images. New services include enhanced (intelligent) telephony such as abbreviated dialing, call waiting with display indication among others. Current standards (for the *narrowband* ISDN) defines a *basic access* at 144 kbit/s (two 64 kbit/s B channels and a 16 kbit/s signaling D channel) and a *primary access rate* at 1.5/2 Mbit/s.

*Broadband* ISDN (or B-ISDN for short) is the next step with an user interface around 150 Mbits/s that will allow full-motion video applications such as teleconferencing and videophone.

Several technological developments led to ISDN. Among them are the fiber-optic and the gallium arsenide (GaAs) technologies [VV87]. Modulation bit rates of optical transmission systems are currently at more than 1 Gbit/s with repeater

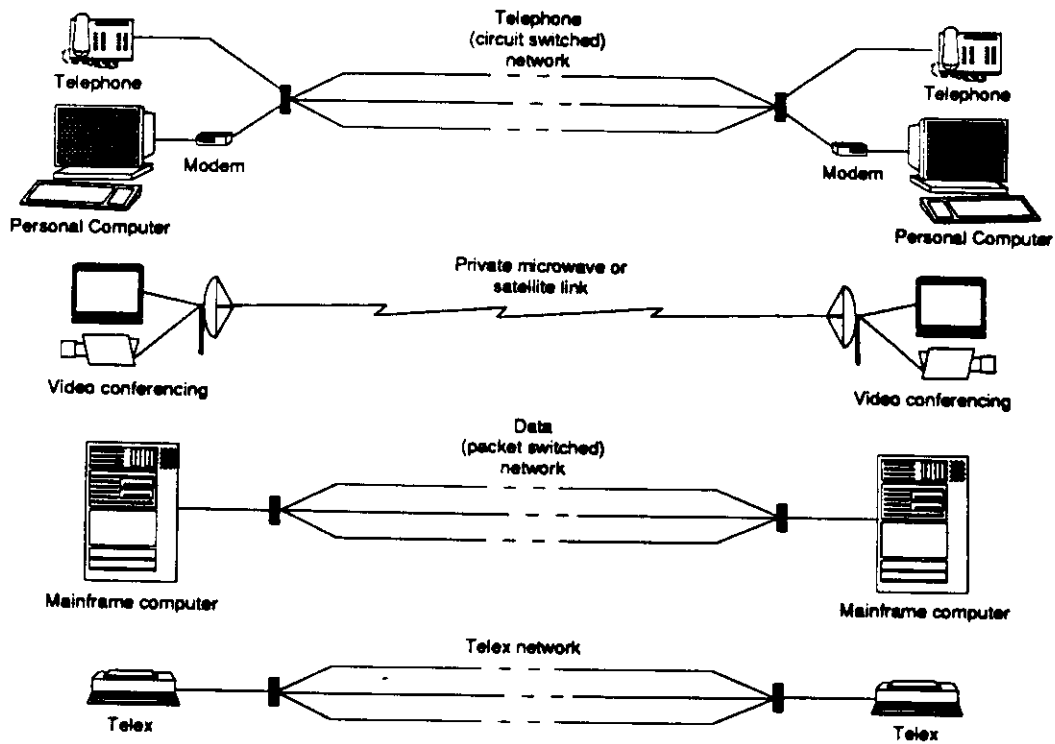


Figure 1.1: Network Communications before ISDN.

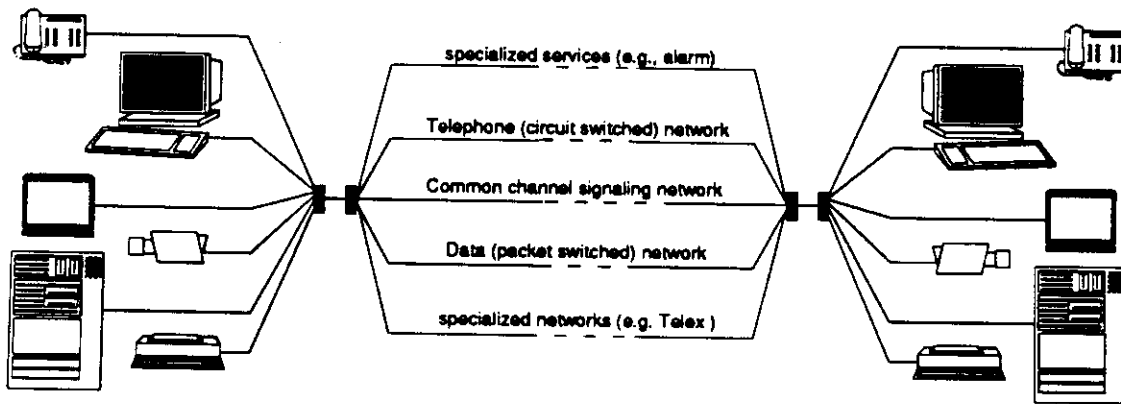


Figure 1.2: Network Communications with ISDN.

span that exceeds 100km [IK88] while a direct modulation of semiconductor lasers was already reported at 16 Gbit/s [HA88]. With the intensive deployment of fibers – initially in the inter-office trunks – the cost of one meter of single-mode fibers has already dropped from \$5 in 1982 to \$.25 in 1988 and is expected to drop to \$.04 by 1993 [MS89]. This cost reduction will make feasible the substitution of copper twisted wire pairs by fibers in the local loop which in turn will allow that B-ISDN services reach the homes.

The architecture, technology, and applications of ISDN (in particular, *narrow-band* ISDN) are discussed in a book edited by Verma [Ver90]. For a comprehensive presentation of ISDN, see also [Sta89]. For a historic view of ISDN concepts see [Hab88].

## 1.2 Broadband Services

As Stephen Weinstein points out [Wei90], no single service can be associated with ISDN. Rather, there are services that ISDN makes available to a large amount of customers at reasonable prices.

The CCITT<sup>1</sup> in its recommendation I.121 [CCI88] defines two main broadband service categories: Interactive and Distribution services. Each of these categories are subdivided into classes as shown in table 1.1.

Conversational class services includes videotelephony [Lio90], video conference, video-surveillance, video/audio information transmission service, and high-speed data transmission. Messaging class services includes video and document mail. Retrieval services includes broadband videotex, video retrieval service, audio and archive information.

---

<sup>1</sup> *Comité Consultatif Internationale de Télégraphique et Téléphonique*, an arm of the International Telecommunication Union of the United Nations.

Table 1.1: CCITT Classification of Broadband Services

|              |  |
|--------------|--|
| Interactive  | Conversational                               |
|              | Messaging                                    |
|              | Retrieval                                    |
| Distribution | without user individual presentation control |
|              | with user individual presentation control    |

Distribution services without user individual presentation control are those services that the user cannot control the start and order of presentation of the broadcast information. They include existing quality TV, High-Definition TV (HDTV), pay TV, and electronic newspaper. On the other hand, distribution services with user individual presentation control are those that the information is provided as a sequence of information entities with cyclical repetition. In this class we have remote education and training, advertising, and news retrieval.

Each one of these services can be characterized by several parameters such as calling rate, average bit rate, peak bit rate, burstiness, hold time, delay and data loss sensibility. By the large spectrum of services one can imagine that their traffic have also very different parameters. To illustrate this point we just need to compare the channel data rate and holding time (duration of a session) of entertainment video and Telemetry connections shown in Figure 1.3 [Wei87].

Some applications are more susceptible to delay or to data loss than others. For example, voice traffic can tolerate a certain amount of degradation but large delays can disrupt a conversation; while data traffic can tolerate short delays but no errors.

All this traffic diversity adds to the complexity of the transport network. In

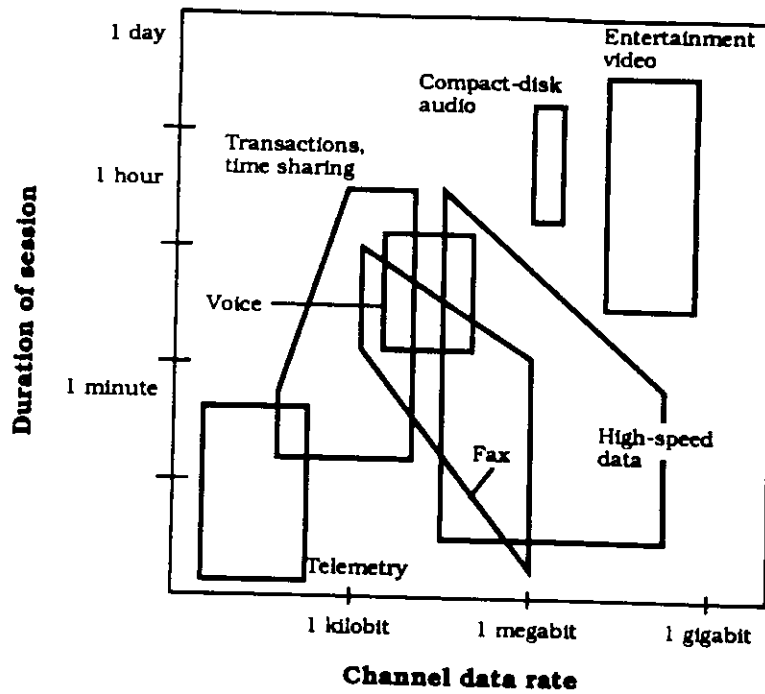


Figure 1.3: Channel Bit Rate and Duration of a Session for a Variety of Services.

fact, in order to deal with this diversity and to benefit from the investment already made on digital telephone and data equipments is that, in its initial phase, ISDN will provide an integrated network interface, but with a transport network as segregated as before (see Fig. 1.2). However, the ultimate goal of ISDN is to have also an integrated transport network with all the benefits of building and maintaining a single network.

### 1.3 Fast Packet Switching

Even though a lot of discussion has taken place on whether to use circuit switching (C/S), packet switching (P/S) or both in the transport network, a simplified form of P/S called *Fast Packet Switching* (FPS) [TW83, Tur86b] or *Asynchronous Transfer Mode* (ATM) [DQ87, GH87, CCI88, Min89], as it is now also known, is

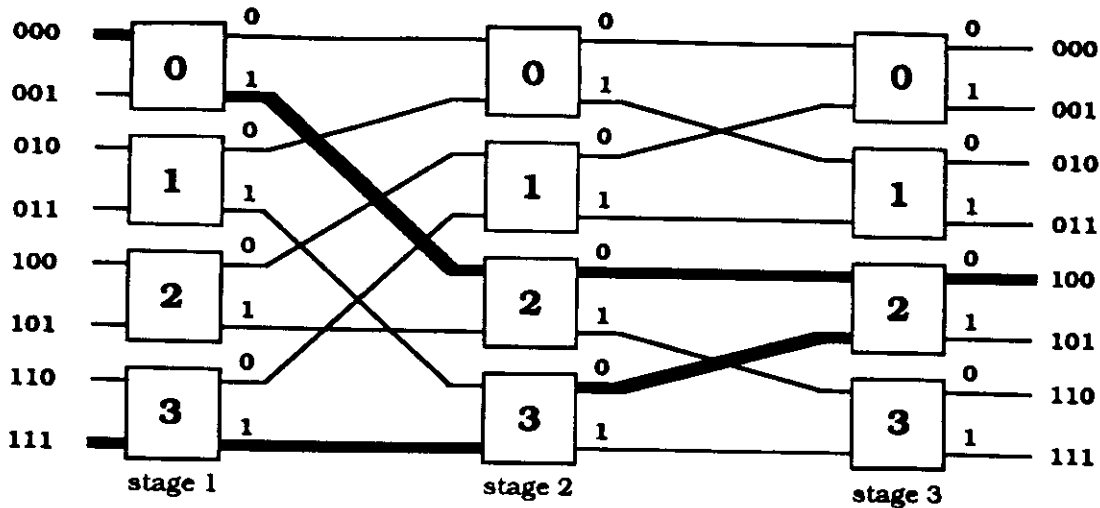


Figure 1.4: Banyan Based FPS Fabric.

the technology of choice.

In FPS, functions such as flow control and error recovery are implemented in an end-to-end rather than in a link-by-link basis as in conventional packet-switching, therefore reducing node processing overhead. Furthermore, the only way a fast-packet switch can achieve the required large bandwidth is through parallelism. Interconnection networks such as *Banyan networks* [GL73] have been proposed as the basis of FPS fabrics [Tur86a] (see Fig. 1.4).

Figure 1.4 shows a 3-stage (8x8) single-buffered Banyan network built with 2x2 switching elements (SEs). Each SE is a crossbar switch where each of its input lines have a buffer of size 1. The interconnection of SEs in a Banyan network is such that there is a unique path from any network input line to any network output line. In the case of Banyan networks with 2x2 SEs, *routing* can be automatically performed according to the destination port binary address. At stage  $i$  the  $(n-i)$ th digit of the destination port binary address is used to decide to which of the SE output links the packet must be routed to: 0 and 1 correspond respectively to the upper or lower output link. In Figure 1.4 it is shown in bold lines two paths from

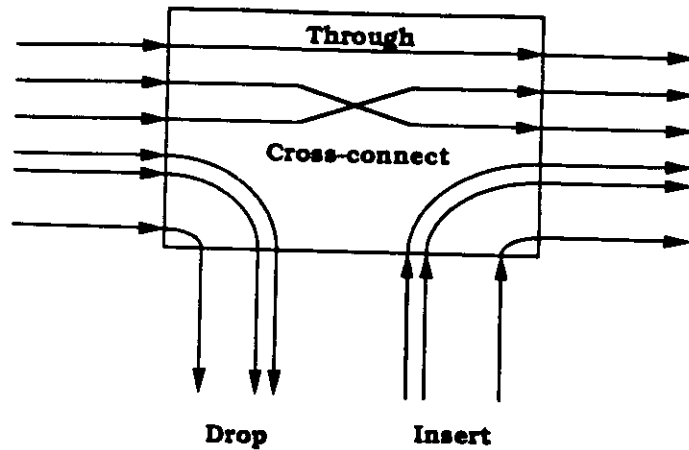


Figure 1.5: Operation of a Digital Cross-Connect System (DCS).

distinct input links (0 and 7) to the same output link (4).

With the potentially high bandwidths offered by fibers, the network *bottlenecks* have moved from the links to the nodes.

#### 1.4 Digital Cross-Connect Systems

Finally, in Broadband ISDNs, the Fast Packet Switches will be connected with multiples of basic rate trunks. These trunks will be obtained from an underlying “pool” of fiber facilities interconnected by Digital Cross-Connect Systems (DCS) [AW88], also known as *Digital Access and Cross-Connect Systems* (DACCS). The operation of a DCS is shown in Figure 1.5. We will also assume that the DCS has the ability to cross-connect entire fibers without the need of demultiplexing the stream of data into lower capacity channels and multiplex them again.

The ability to reconfigure a customer network dynamically is a well known advantage of Digital Cross-Connect Systems and has been reported extensively in the literature [AW88], [YH88], [Zan88], [Ami88].

## 1.5 New Design Paradigms

The new technologies that make possible broadband ISDNs, such as fibers, fast packet switches, and ATM are also responsible for a shift in design paradigms. This is true whether we compare them with circuit or packet switching networks.

These can be summarized as follows:

- Fiber deployment.
- Switching costs higher than fiber costs.
- Propagation delay dominates switching and transmission delays.
- Low cell loss probability requirement.
- Streamlined protocols.
- End-to-end flow control only.

Indeed, in traditional data communication networks most of its costs comes from link costs. However, due to the high bandwidth available in fibers associated to its small volume and consequent economy of scale, brings fiber costs down. In the other hand, exactly due to the high bandwidth available on fibers, the switches become the bottleneck. Therefore, the trend is to favor direct connections.

In low speed networks, transmission and switching (queueing) delays dominate over propagation delays. In high speed networks the opposite is true. The propagation delay is limited by the speed of the light, which is particularly significant in cross-country links.

A low cell loss probability is required if our goal is to accommodate all kinds of services, including traditionally circuit switched services, which cells are lost only due to noise.



High bandwidths require little processing at intermediate nodes. Therefore, all intermediate protocols should be reduced to the bare necessary. In particular, flow control is left just to end nodes.

## 1.6 Dissertation Overview

In this dissertation, we address some of the new design issues associated with the design of broadband ISDNs. Among these we devote particular attention to congestion control problems, such as, the bandwidth allocation and input rate control (bandwidth enforcement) problems.

This dissertation is organized as follows: Chapter 2 gives a more detailed introduction on Broadband ISDNs. Chapter 3 introduces the traffic models that will be used throughout the dissertation, and the resource allocation problem, i.e., how many resources (in particular) bandwidth must be allocated to a given connection in order to achieve its desired grade of service (GOS)? Chapter 4 studies input rate control mechanisms, i.e., mechanisms designed to guarantee that the traffic sources stick to their specifications. Chapter 5 reviews the issues on the design of B-ISDN systems. Finally, Chapter 6 concludes with a summary of the original contributions of this dissertation, and directions for further work.



## Chapter 2

### Broadband ISDN

In this Chapter we give some more background information on broadband ISDNs, specially on those aspects that will be considered in later chapters. We start in section 2.1 with the characterization of broadband services and major traffic types, namely: bursty, variable bit rate, and constant bit rate traffic. In Section 2.2 broadband transport options are discussed and the *Asynchronous Transfer Mode* is detailed. Section 2.3 presents some of the architectures proposed for ATM switching. Results on the performance of such switches are reported with special emphasis on non-blocking switches with output buffers. Section 2.4 describes the digital access and cross-connected systems (DCS). Finally, Section 2.5 discusses the evolution of broadband ISDN from the currently deployed networks, in its local access loop and backbone networks.

#### 2.1 Broadband Services

With the advent of broadband networks, old services will be accessed through a common transport network, and new services will be accessible to a larger number of customers than with existing networks.

Some of the services are well established and well known with current technology, such as, telephony, and cable TV. For these services, demand is known, as well as traffic characteristics. For newer services, however, both demand and traffic characteristics are for the most part, unknown. Traffic characteristics depends

on the codification used and typical usage pattern, while demand will depend on customer acceptance, which may depend largely on marketing strategies.

In the remainder of this Section, we will define some of the parameters that can be used to characterize a service, as well as, basic traffic types considered in the literature.

### 2.1.1 Service Parameters

Each service, or ultimately, each call, can be characterized by several parameters such as call arrival rate, average call holding time, call blocking requirement, average bit rate, peak bit rate, burstiness, cell loss probability requirement, and maximum cell delay requirement. Let us now consider each of these in turn.

**Call arrival rate:** This parameter which refers to a given service gives the rate in which a new call of that same type is offered to the system. Usually it is assumed that the arrivals form a Poisson process. If this assumption still holds, the average arrival rate is enough to characterize the process, otherwise other parameters are required.

**Average call holding time:** Again, this parameter relates to a group of calls of the same type, and expresses the average duration of a connection of that type. A typical assumption is that the holding time is exponentially distributed.

**Call blocking requirement:** A call is said to be blocked, if upon arrival (call connection time) it does not find any available link that can accommodate this call. In circuit-switching networks, a call is blocked if there is not an available channel in the trunk. In ATM networks, a call is blocked if its acceptance would degrade the service provided to the other connections (e.g., cell loss probability) below the desired grade of service (GOS). This parameter is usually expressed by the blocking probability.

**Average bit rate, peak bit rate, and burstiness:** These parameters give a first order characterization of the traffic source. Usually, we just need two of these parameters, since burstiness can be defined as the ratio between peak and average bit rates.

**Cell loss probability requirement:** In a statistically multiplexed environment such as ATM, cells compete for common limited resources, and therefore, some of them can get lost. Some traffic types can tolerate a moderate number of losses, while others may require a more strict cell loss probability. This is one of the most important service parameters, and is usually the one that defines its grade of service.

**Maximum cell delay requirement:** Again, some services are more sensible to delay than others. For example, voice information must arrive within a given deadline or they are worthless. In the other hand, data traffic are usually insensitive to delay. This maximum delay requirement restricts the amount of buffering that a source can tolerate.

From the above parameters, some will be used during call set-up depending on the bandwidth allocation scheme used. Users might have the option of selecting previously defined profiles or defining their own.

### 2.1.2 Traffic Types

The basic traffic types that we will consider throughout this dissertation are:

- Constant Bit Rate (CBR or CBO),
- Bursty, and
- Variable Bit Rate (VBR).

In *constant bit rate traffic*, cells are periodically transmitted according to its average bit rate (see Figure 2.1). The peak bit rate is the same as the average, and therefore, the burstiness is one.

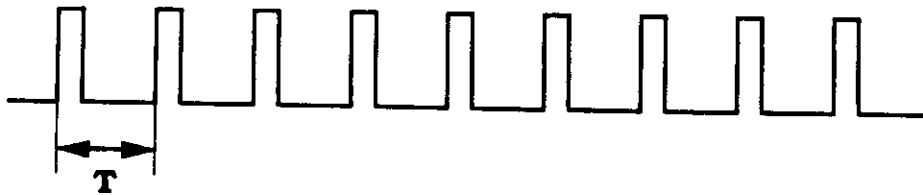


Figure 2.1: Constant Bit Rate (Periodic) Traffic.

*Bursty* traffic sources alternate between active periods in which they transmit at peak bit rate, and silence periods in which they keep quiet (i.e., no cell is transmitted) (see Figure 2.2).

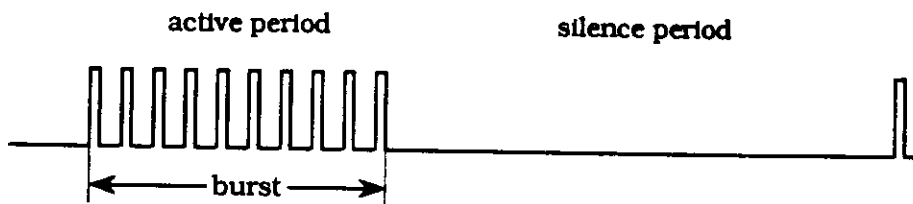


Figure 2.2: Bursty Traffic.

*Variable bit rate* traffic as the name says have a variable bit rate. This traffic type comes up for example, in the context of real-time video. By using a differential encoding, at the end of each frame, we just need to transmit the information of what have changed in the picture from the previous frame. Therefore, the amount of data that needs to be transmitted at the end of each frame is extremely dependent on the image sequence and on the coding scheme used. Figure 2.3 shows an example of the evolution of the bit rate, frame by frame, for a videotelephone type traffic.

Other than classifying traffic sources by their bit rate, we can classify them also by their priority, or alternatively, by their delay sensitivity. Gersht and Lee [GL89]

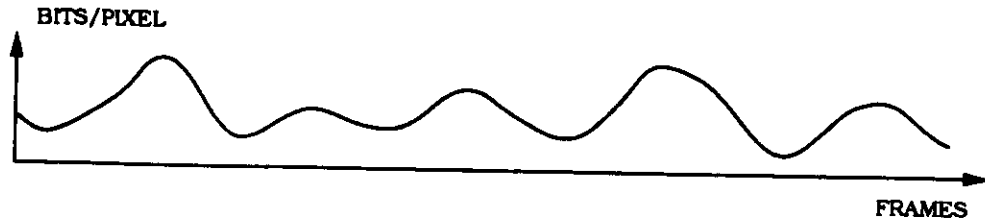


Figure 2.3: Variable Bit Rate Traffic.

classify the traffic into *express* and *first class* traffic. Express traffic corresponds to voice and video, while first class traffic corresponds to data traffic. In the other hand, Ramamurthy and Dighe [RD89] divides data traffic into *delay sensitive* (interactive traffic) and *delay insensitive* (file transfer or long response).

## 2.2 Broadband Transport

### 2.2.1 Introduction

The main technological force behind broadband networks is that provided by fibers and their ubiquitous deployment. This is due by their large available bandwidth, low loss, and small volume.

However, as far as the transport network is concerned, these fibers can be interconnected in different ways. Among the alternatives we have: segregated, integrated, and hybrid transport networks.

Currently, we have several segregated networks: one for telephony, one for data communications, one for telex, one for cable TV, etc. With B-ISDN we could have a common integrated interface but segregated transport network. In other words, video, voice, and data would all be transmitted through distinct transport networks.

Another alternative, which is the spirit of the Asynchronous Transfer Mode to be presented in the next subsection, is that of providing an integrated environment

not only for the user interface, but also for the transport of data from all different traffic sources.

Finally, a third alternative would be that of having some services sharing a common transport network, while others would have networks dedicated to them.

### 2.2.2 Asynchronous Transfer Mode (ATM)

Asynchronous Transfer Mode (ATM) is the switching and multiplexing technology that CCITT has targeted as the choice for implementing a broadband ISDN [CCI88, Min89]. ATM combines elements of both circuit and packet switching in order to achieve efficiency at very high speeds. It was originally proposed as Fast Packet Switching (FPS) [TW83], and Asynchronous Time Division (ATD) [TCS84].

As opposed to Synchronous Transfer Modes (STM), ATM has no slots reserved for channels. As in packet switching, each data unit (called *cell* in ATM jargon) contains its destination or the connection to which it belongs to, and this information is used by the switches in order to determine where it has to be sent. ATM has fixed size cells of 53 bytes (48 bytes of data plus 5 bytes of header).

The advantage of using ATM resides in the statistical gain obtained by multiplexing bursty services, and its flexibility in accommodating new services.

Minzer [Min89] presents a tutorial on broadband ISDN and ATM, reporting also some of the agreements reached by CCITT Study Group XVIII in their June 1989 meeting. For a critique to ATM see [GO89].



## 2.3 Broadband Switching

### 2.3.1 Switch Architectures

In order to achieve the large bandwidth made available by fibers and required by broadband services, fast-packet switches must rely on streamlined protocols and parallelism.

Functions such as flow control and error recovery are implemented in an end-to-end rather than in a link-by-link basis as in conventional packet-switching, therefore reducing node processing overhead.

Parallelism is achieved through the use of interconnection networks<sup>1</sup> (INs) [Fen81, Sie90] as the basis of FPS fabrics. In particular, two topological classes of INs have been extensively used: *Multistage interconnection networks* (MIN's) and *Crossbar*. For FPS applications, MIN's can be further divided into two classes: blocking, and non-blocking.

In blocking networks, the simultaneous transmission of packets may result in conflicts in the internal links. On the other hand, non-blocking networks can deliver all packets without any conflict provided that all of them go to distinct destination ports.

Each stage of a MIN is built from crossbar switches called switching elements (SEs). In order to avoid packet loss when conflict arises (i.e., packets at more than one input port is destined to the same output port), SEs can be buffered. If a MIN has internal buffers it is called *buffered*, otherwise it is called *unbuffered*.

Usually some sort of queueing is necessary since even if the interconnection network is non-blocking, there is always the possibility of more than one packet being simultaneously addressed to the same output port. Packets can be queued

---

<sup>1</sup>Proposed or used for connections in tightly coupled multi-processor systems.

at input or output ports.

Below we present some representative fast packet switch architectures according to the general classification given in precedence.

- **Multistage Interconnection Networks**

- blocking:

- \* **Fast Packet Switch** (Bell Labs.) [Tur86a]: buffered-Banyan network.
    - \* **Load-Sharing Banyan** (Georgiatech) [Lea86]: Banyan networks with multiple paths between a source and a destination pair.
    - \* **MSSR - Multi-Stage Self Routing** (Fujitsu Labs.) [HMI+88], [KSHM88]: Three-stage link configuration built from Self-Routing Module. The ATM switch operates at about 18 Mbps in 8-bit parallel.

- Non-blocking:

- \* **Starlite** (AT&T Bell Labs) [HK84]: Sorting (Batcher [Bat68]) and trap networks.
    - \* **Batcher-Banyan fabric** (Bellcore) [Hui87] [HA87]: If more than one packet is destined to the same output only one will get transmitted while the rest remains in their input queues.
    - \* **Parallel Delta Networks** (Fond. Ugo Bordonni) [BFL88]: Minimum number of Delta subnetworks that assures non-blocking condition is given.

- Crossbar

- **Knockout Switch** [YHA87] (AT&T Bell Labs.): Tournament algorithm at output ports (loss probability should be less than the one caused by other sources).
- **Bus Matrix Switch** (Fujitsu Labs.) [NTFH87]: Input and output transmission buses interconnected by cross-point memories.

The original FPS switch proposed by Turner [Tur86a] is built from a blocking single-buffered Banyan network (see fig. 1.4). The performance of Banyan networks can be severely reduced if the input traffic pattern is non-uniformly distributed to the output ports. In order to overcome this problem, Turner proposed the use of a *distribution network* in front of the Banyan network which would attempt to distribute packets evenly across all its output ports.

Other approaches to solve this problem are: the load-sharing Banyan network proposed by Lea [Lea86] where the Banyan topology is modified in order to create multiple paths between a source and a destination pair; the parallel Delta networks proposed by Bernabei et al. [BFL88] in which a minimum number of Delta subnetworks that assures non-blocking condition is used in parallel; and the MSSR - Multi-Stage Self Routing network [HMI+88], [KSHM88] which has a three-stage link configuration and multiple routes between a first stage and a third stage switching module.

Two of the above mentioned switch proposals uses sorting networks to avoid internal blocking. In the Starlite switch [HK84] packets are sorted according to their destination addresses. A packet will loose an output contention if the preceding packet in the sorted order has the same destination address. Loosing packets are fed back to the front end of the sorting network for reentry through a so called

trap network. However, it has some drawbacks: packets can be lost due to blocking within the reentry network; packets can be delivered out of sequence; and at least half of the input ports are dedicated for reentry.

The Batcher-Banyan fabric proposed by Hui et al. [Hui87] [HA87] uses the fact that a Banyan network is internally non-blocking if the packets are sorted according to their destination addresses. Packets are buffered at the inputs. If more than one packet is destined to the same output only one will get transmitted while the rest remain in their queues.

In the Knockout Switch [YHA87], each output port interfaces the input buses through packet filters that identifies the packets destined to that output port. A concentration circuit at each output selects a fixed number of packets simultaneously destined to the same output through a tournament algorithm (the loss probability should be less than the one caused by other sources) followed by an output queueing stage.

The Bus Matrix Switch [NTFH87] is built from a matrix of input and output transmission buses interconnected by cross-point memories. At a input bus, a packet is addressed to the memory at the cross-point with the desired output port. At the output port, a control interface goes sequentially through all cross-point memories connected to that output bus in search for packets to be transmitted in that output port.

Daddis and Torng [DT89] present a taxonomy of broadband integrated switching architectures, which also includes shared link architectures such as FDDI [Ros86] and Tree-net [Ger88], among others.

For surveys on switch architectures, see [AD89, Jac90].

### 2.3.2 Switches Performance

The performance measures that we are interested in are *average delay*, *throughput*, and *cell loss probability*. Basically, we want to study the behavior of these measures for various traffic loads.

Usually the best performance is achieved when the switch is submitted to a non-interfering traffic pattern. However, it is common to consider the well-behaved case of *uniform traffic*, where each input link has a uniform load, and each packet arriving at an input link is uniformly destined to all output ports.

We might also want to consider some special cases of non-uniform traffic patterns such as *point-to-point*, *mized point-to-point/uniform traffic*, and *hot spots*. In point-to-point traffic we have at least one input link sending all of its load to a unique output link. In mixed point-to-point/uniform traffic there is exactly one dedicated channel and the rest of input and output links form a community of uniform traffic. We say that we have a “hot spot” whenever a significant amount of the input traffic is destined to a single output port.

In the next two subsections we will review the work done on the performance analysis of buffered-Banyan and non-blocking networks, respectively.

#### 2.3.2.1 Performance of Buffered Banyan Networks

Buffered-Banyan networks have been mainly modeled by discrete-time Markov chains. The basic assumptions are: (1) clocked network operation, (2) fixed size packets, (3) immediate removal of packets at output ports, (4) random selection in case of internal contention, and (5) random reassignment of destination ports for blocked packets. These models lead to second order systems of non-linear equations that can be solved numerically.

Dias and Jump [DJ81] studied the throughput and delay of buffered delta networks (topologically equivalent to Banyan networks) under constant uniform maximum load.

Jenq [Jen83] modeled buffered-Banyan networks for any level of incoming uniform traffic. Jenq also analyzed the switching delay of a packet switch with infinite and finite input buffers. The infinite input buffer case is modeled by a discrete queueing system with independent arrivals and geometric departures. It was found that the switch delay is not significant until the input load approaches the maximum achievable throughput (0.4528). The blocking probability for the finite buffer case was obtained by modeling the system as an M/G/1/K queue with vacation.

Bernabei et al. [BILV85] [BFIL87] generalized Jenq's result for any SE size and buffer capacity.

In these models, in order to avoid a state-explosion problem, and since the traffic is uniform, it is assumed that the state of one stage can be well-characterized by the state of one of its SEs. Furthermore, if we assume that the SE buffers are independent [Jen83] a stage state can be characterized by the state of only one of its buffers. Using this approach, a second order system of non-linear difference equations is obtained and can be solved numerically.

Kruskal and Snir [KS83] derived an expression that yields a good approximation for the average packet delay of a buffered-Banyan network with large buffers. It was obtained through an embedded Markov chain (identical to the one used to analyze M/G/1 queues [Kle75]). The average transit time of a packet through the  $n$  stages of a Banyan network of degree  $m$  (SE dimension) is given by

$$D_{KS} = n \cdot \left( t_r + t_c \frac{(1 - 1/m)p}{2(1 - p)} \right) \quad (2.1)$$

where  $t_r$  is the *transit time* of a packet from one switch to the next one,  $t_c$  is the

*cycle time* of the switch (clock period), and  $p$  is the probability that an input port receives a packet at each cycle time.

Kruskal, Snir and Weiss [KSW86] found the generating function for the distribution of waiting time at the first stage of a buffered Banyan network (SE with output buffer) for a very general class of traffic. Traffic can be uniform or non-uniform, messages can have different sizes, and messages can arrive in batches. Using the delay formulas for the first stage under uniform traffic they developed good approximations for the delay at the other stages and, therefore, also for the total message delay. The total waiting time for messages of size one (one packet) and uniform traffic is approximately given by:

$$\bar{W}_{KSW} \approx n \left[ 1 + \frac{4p}{5m} \left( 1 - \frac{1 - \alpha^n}{n(1 - \alpha)} \right) \right] \frac{(1 - \frac{1}{m})p}{2(1 - p)} \quad (2.2)$$

where  $n$  is the number of stages,  $p$  is the probability that an input port receives one message at each unit of time,  $m$  is the number of inputs at each switching element of the interconnection network (typically  $m = 2$ ), and  $\alpha = 2/5$ . For a given Banyan network the only variable in Equation 2.2 is  $p$ .

Wu [Wu85], and Kim and Leon-Garcia [KLG88] studied the case of non-uniform input traffic pattern in single-buffered Banyan networks. Each switching element (SE) is modeled individually by a simple Markov chain and the relationships among different SEs are described through probabilistic means. Some of the considered non-uniform traffic patterns were point-to-point and mixed point-to-point/uniform traffic. Kim and Leon-Garcia also considered multibuffer and parallel Banyan networks.

Bubenik and Turner [BT89] used simulation to study the performance of the FPS proposed by Turner and briefly discussed in Subsection 2.3.1. They measured the delay and throughput of the switch under uniform traffic and under non-

uniform traffic with and without randomization (Distribution Network). They studied also the effect of *cut-through*, node size, and bypass queueing discipline (in which a non-blocked packet is allowed to proceed if all its predecessors in the queue are blocked). They examined also the performance of the copy network in a broadcast version of the FPS.

### 2.3.2.2 Performance of Non-blocking Networks

Even in non-blocking networks such as crossbar, queueing is necessary in cases where more than one packet is simultaneously addressed to the same output port. Karol, Hluchy, and Morgan [KHM87] compared input with output queueing, while Hluchy and Karol [HK88] studied the performance of non-blocking networks for four queueing alternatives: input queueing, input smoothing, output queueing, and completely shared buffering. Uniform traffic is assumed and packet arrivals at the input trunks are governed by identical and independent Bernoulli processes.

Input queueing was found to have a maximum throughput of 0.586 (compare it with 0.4528 for buffered-Banyan networks [Jen83]) when the switch is saturated, the number of input ports is large, and contention is solved randomly. When the input link utilization is high enough, switch throughput can be improved by dropping packets that loses a contention resolution. Waiting time for the random selection policy was obtained through the analysis of a discrete-time Geom/G/1 queue. Simulation was used to obtain the average waiting time for the longest queue and fixed priority selection policies.

In the input smoothing scheme, packets within a frame of size  $b$  are simultaneously presented to a  $Nb \times Nb$  switch fabric, where  $N$  is the number of input (and output) ports. However, it was found not to have much practical value.

Output queueing was modeled by discrete-time queues. The mean steady-state



waiting time for output queueing with infinity buffer size is given by

$$\bar{W} = \frac{(N - 1)}{N} \bar{W}_{M/D/1}$$

where  $N$  is the number of input (and output) ports. Note that the average waiting time approaches that of a  $M/D/1$  queue for large  $N$ . Loss probabilities can be obtained through the numerical solution of a finite state, discrete-time Markov chain.

Completely shared buffering allows savings on the total amount of buffering needed to achieve a desired packet loss probability, at the cost of increased complexity in buffer management. The steady-state number of packets in the buffer is modeled as the  $N$ -fold convolution of  $N$   $M/D/1$  queues.

Eckberg and Hou [EH88] found that buffers for output buffer sharing estimated using random traffic are overestimated by as much as 30% at 90% load, if the negative correlation between queues are ignored.

Eklundh et al. [ESS88] studied the block loss rate of switch nodes with output queues. An output port can be modeled as a single-server queue with deterministic service time. The arrival process is more difficult to model. Models used for regularized arrival processes include:  $Geo/D/1/K$ ,  $M/D/1$ , and  $nD/D/1$ . They proposed the use of a  $nTri/D/1/K$  model where the distribution of the interarrival time is the convolution of two uniform distributions and, therefore, it is triangular shaped. However, the model is extremely heavy on computation. The  $nTri/D/1/K$  model is compared to  $nD/D/1/K$ ,  $M/D/1/K$ , and  $Geo/D/1/K$  batch arrival models. They concluded that for systems with a moderate number of arrival streams, complex models have to be used and the results show that considerably fewer buffers are necessary. For systems with several hundreds or thousands of streams, instead, simple models like  $M/D/1/K$  are sufficient even if the arrival process is

regularized.

Hui and Arthurs [HA87] modeled their Batcher-Banyan switch with input buffering as  $N$  independent single-server queues, where  $N$  is the number of input ports. Packets arrive at each input port with probability  $\lambda$  per slot. They assume that each input port has a probability  $q$  of winning a switch arbitration.  $q$  is determined as a function of  $\lambda$ , and the individual input queues are analyzed.  $q$  is given by

$$\frac{1}{q} = 1 + \frac{\lambda}{2(1-\lambda)}$$

The total average delay ( $D$ ) (in number of slots) is given by

$$D = \frac{(2-\lambda)(1-\lambda)}{(2-\sqrt{2}-\lambda)(2+\sqrt{2}-\lambda)}$$

And the probability of loss for input buffers of size  $B$  is upper bounded by

$$P(\text{loss}) < P(K > B) = \frac{\lambda(2-\lambda)}{2(1-\lambda)} \omega_\lambda^B$$

where

$$\omega_\lambda = \frac{\lambda^2}{2(1-\lambda)^2}$$

and  $K$  is the average number of packets for the infinite buffer case.

Oie et al. [OMKM89] extended Karol et al. [KHM87] study of input versus output queueing, to include switching speedups ( $L$ ) between the two extremes: 1 and  $N$  (i.e.,  $1 \leq L \leq N$ ). They show that large throughputs can be achieved even at rather low speeds.

For a more complete survey of fast packet switch performance results, see [OSMM90b, OSMM90a]. In the sequel, we assume that the switches are non-blocking with output buffers.

## 2.4 Digital Cross-Connect Systems

*Digital Cross-Connect Systems* (DCS), also known as *Digital Access and Cross-Connect Systems* (DACs), are circuit-switching switches used to combine channels from different inputs into specific outputs. The switching pattern generally remains fixed for a long period of time. Channels multiplexed in a single fiber (via Time Division Multiplexing, Frequency Division Multiplexing, or both), are demultiplexed to the cross-connect rate, cross-connected, and multiplexed again in the output to the line (fiber) rate (see Figure 2.4). Current DCS's operate at DS-1 or DS-3 rates. Eng and Santoro [ES89] discuss two techniques for the implementation of an electro-optical cross-connect switch.

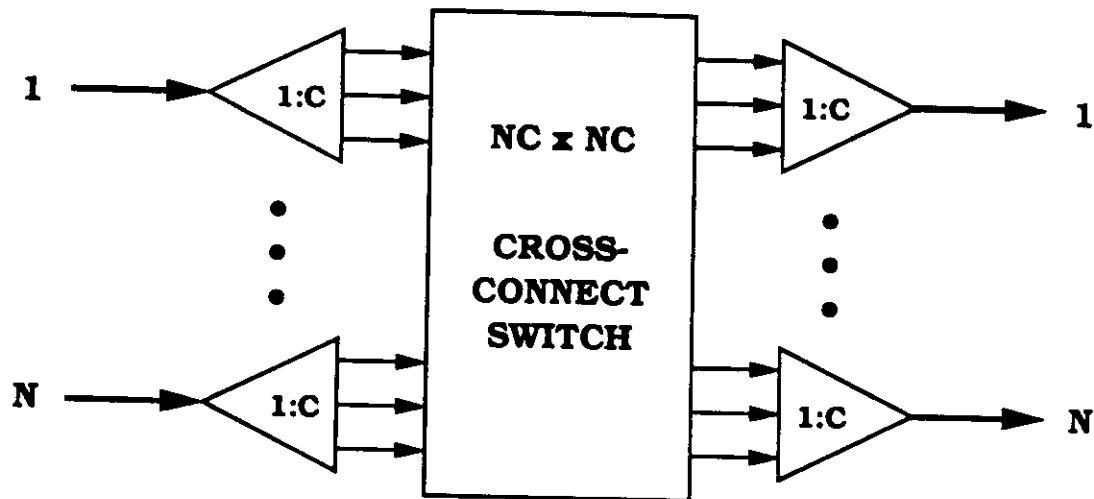


Figure 2.4: Digital Cross-Connect Switch.

DCSs are used for automatic restoration of networks after link failures [Gro87] or automatic reconfiguration due to changes in traffic pattern. This later use has as a goal to reduce network congestion.

## 2.5 B-ISDN Architecture

In this Section we review some of the proposals for the broadband ISDN architecture.

The evolution from conventional networks to broadband integrated networks passes by an interim phase where the interface is integrated, but the transport network is as segregated as before. This is exactly what the *narrowband* ISDN is.

Batorsky et al. [BST88] describe a long-term broadband network architecture, which is summarized in Figure 2.5. It is divided into several *plants* (hierarchical levels): distribution, sub-feeder, feeder, and interoffice plants. The interworking unit (IWU) located in the customer premises is the user access interface. As shown in Figure 2.5, services may include narrowband ISDN, Plain Ordinary Telephone Service (POTS), Local Area Networks (LAN), and HDTV. The distribution plant interconnects the IWU to the Remote Multiplexer (RM) in a star topology. The RM was introduced between the IWU and the Remote Electronics (RE), in order to reduce costs, since the connection to the IWU is usually made through a dedicated link (fiber). The interconnection of RMs to a RE in a sub-feeder plant can have several topologies, including star and ring, as shown in Figure 2.5. Similarly, REs are interconnected to Central Offices (CO) in the Feeder plant, and COs are interconnected to other COs or Hubs.

The choice of using a remote multiplexer or running a longer fiber to an RE will be dictated by economics. The same applies regarding the use of REs.

For other views on the evolution of B-ISDN see [Dec87, SA88, Car89, Kue89, MVB89].

As in conventional networks, we can make a distinction between *local access loop* (distribution) and *backbone network*. All the switching (including cross-connection)

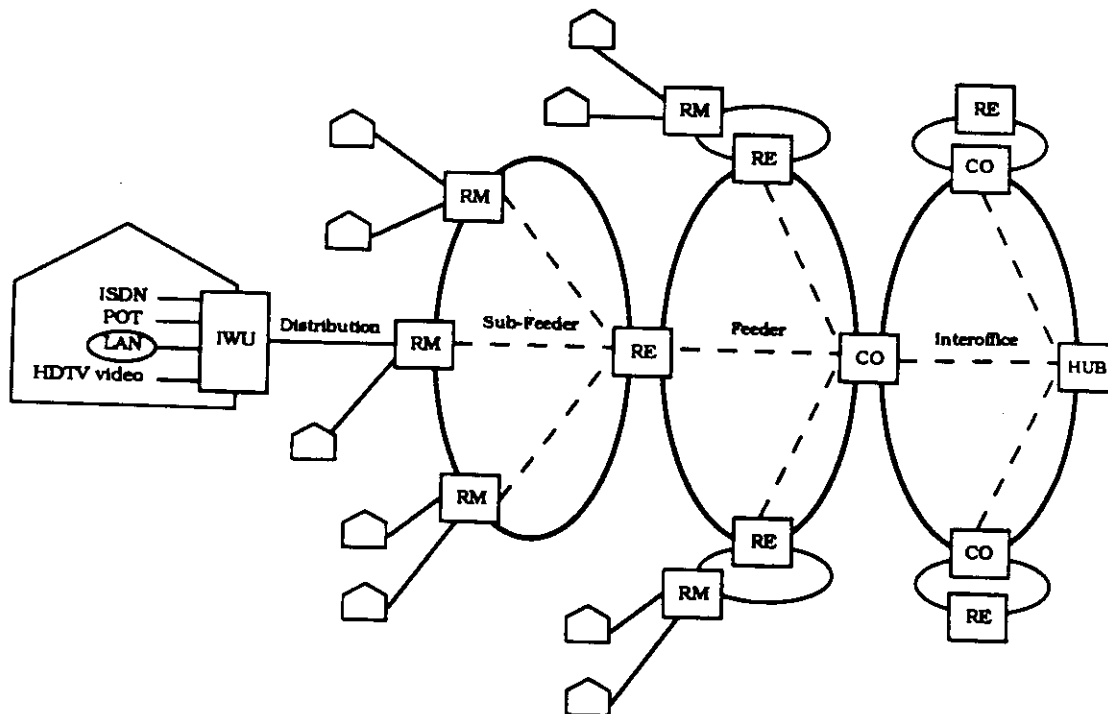


Figure 2.5: Long-term broadband network architecture.

is done at the backbone network. Liew and Cheung [LC89] proposes a multiple wavelength local-access optical network which is based on the use of an acousto-optic tunable filter. For the state-of-the-art on subscriber loop technology, see [SKY89], and for a cost analysis of local access loops see [ELLW89].

## 2.6 Reconfiguration with DCS

The ability to reconfigure a customer network dynamically is a well known advantage of Digital Cross Connect Systems and has been reported extensively in the literature [YH88, Zan88, FP88]. Most of the previous studies, however, have been based on transparent, circuit-switched type networks in which channels of various rates are established end-to-end between pairs of user sites. The main goals in the design of such systems were the dynamic network reconfiguration

following trunk failures, and the reassignment of trunks to applications following a predefined time schedule, or on a reservation basis, or in response to sudden traffic changes.

Rather than being concerned with the configuration of the transparent, circuit switched type network, in Chapter 5 we are interested in the ATM network built on top of the facility network. We want to exploit the DCS flexibility in order to obtain a more efficient design and operation of the ATM network.

To illustrate the point, consider the network shown in Fig. 2.6. From the original (backbone) topology, several embedded topologies can be derived. The embedded topology of figure 2.7 is identical to the backbone topology, whereas the topology of figure 2.8 has introduced a number of “express pipes” between remote nodes. Express pipes reduce the number of intermediate hops along the path and thus reduce store and forward delay and nodal processing overhead.

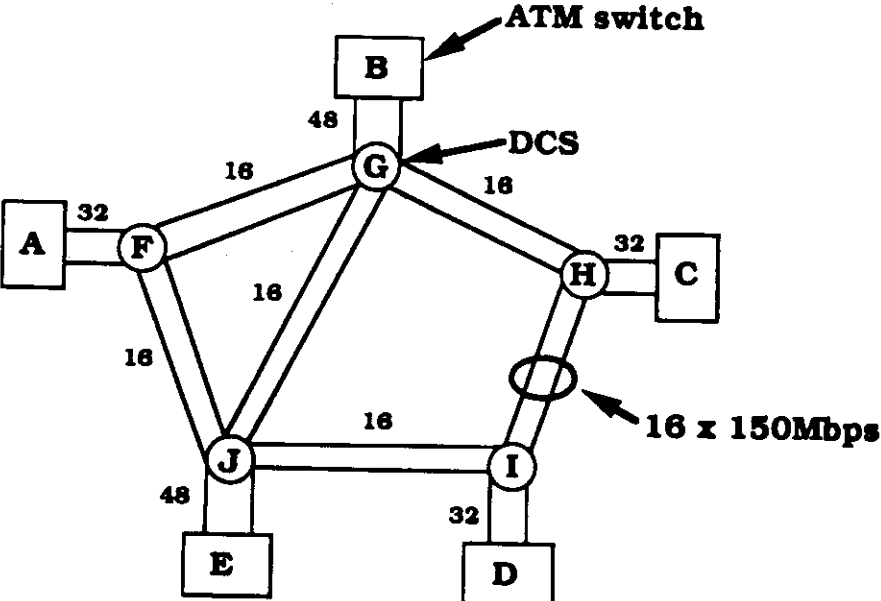


Figure 2.6: Backbone topology.

Express pipes also reduce the number of packet switch terminations. Note that

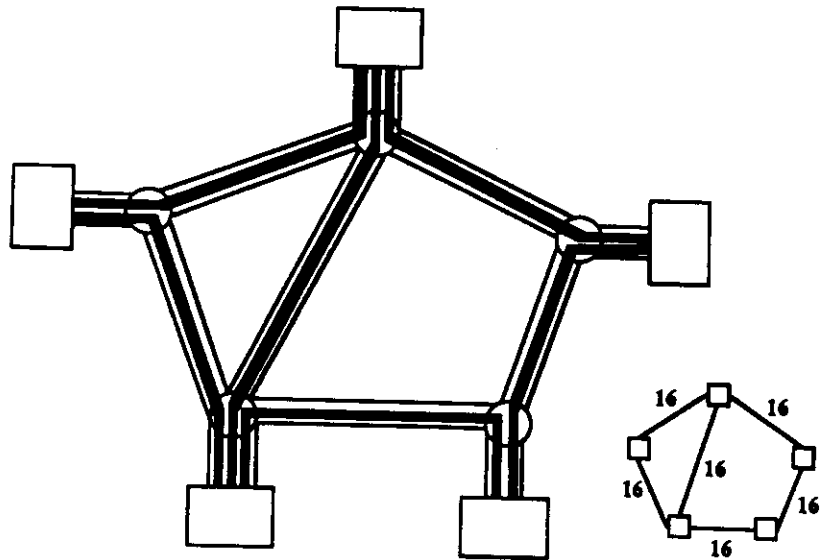


Figure 2.7: Embedded topology A.

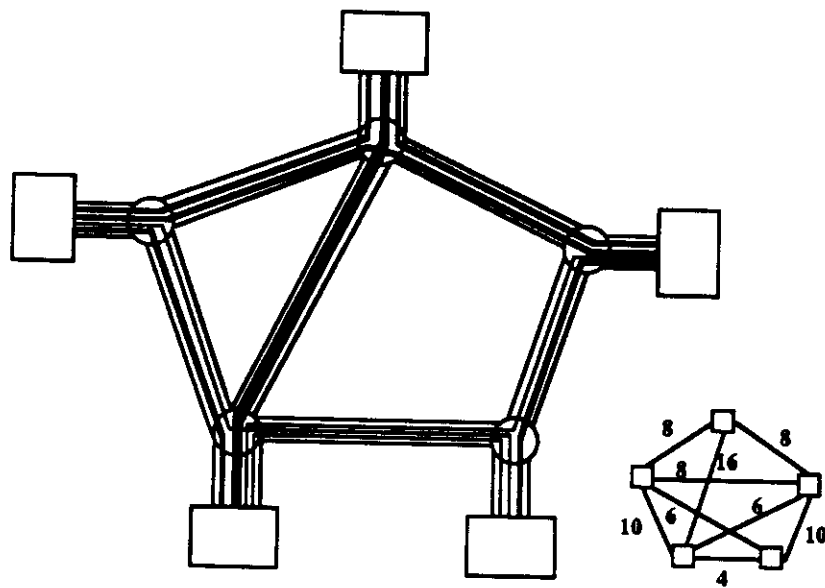


Figure 2.8: Embedded topology B.

topology 2.7 requires 192 packet switch terminations while topology 2.8 (which has more express pipes) requires 152 terminations. Using a fully piped network with a direct pipe (of capacity 4, say) also between nodes B and D will further reduce the number of terminations to 144. This is a very important point, since current trends in transmission and processing costs indicate that terminations costs will soon dominate the cost of fiber trunks. Thus, adding more express pipes will imply reducing overall costs.

Another important advantage of express pipes is that of simplifying the congestion control problem. In fact, when a pipe becomes congested, the offending source(s) can be immediately identified and slowed down. In contrast, in a purely meshed packet network, it is often very difficult to trace the sources that cause internal congestion, let alone control them.

There are also drawbacks in the configuration of fully piped (i.e. fully connected) embedded topologies. For example, in networks with a large number of nodes the bandwidth may become too fragmented, and the advantages of statistically multiplexing several sessions on the same trunk may be lost. Thus a good balance must be sought between express pipes and large trunks.

From the network management point of view, the DCS provides added flexibility in that it permits to dynamically tailor the topology to traffic demands. This "topology tuning" is of particular interest in broadband networks implemented using ATM techniques. In fact, ATM nets are stripped of most of the congestion and flow control procedures found in conventional networks, in order to improve switch throughput. If there is a mismatch between offered traffic pattern and network topology, congestion would be inevitable. The problem can be alleviated by dynamically tuning the topology to traffic pattern.



## Chapter 3

### Resource Allocation

#### 3.1 Introduction

One of the major advantages of a broadband network based on ATM is its flexibility in accommodating services with different traffic statistics. Resources can be allocated according to the specific needs rather than in a fixed fashion. In the other hand, in a shared environment such as ATM's we have to devise methods that will allow the network to allocate enough resources to each accepted connection so that they will not interfere with each other.

The state of a terminal can be split into several layers [Hui90, Hui88]: *subscription*, *call*, *burst*, and *cell* levels (see Figure 3.1). Each one of them has a characteristic time frame. A subscription period usually lasts for months or even years. The duration of a call depends on the specific service and it may last from few minutes to several hours. Bursts in the other hand, are an aggregate of information, typically in the order of milliseconds. Finally, cells in ATM are of fixed size with a transmission time of  $2.83 \mu\text{secs}$  at 150 Mbps.

Resources must be allocated for each of these levels. For example, at the subscription level at least a dedicated access connection must be allocated. At call level, or better at call connection time, a decision must be made on whether or not the network can support it. In multichannel schemes a set of channels is shared by a set of connections, and a particular channel is chosen in a burst-by-burst basis [Hui88, Pat88].

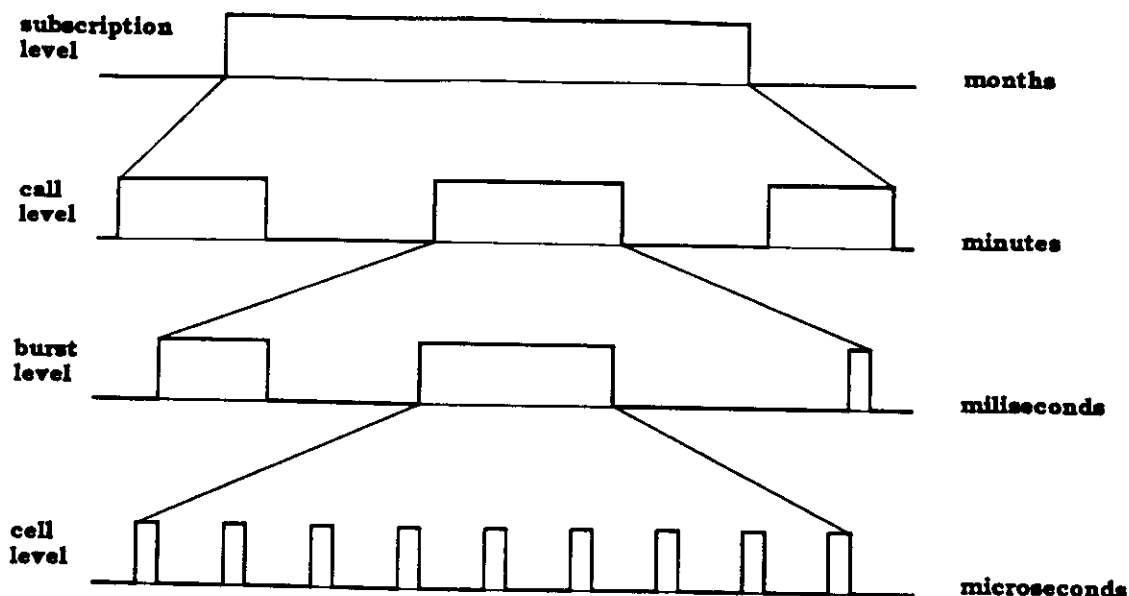


Figure 3.1: Terminal State Layers.

### 3.1.1 Call level

At the call level, we assume that all services are connection based, and connections can be of several types. In *point-to-point* connections, which corresponds to the conventional telephonic service, we just have two parties. *Multicast* connections allow for a large number of users to share the same connection. Multicast connections can be one-way only, as in distribution TV, or two-way as in video-conferences.

The exact amount of resources that have to be allocated to a given service, or connection, will also depend on the desired *Grade Of Service* (GOS). The GOS can be expressed by several performance measures, such as: cell loss probability, call blocking, and maximum cell delay.

In Hui's scheme [Hui88], when a call arrives, it calculates the burst blocking probability at each of the alternative trunk groups and accept the call only if the resulting blocking probability in at least one of the trunk groups is acceptable.

In Gersht and Lee scheme [GL89], bandwidth is reserved according to the traffic type. Express traffic (voice and video) is reserved the desired bandwidth, while first class traffic (data) is reserved only the guaranteed bandwidth with potential maximum bandwidth specified by the desired bandwidth parameter.

### 3.1.2 Burst level

Burst bandwidth allocation [Hui88] is based on *Fast Circuit-Switching*. A pilot packet initiates a bandwidth request and selects one channel from the channel group allocated to the connection during call set-up to transmit the whole burst. Burst allocation is not necessary when the trunk group is a single high-speed trunk, or when packets within a burst are randomly spread among the trunks for transmission. However, Hui argues that for large bursts of data, the processing overhead in finding an available channel might be negligible, and out-of-sequence packets are avoided.

### 3.1.3 Cell level

Gersht and Lee [GL89] proposes to use a choke/relieving scheme at network interface only for first class traffic (data).

In the other hand, Cassandras et al. [CKT89] proposes that real time messages be allowed to be rejected. The idea is to reject some messages that may not make their deadlines anyway.

### 3.1.4 Road Map

In this Chapter we study the problem of *bandwidth allocation* for three traffic models: Constant Bit Rate, Bursty, and Variable Bit Rate. Section 3.2 present the above traffic models. Section 3.3 studies the amount of bandwidth that must

be allocated to a given connection in order to guarantee a certain degree of cell loss probability for identical traffic sources. Section 3.4 explores some admission control policies, in particular of heterogeneous traffic sources. Finally, Section 3.6 studies the multiplexing of bursty sources in the network internal buffers.

## 3.2 Traffic Models

In this section we present the traffic models we will be using in the next Sections and Chapters, namely, constant bit rate (CBR), bursty, and variable bit rate (VBR) traffic models.

### 3.2.1 Constant Bit Rate Traffic

In *constant (continuous) bit rate traffic*, cells are periodically transmitted according to its average bit rate (see Figure 2.1). The peak bit rate is the same as the average, and therefore, the burstiness is one.

Several authors have studied the single server queue with periodic arrival process and deterministic service times. Eckberg [Eck79] derived an algorithm for computing the exact delay distribution of such a system with a number of sources with the same periodicity, and his results indicate that an M/D/1 approximation can be quite pessimistic. Virtamo and Roberts [VR89] obtained bounds for the buffer dimensioning of an ATM multiplexer (deterministic service) which input process is a superposition of a large number of periodic traffic ( $\sum N_i \cdot D_i / D / 1$  queue).

### 3.2.2 Bursty traffic model

A source is said to be *bursty* if it interleaves active periods (in which it transmits at the *peak* rate) with idle periods (in which it remains silent), see Figure 3.2.

Several traffic sources can be characterized as bursty. The best known examples are packet voice (coded with silence detection), and imaging services.

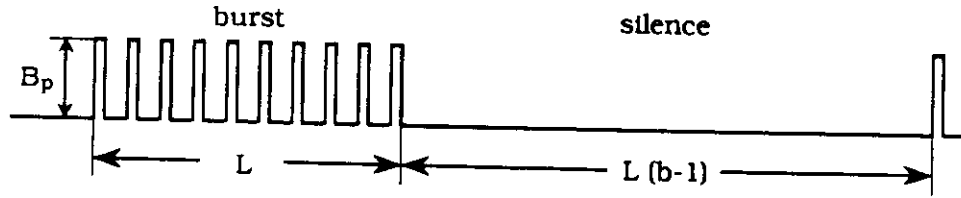


Figure 3.2: Bursty Source.

Following the notation of [GRF89], we characterize a bursty source by the following parameters:

- $B_p$ : peak bit rate.
- $B_m$ : mean bit rate.
- $T$ : mean burst duration.

Several burstiness measures can be defined. We define *burstiness* ( $b$ ) as the peak to mean bit rate ratio (i.e.,  $b = B_p/B_m$ ).

We assume that both active and silence periods are exponentially distributed with averages  $T$  and  $S = T(b - 1)$ , respectively. Therefore, a single bursty source can be modeled as a two state Markov chain as shown in Figure 3.3, where  $\lambda = 1/S$  and  $\mu = 1/T$ .

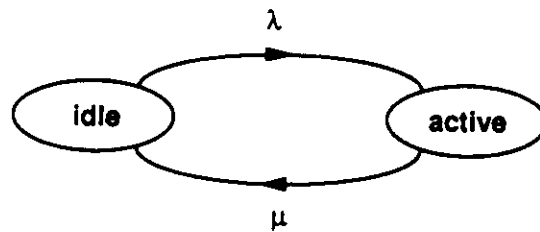


Figure 3.3: Markov Chain Model for a bursty Source.

ATM networks use fixed-size data units called *cells*. We will represent the cell length (in bits) by  $n_{cell}$ . Furthermore, unless otherwise specified, we will express the burst average length ( $L$ ) and the multiplexer buffer size ( $K$ ) in number of cells.

The burst average length ( $L$ ) relates to the other parameters through the following equation:

$$L = \frac{T}{n_{cell}/B_p}. \quad (3.1)$$

### 3.2.3 Variable-Bit-Rate (video) traffic model

Some traffic sources are neither strictly bursty (with active periods alternated with silence periods), nor constant bit rate. One example is video traffic that uses a Variable-Bit-Rate (VBR) encoding scheme [VPV88]. Although video could be transmitted at constant rate, a differential, VBR encoding scheme is more attractive because it takes advantage of the statistical multiplexing offered by ATM networks. In VBR, bit rate continuously changes on a frame by frame basis between a minimum and a maximum bit rate (Figure 3.4).

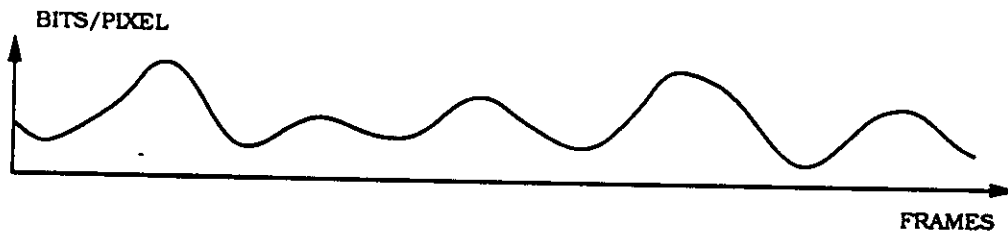


Figure 3.4: Variable Bit Rate (VBR) Source.

In Section 3.3.3 we present the analysis of the multiplexing of a video-phone type source traffic.

### 3.3 Bandwidth Allocation

#### 3.3.1 Introduction

In this Section we want to study the statistical multiplexing gain in ATM networks for homogeneous and heterogeneous traffic sources at the cell level. To this goal, we study the required bandwidth, for bursty sources as well as variable bit rate sources in a finite buffer multiplexer in order to achieve a given Grade Of Service (GOS)<sup>1</sup>, expressed by the cell loss probability.

An alternative approach to bandwidth allocations is that of *buffer dimensioning*, i.e., given the available bandwidth, what is the required buffer size in order to achieve a given GOS?

An extensive amount of related work has recently appeared in the literature. A comprehensive survey by Kawashima and Saito [KS89] reviews the teletraffic issues in ATM networks and reports on various voice and video statistical models. Some models apply only to voice sources while others apply to both voice and video sources. Among the latter we find the Markov Modulated Poisson Process (MMPP) [DL85, DL86, HL86], and the Uniform Arrival and Service (UAS) model [AMS82, DL86, Tuc88, MAS<sup>+</sup>87, MAS<sup>+</sup>88, DJ88]. As for the mix of heterogeneous traffic sources in the context of call admission control, models have been proposed by Dziong et al. [DCLM89], and Woodruff et al. [WKFR89], while Decina and Toniatti [DT90] studied the mixing of homogeneous and heterogeneous bursty sources through simulation.

We formulate the *bandwidth assignment problem* as follows: given a mix of  $N$  distinct type sources that share a transmission link, with buffer size  $K$ ; find the link bandwidth  $W$  that has to be assigned to this mix of traffic in order to satisfy

---

<sup>1</sup>Also referred to as Quality Of Service (QOS).

a given Grade Of Service (GOS) requirement.

The mixt of the  $N$  distinct type sources is represented by the tuple  $(n_1, n_2, \dots, n_S)$  where the  $n_i$ 's are the number of sources of type  $i$ ,  $S$  is the number of distinct source types, and  $N = \sum_{i=1}^S n_i$ . The GOS requirement is defined as the cell loss probability ( $P$ ). For ATM a typical value for  $P$  is  $10^{-9}$ .

We will express the assigned link bandwidth by the *expansion factor*,  $R$ , defined [GRF89] as the ratio of the assigned bandwidth ( $W$ ) over the total average bit rate produced by the  $N$  sources:

$$R = \frac{W}{\sum_{i=1}^S n_i B_m^i}$$

where  $B_m^i$  is the average bit rate for a source of type  $i$ .

The expansion factor gives us a measure of the excess bandwidth (relative to the average) that must be assigned to the incoming traffic in order to account for its burstiness. Note that  $R = \frac{1}{\rho}$ , where  $\rho$  is the multiplexer utilization factor.

Due to the multiplexing effect, we can expect that as the number of sources increases, the expansion factor should decrease until reaching  $R = 1$ . For single type bursty traffic sources, a peak bandwidth assignment implies  $R = b$ , while the lower bound (unattainable) is  $R = 1$ .

### 3.3.2 Bursty Traffic Analysis

Several models were proposed in the literature for the analysis of bursty traffic. Among them we mention the Markov Modulated Poisson Process (MMPP) [DL85, DL86, HL86], and the Uniform Arrival and Service (UAS) model [AMS82, DL86, Tuc88, MAS+87, MAS+88, DJ88]. For a survey of models and their references see [KS89]. Other results were obtained through simulation [GRF89].

For bursty traffic,  $R$  is a function of: the burstiness,  $B$ ; the number of identical sources,  $N$ ; the average burst length,  $L$ ; and; the buffer size,  $K$ . Recently, Gallassi



et al. [GRF89] observed that  $R$  is independent of the particular value of the peak bit rate ( $B_p$ ). In Appendix C, we show that  $R$  depends on  $L$  and  $K$  only through their ratio.

### 3.3.2.1 Solution approach

For a given bandwidth allocation, number of identical bursty sources, and buffer size, the cell loss probability  $P$  can be found using the Uniform Arrival and Service (UAS) model for finite buffer size, described in [Tuc88] (for a summary, see Appendix B).

In our problem, we want to determine the bandwidth that achieves a given GOS (cell loss probability). Thus, our algorithm searches for the appropriate value of bandwidth using a logarithmic interpolation method in the range  $N \cdot B_m < W < N \cdot B_p$ , which corresponds to  $1 > P > 0$ , where  $P$  is the cell loss probability. The search ends when the GOS is obtained within a given tolerance.

Since the expansion factor is non-increasing with the number of sources, if we are obtaining  $R$  for several number of sources, we can use the previous value of  $R$  as a starting point. In other words, the search can be restricted to the range  $N_i \cdot B_m < W < N_i \cdot B_m \cdot R_{i-1}$ , where  $R_{i-1}$  is the expansion factor for  $N_{i-1}$  sources, and  $N_{i-1} < N_i$ .

The UAS solution approach involves the computation of eigenvalues and eigenvectors for a tridiagonal real matrix and the solution of a set of linear equations (see [Tuc88] for details). Both numerical algorithms are of complexity  $O(N^3)$ . Since the interpolation converges in a number of steps which is independent of  $N$  (and in fact tends to decrease when channel utilization increases, i.e.,  $N$  becomes large), we conclude that the complexity of finding the bandwidth allocation that guarantees the GOS requirement is also  $O(N^3)$ .

### 3.3.2.2 Numerical results

Figure 3.5 compares the simulation results reported in [GRF89] with the UAS model results for bursty sources. Average burst length,  $L = 100$  cells. Cell length,  $n_{cell} = 36$  bytes. Buffer size,  $K = 50$  cells ( $100\mu\text{sec}$  delay). Cell loss probability  $P = 10^{-5}$ . The value chosen for  $P$  was much higher than typical industry requirements. This choice was dictated by simulation run-time constraints. In fact, one of the by-products of this research is to show that we can overcome simulation limitations using analytic modeling. To this end, we first validate the analytic model (by comparison with simulation) for  $P = 10^{-5}$ ; then, use the analytic model for  $P = 10^{-9}$ .

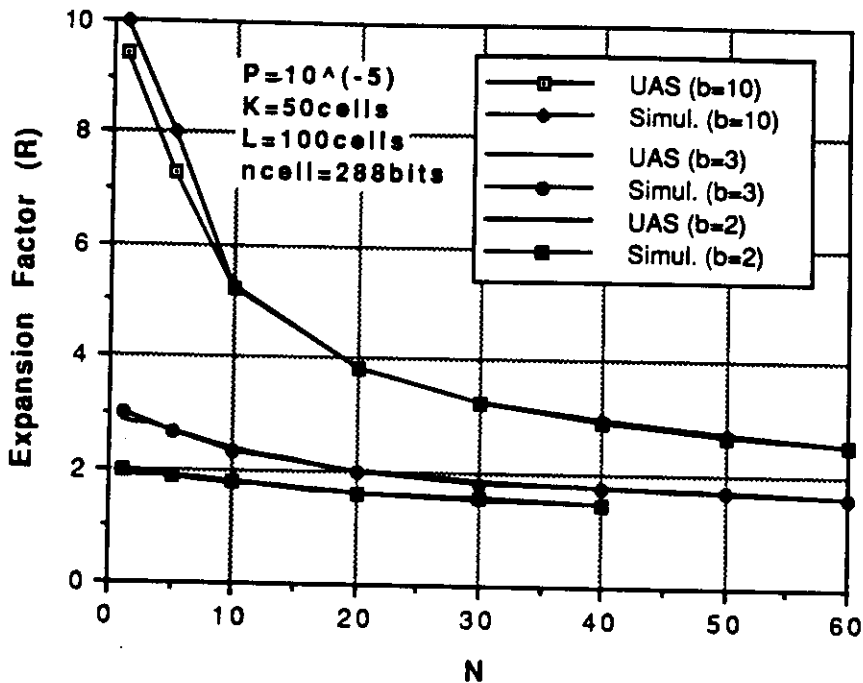


Figure 3.5: Simulation vs. UAS comparison ( $P = 10^{-5}$ ).

Figure 3.5 exhibits an almost perfect match between the simulation and the UAS model (of course, the results from the UAS model can be obtained much faster). Since the assumptions on which the model is based remain valid as  $P$

decreases, we can assume that the model will be accurate also for much smaller values of  $P$ . Thus, we will use the UAS model to perform a sensitivity analysis of the expansion factor with the source and buffer parameters, and more importantly, to obtain results for  $P = 10^{-9}$ . These results cannot be obtained by simulation, because of excessive run-time.

In the following experiments we will use the standard ATM cell length of 53 bytes (48 data bytes + 5 header bytes) [Min89].

Figure 3.6 shows the normalized expansion factor  $R$  for burstiness factors  $b = 2$ ,  $b = 3$ , and  $b = 10$ , where  $NormR(N, b) = R(N, b)/b$ . This normalization was performed so that all curves would start from the same value at  $N = 1$ . From this figure we can observe the effect of burstiness on statistical multiplexing. As expected, the higher the burstiness, the larger the multiplex effect (i.e., the faster the expansion factor drops with the increase of the number of sources). The buffer size ( $K=35$  cells) was chosen in order to keep queueing delay below  $100 \mu\text{sec}$ .

Figures 3.7, 3.8, and 3.9 show the effect of the decrease in  $P$  from  $10^{-5}$  to  $10^{-9}$  on the expansion factor. We note that  $R$  increases as  $P$  decreases, as expected. We also note that the statistical multiplexing effect is less pronounced for  $P = 10^{-9}$  than for  $P = 10^{-5}$ , specially for small burstiness factors.

Next, we study the sensitivity of  $R$  with respect to buffer size,  $K$ , and burst length,  $L$ . Figure 3.10 shows the variation of  $R$  with  $L$  for fixed buffer size ( $K = 35$  cells), for various number of sources. As it can be seen, the expansion factor increase is more significant for burst lengths in the range  $1 \leq L \leq K = 35$  than for larger ones ( $L > K = 35$ ). Note that the horizontal axis is in logarithmic scale. This can be explained by the fact that as long as the average burst length of a single source is shorter than the buffer size, the cell losses will be quite small, because the entire burst can fit into the buffer. When the average burst becomes

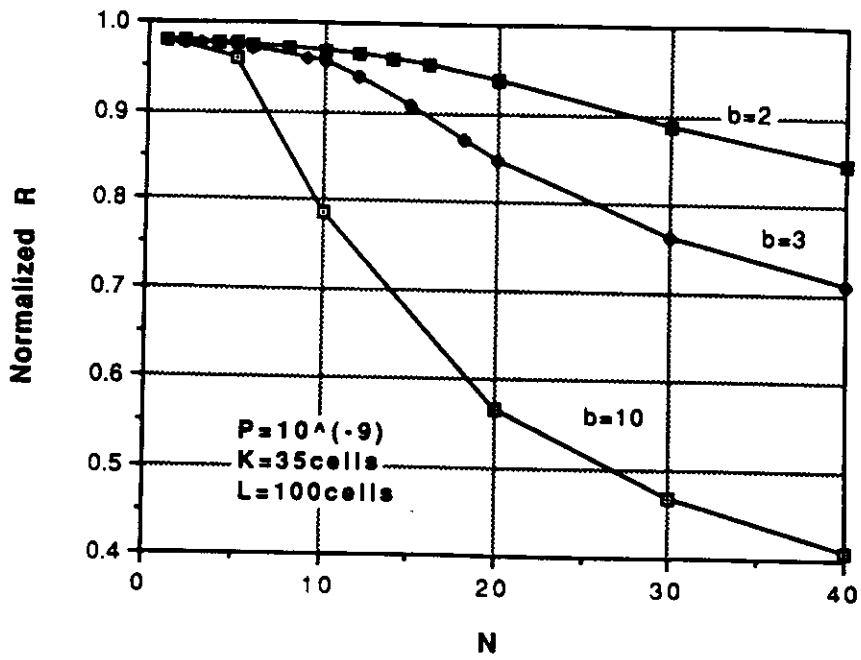


Figure 3.6: Normalized  $R$  ( $P = 10^{-9}$ ).

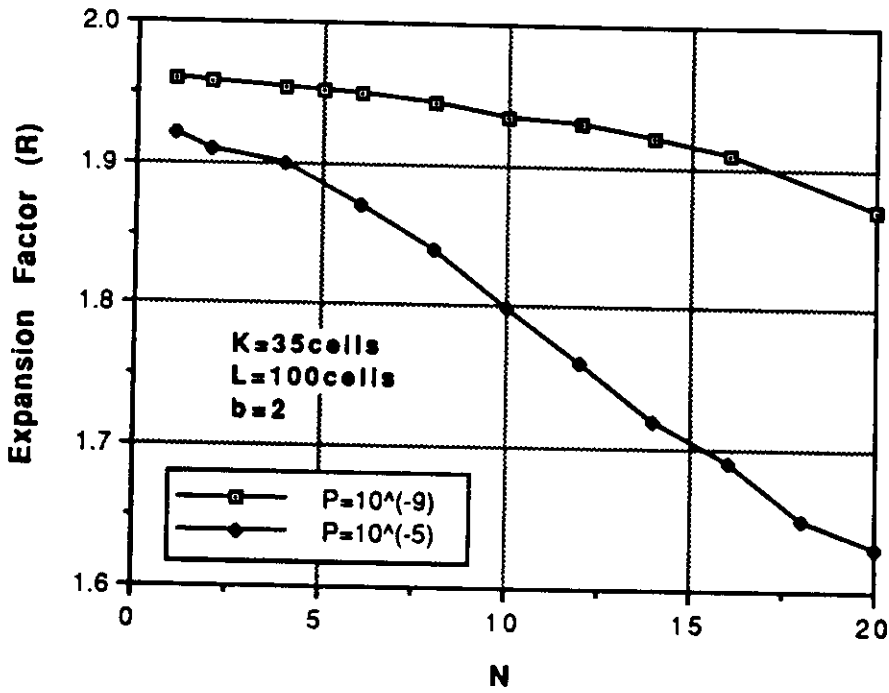


Figure 3.7:  $R$  sensitivity with respect to  $P$  ( $b = 2$ ).

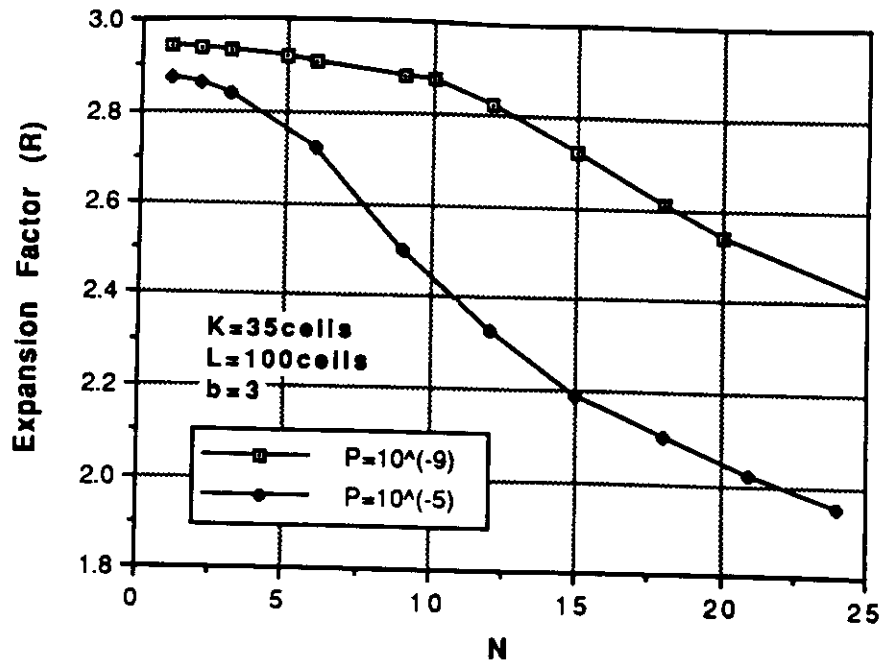


Figure 3.8:  $R$  sensitivity with respect to  $P$  ( $b = 3$ ).

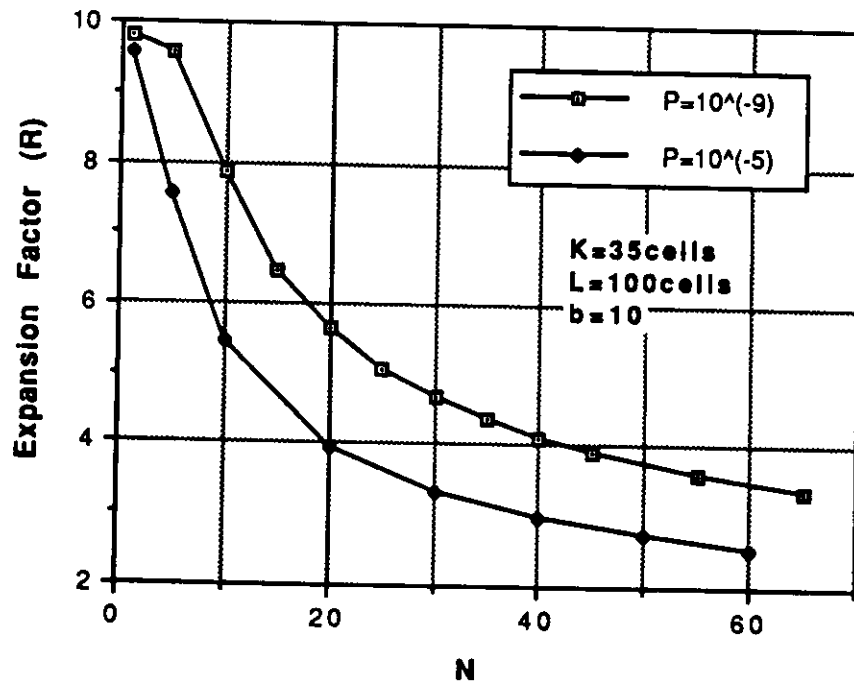


Figure 3.9:  $R$  sensitivity with respect to  $P$  ( $b = 10$ ).

longer than the buffer size, enough bandwidth must be allocated so that the cells are served at the same rate as they come in (to keep loss probability  $P$  bounded).

Thus, the actual burst length is not that critical anymore.

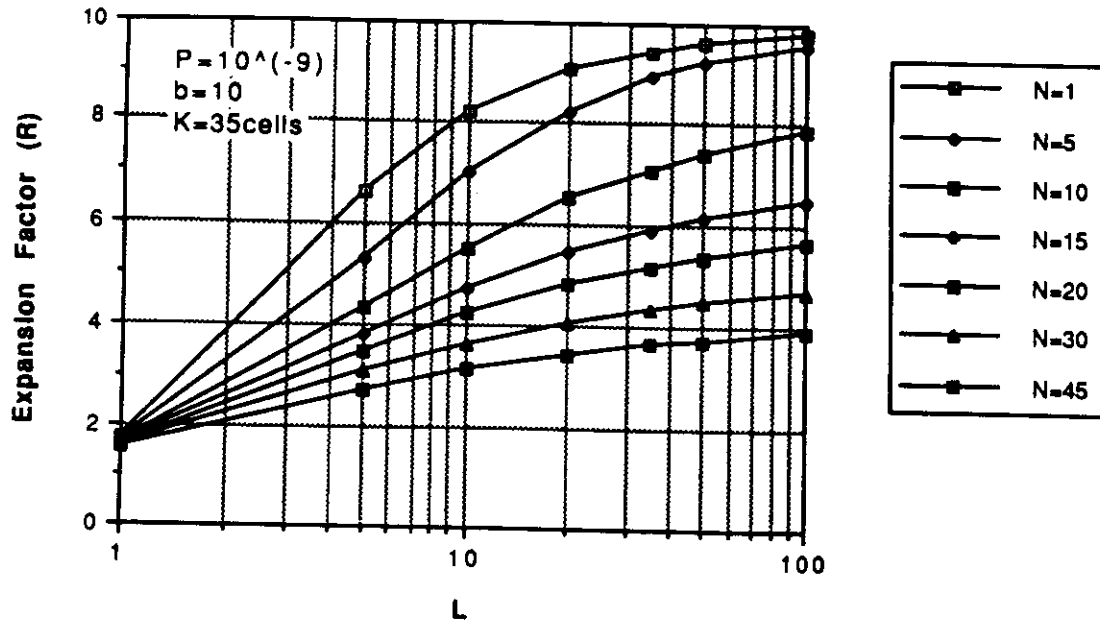


Figure 3.10:  $R$  sensitivity with respect to  $L$ .

As we mentioned before, we observed that  $R$  does not depend on  $K$  and  $L$  independently, but on their ratio. In other words, if we multiply both  $K$  and  $L$  by the same factor,  $R$  remains unchanged. This property first observed by Li [Li89], is formally proved in Appendix C. This result suggests the definition of a parameter  $\beta$  as the ratio between the average burst length of a single source and the buffer size, i.e.,  $\beta = L/K$ . Thus, we do not need to carry out another set of experiments where  $L$  is kept fixed and  $K$  is a variable.

Finally, in Figure 3.11 we show the variation of  $R$  with the parameter  $\alpha$  defined in [DJ88] as the fraction of the average burst length to the buffer capacity per source, i.e.,  $\alpha = NL/K$  or  $\alpha = N\beta$ . A value of  $\alpha \leq 1$  means that the average burst length can fit completely in its "share" of the buffer. Therefore, it is no

surprise to find in Figure 3.11, that the case  $\alpha = .4$  requires a lower expansion factor than the case  $\alpha = 4.0$ . The parameter  $\alpha$  is important in the dimensioning of the buffer size. However, during operation, when buffer size and traffic type are defined,  $\beta$  is a more meaningful parameter.

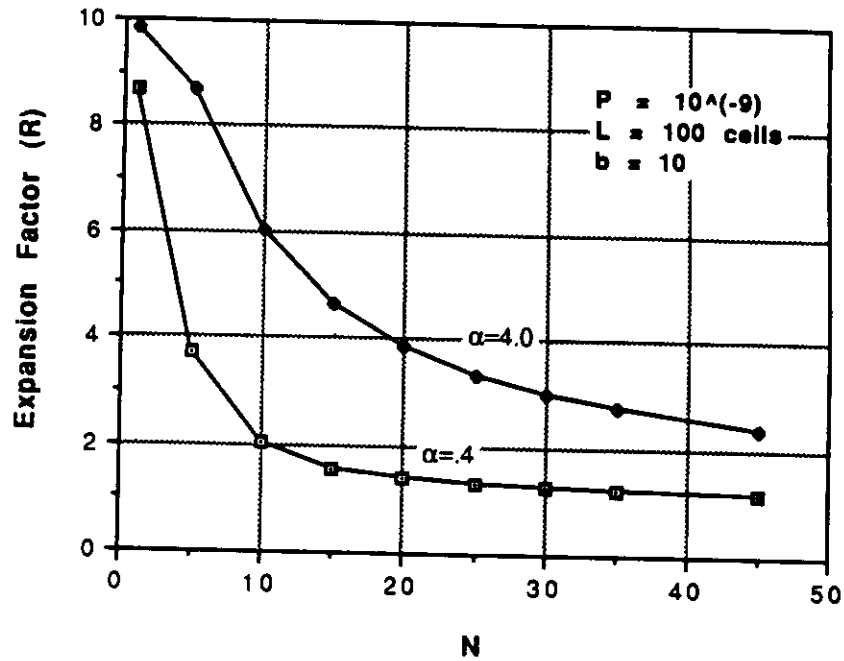


Figure 3.11: R sensitivity with respect to  $\alpha$ .

### 3.3.3 Variable-Bit-Rate Traffic Analysis

Several models have been proposed for the characterization and multiplexing of VBR sources. An excellent survey can be found in [KS89]. In this dissertation, we use the model presented in [MAS<sup>+</sup>87, MAS<sup>+</sup>88]. Namely, the aggregate bit rate for a number of identical videophone sources is modeled as a discrete finite-state, continuous-time Markov process. In this Markov model, the state corresponds to a quantization level of the aggregate bit rate. The quantization step ( $A$ ), the number of states ( $M + 1$ ), and the transition rates are tuned to fit the average variance

and autocovariance function of the measured data (see Appendix D for a summary of the model).

### 3.3.3.1 Solution approach

Maglaris et al. [MAS<sup>+</sup>87, MAS<sup>+</sup>88] solved the multiplexing model above using the fluid-flow approach [AMS82]. More precisely, they observed that the above Markov process is equivalent to a process consisting of  $M$  independent *minisources*, each alternating between sending 0 bits/pixel and  $A$  bits/pixel. We implemented this solution approach to obtain the cell loss probability for a given buffer size.

Similarly to what was done with the bursty traffic, the bandwidth required to achieve a given GOS (cell loss probability), is determined by using a logarithmic interpolation method in the range  $N.B_m < W < N.B_p$ , which corresponds to searching the interval  $1 < P < 0$ . The search ends when the GOS is obtained within a given tolerance.

### 3.3.3.2 Numerical results

First, we are going to validate the results obtained with the analytical (UAS) method with results obtained through simulation for  $P = 10^{-5}$ . The simulation was performed using a continuous-state autoregressive Markov model described in [MAS<sup>+</sup>87, MAS<sup>+</sup>88] which is easy to simulate but difficult to analyze.

In [MAS<sup>+</sup>87, MAS<sup>+</sup>88] they assume that the data generated by a video source during a frame period is stored in a pre-buffer, and at the end of a frame, the collected data is packetized and transferred to the common buffer. From their article one can deduct that the transfer of packets from the pre-buffer to the common buffer is made as a bulk arrival (at infinite speed). This transfer mode is feasible only for large common buffers. This is because 338.5 cells are generated on



average per frame, per source. Therefore, if the buffer length is small, no matter what the multiplexer output rate is, a large portion of cells would always be lost.

The actual transfer mode will depend on the codec used and the capacity of the link that connects the codec to the multiplexer. For simplicity, we will only consider two cases: bulk transfer and continuous transfer. The bulk transfer would be the worst case (higher loss probability), while a continuous transfer during a frame period would be the best case (lower loss probability). The continuous transfer corresponds to the fluid-flow model.

Figure 3.12 compares the results of the analytical model with the simulation results for a bulk and a uniform transfer for  $P = 10^{-5}$  and a buffer size of 10,613 (53 byte) cells which corresponds to 30 ms of transmission at 150 Mb/s speed. In the uniform transfer, the packets (cells) generated in a frame period are uniformly spread over the next frame period. The analytical model exhibits a good agreement for small number of sources (low utilization), and become optimistic for large numbers (higher utilizations).

In Figure 3.13 the results of the analytical model are compared with the simulation results for a uniform transfer for  $P = 10^{-5}$  and a buffer size of 35 (53 byte) cells which corresponds to  $100\mu\text{s}$  of transmission at 150 Mb/s speed. The agreement is very satisfactory.

Now that we validated the analytical model for  $P = 10^{-5}$ , we are going to use the model to study the behavior of our video traffic sources for  $P = 10^{-9}$ . Figure 3.14 shows how the multiplexing is affected by the GOS requirement for 2 different values of buffer size ( $K = .1$  msec, and  $K = 30$  msec). Figure 3.15 shows the sensitivity of  $R$  with respect to buffer size, for  $P = 10^{-9}$ .

Note that even though the required bandwidth is larger when buffers are smaller for the same number of sources, the multiplexing advantage is much more evident.

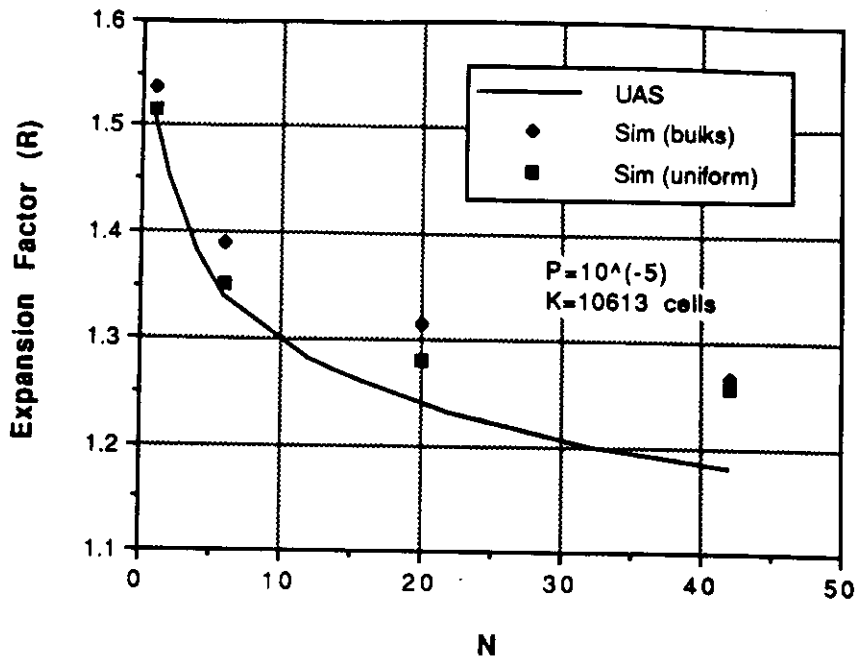


Figure 3.12: Simulation vs. UAS results for VBR sources ( $K=30$ ms).

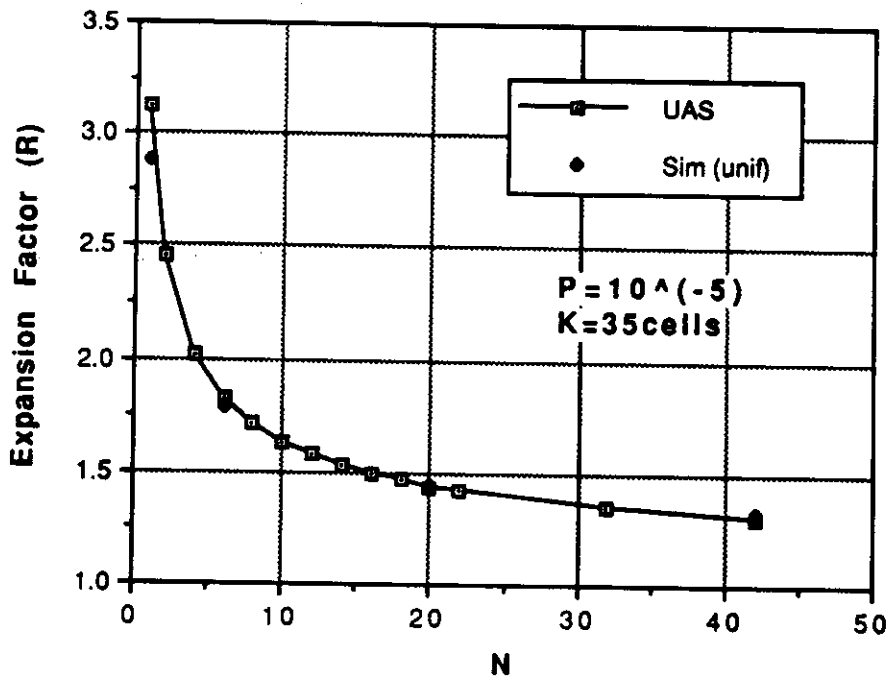


Figure 3.13: Simulation vs. UAS results for VBR sources ( $K=0.1$  ms).

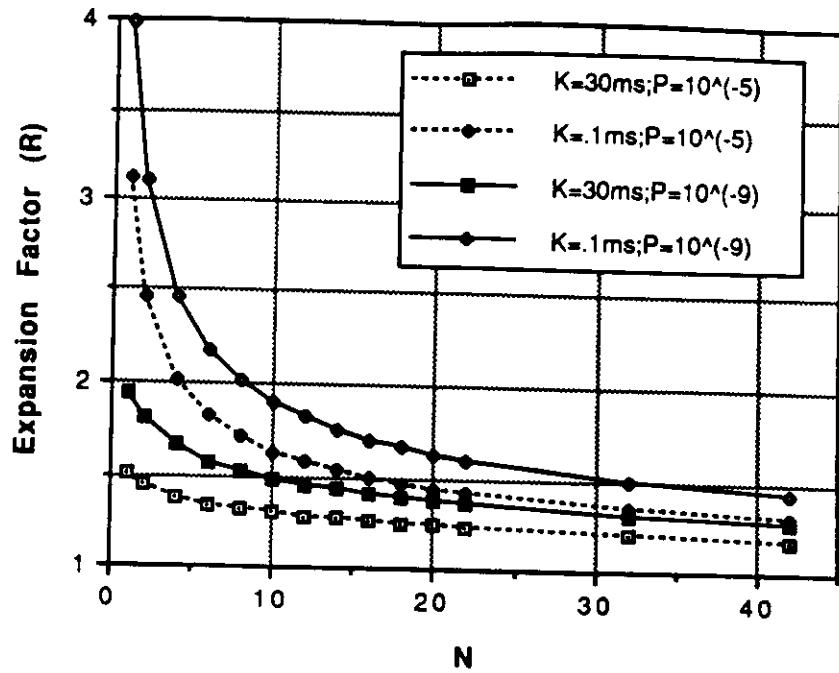


Figure 3.14:  $R$  sensitivity with respect to the GOS requirement ( $P$ ), and buffer size ( $K$ ) for VBR sources.

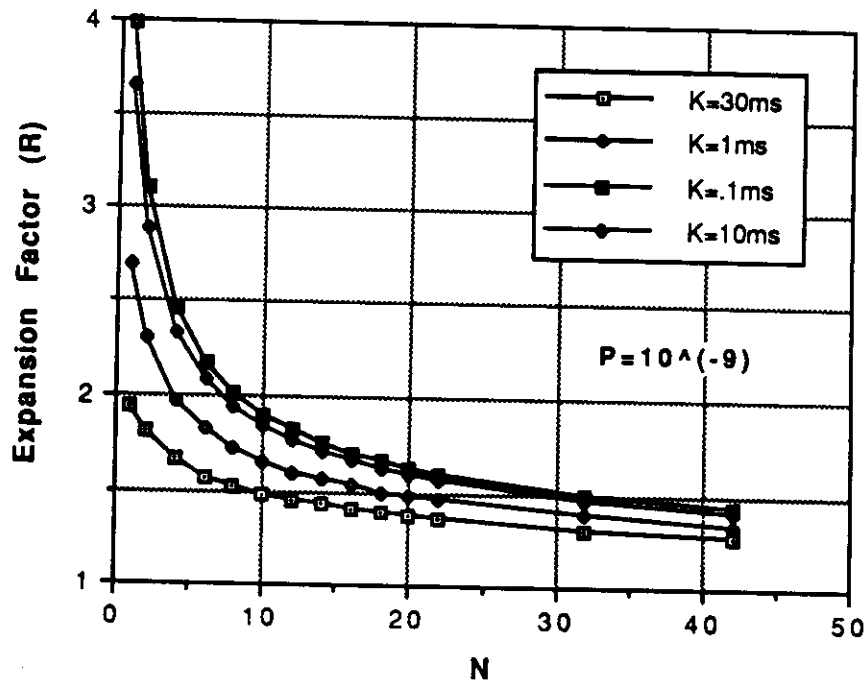


Figure 3.15: Sensitivity of  $R$  with respect to the buffer size ( $K$ ) for VBR sources.

### 3.3.4 Comparison of Bursty and Variable Bit Rate Traffic

Figure 3.16 compares the multiplexing effect for VBR sources with the multiplexing effect for bursty sources of similar burstiness. The VBR sources have a burstiness of  $b_v = B_p^v/B_m^v = 10.575\text{Mbps}/3.9\text{Mbps} = 2.71$ . These sources are compared with bursty sources with burstiness factor 3 for three distinct burst lengths.

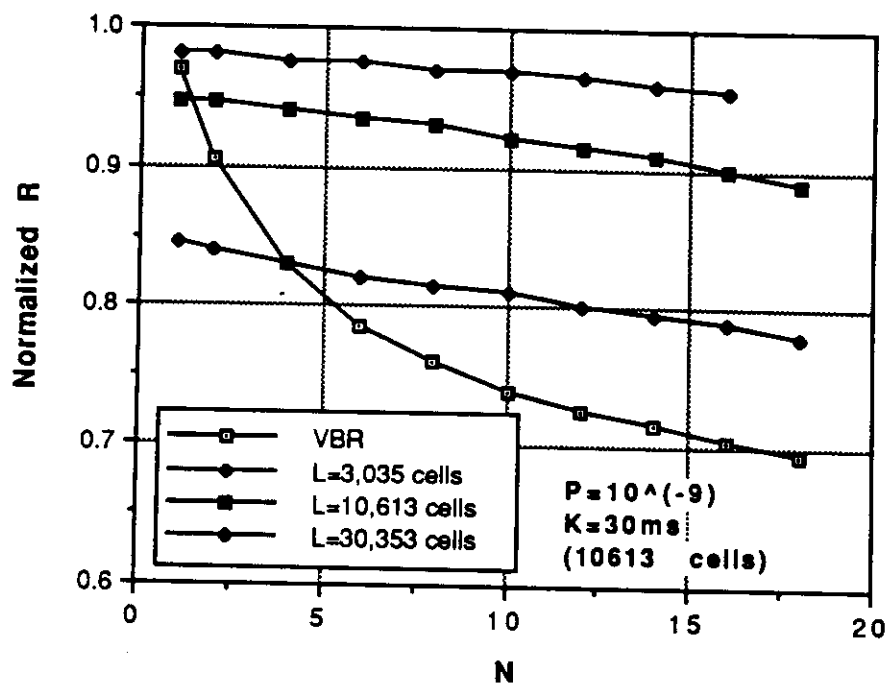


Figure 3.16: Comparison of Bursty and Variable Bit Rate Traffic.

The benefit of statistically multiplexing VBR sources is quite evident. This reduction of the effective bandwidth per source is much better than that for the bursty sources. As a consequence, approximating the multiplexing behavior of VBR sources by the behavior of bursty sources with same burstiness would usually be too pessimistic. However, it can give us a lower bound in the number of sources that can be multiplexed in a link of a given capacity.

### 3.4 Admission Control

Admission control is the mechanism that decides whether or not a new connection can be admitted to the network. Several criteria can be used. The basic one is to assure that the connection can be accepted without compromising its required GOS, and the GOS of the already established connections. We could also require a minimum available bandwidth for certain traffic services in order to assure fairness. In this Section we will limit ourselves to the above basic criteria.

The results obtained in the previous sections can be used to devise call admission policies. If the incoming traffic is homogeneous, we can readily obtain the maximum number of sources  $x_i^{max}$  of type  $i$  that can be multiplexed in a channel of capacity  $C$  yet satisfying the GOS requirement from the following equation:

$$x_i^{max} = \frac{C}{R_i(x_i^{max})B_m^i}.$$

However, ATM networks will be used for a variety of services, so in general the traffic submitted to a channel is not homogeneous. In order to obtain the region of feasible mix of non-homogeneous traffic, Akhtar [Akh87] proposed a linear approximation in which each source is assumed to require the same bandwidth as it requires when all the channel is occupied by sources of its same type, i.e.,

$$V_i = R_i(x_i^{max})B_m^i = \frac{C}{x_i^{max}},$$

where,  $V_i$  is the effective bandwidth per each source of type  $i$ .

Simulation and analytical results shows that this allocation is too optimistic [DCLM89]. That is, the effective bandwidth per source is underestimated, and therefore, if all channel capacity is allocated, the GOS requirement will not be satisfied. Decina and Toniatti [DT90] suggest the use of a quasi linear approximation with a security coefficient.

A pessimistic bandwidth allocation would be to consider the multiplexing effect only within each subset (i.e., we evaluate the bandwidth required by each subset independently and then we add up the bandwidth requirements). This is obviously an upper-bound since we do not take into account the multiplexing of traffic across subsets of sources with different characteristics.

Gallassi et al. [GRF89] proposed a better upper-bound, called *class related rule* (CRR) (see also [DTV90]), which is the minimum between the above (pessimistic) upper-bound and the required bandwidth if the total average traffic was generated by the sources with the largest burstiness.

Dziong et al. [DCLM89] proposed a non-linear approximation which exhibits a good matching with results using the MMPP model for both homogeneous and heterogeneous bursty sources.

Figure 3.17 presents the results obtained by simulation of the statistical multiplexing of bursty and VBR sources. The VBR source characteristics were described earlier. The bursty sources have burstiness factor  $b = 10$ , peak bit rate  $B_p = 10\text{Mb/s}$ , and burst length  $L = 100$  cells. For the analytical and approximation results, we assume that the number of sources is a continuous variable. While in the simulation they can only assume discrete values. In Figure 3.17 we show the feasible  $(N_1, N_2)$  mixes obtained with the various models. Note that the upperbound on required bandwidth becomes in the Figure a lower bound on number of terminals supported.

By comparison with simulation results, we note that the linear approximation is indeed optimistic; the upper-bound is too pessimistic; while the non-linear approximation proposed by Dziong et al. [DCLM89] exhibits a good match. More extensive work is required in order to determine whether Dziong's approximation is adequate, or a better one must be sought.

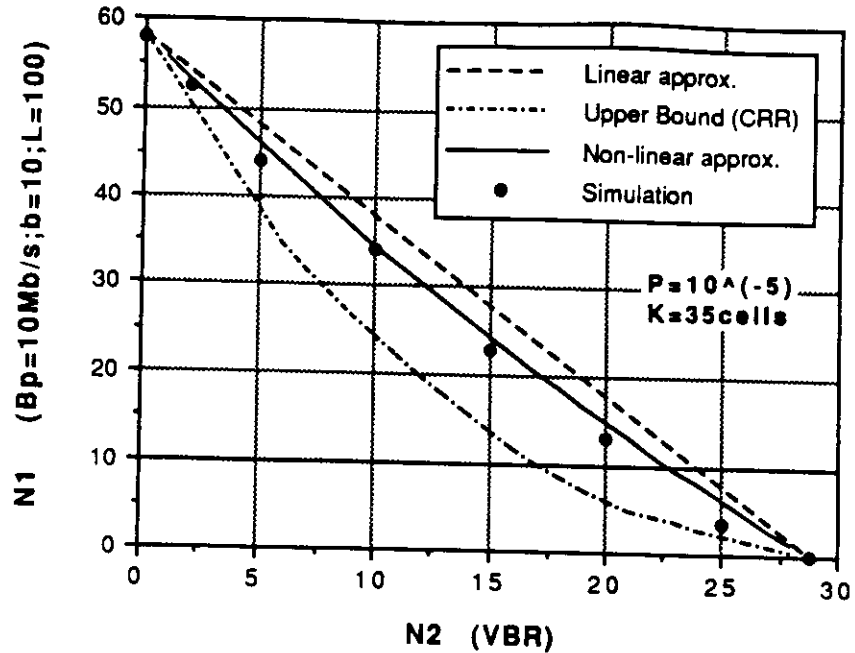


Figure 3.17: Mixing of Bursty and Variable Bit Rate Traffic.

### 3.5 Traffic Mixing Trade-offs

In this Section, we want to investigate the trade-offs involved in the mixing of heterogeneous traffic sources.

#### 3.5.1 To integrate or not to integrate, that is the question

Even using integrated switches, the question arises on whether or not should we integrate different traffic classes in the same channel.

If we look at the mixing of two traffic classes it seems that the best case occurs when the traffic classes are as similar as possible (i.e., linear approximation is valid). And the worst case seems to be when one of the traffic is of a constant (periodic) bit-rate (CBR) traffic (i.e., its average and peak bit rate are the same).

The presence of the constant bit-rate traffic would have the effect of reducing the channel bandwidth available to the other traffic by the amount allocated to

the CBR traffic. Since the expansion factor ( $R$ ) is a non-increasing function of the number of sources ( $N$ ), for a reduced total bandwidth, each source would require at least the same amount of bandwidth as if they were alone in the channel.

In order to test these hypothesis, we simulated the mix of either bursty or VBR sources with a CBR source with several bit rates. For each simulated CBR bit rate, we found (through simulation) the maximum number of bursty or VBR sources that mixed with the CBR traffic still keeps the total average cell loss probability below  $10^{-5}$ .

Figure 3.18 compares the maximum number of bursty sources that can be mixed with a CBR of increasing bit rates (CBR bit rates are expressed in 10 Mbps "units") obtained through simulation with a linear approximation, and the allocation assuming an independent mix (i.e., no multiplexing among different class sources). As we can see, for CBR traffic below 50Mbps, the maximum number of our bursty source that can be multiplexed is higher than the value obtained by using a linear approximation. This can be explained by the fact that for small CBR bit rates, the periodic traffic does not interfere as much with the bursty one. Furthermore, our bursty traffic has a relatively high burstiness ( $b = 10$ ), which reinforces the non interference with the CBR traffic. As the CBR bit rate increases, the available bandwidth for bursty traffic is also reduced, and therefore, the simulation approaches the independent mix curve.

Figure 3.19 makes a similar comparison for the mix of our VBR sources with CBR traffic. This time the simulation results are even below the independent mix curve. This is probably due to errors from the analytical model. The simulation curve does not go above the linear approximation curve because of the more predictable traffic of our VBR source ( $b = 2.71$ ).

From these considerations we conclude that as far as total required bandwidth



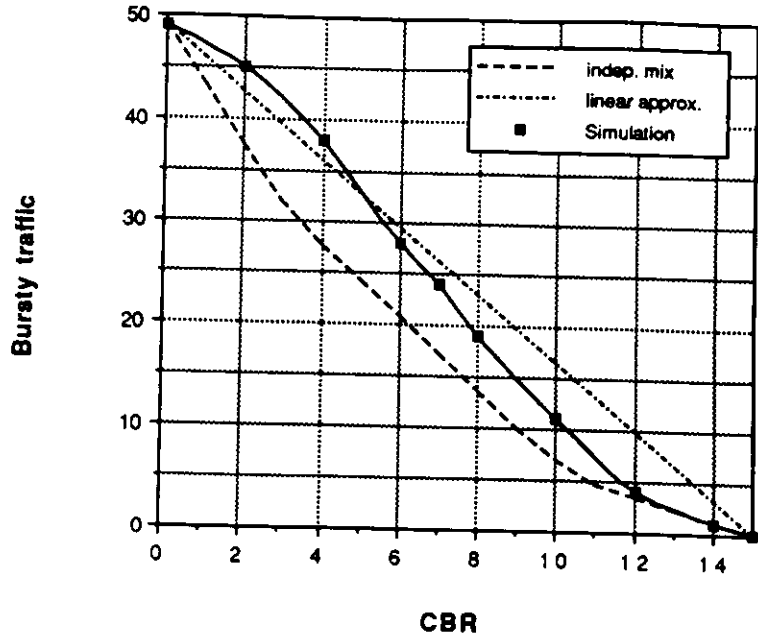


Figure 3.18: Mix of Bursty and CBR Sources ( $GOS=10^{-5}$ ).

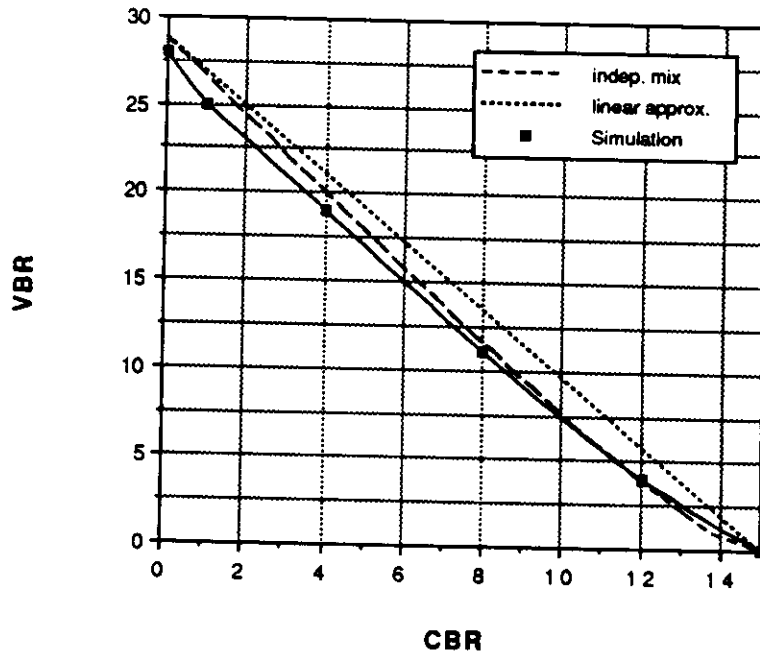


Figure 3.19: Mix of VBR and CBR Sources ( $GOS=10^{-5}$ ).

is concerned, usually we do not gain anything with integration of different traffic classes in the same channel.

However, we may want to integrate traffic in cases where there would still be some spare capacity in the channel. This can happen because an integral number of sources does not fill the channel completely, or just because there are just not enough sources to fill the channel. This later case is particularly true for local access channels.

Even if we assume that fibers come for free, we still have the termination and switch costs, what favors a reduced number of fibers. Besides, if during operation we keep the number of busy channels to a minimum, reconfiguration will be facilitated.

### **3.6 Multiplexers in Tandem and Bandwidth Allocation**

In Section 3.3 we studied the statistical multiplexing gain in ATM networks for homogeneous and heterogeneous traffic sources.

We obtained the bandwidth required by a number of sources so that the desired Grade of Service (GOS) (i.e., cell loss probability) is achieved. Or alternatively, we can obtain the maximum number of homogeneous sources ( $N_{max}$ ) that can be multiplexed in a given multiplexer, still achieving the desired Grade of Service (GOS). This knowledge can be used by the admission control mechanism in order to decide whether or not a new call can be accepted.

These results were obtained for a single stage multiplexer. A different behavior is expected for the multiplexing at intermediate multiplexers. In this Section we study the statistical multiplexing of bursty traffic in multiplexers in tandem, and show that an admission control based on the required bandwidth at the first stage multiplexer is as conservative as the aggregate traffic follows a single path.

Throughout this Section we are assuming that all buffers are of the same size ( $K$ ), that the output rate ( $W$ ) is 150 Mbps, and that the cell length ( $n_{cell}$ ) is 53 bytes.

The Section is organized as follows: Subsection 3.6.1 studies the output process for a single multiplexer. The subsequent sections consider in turn, each of the possible merging/splitting schemes for two stages of multiplexers. Subsection 3.6.2 considers two multiplexers in series with no merging of traffic in the second stage multiplexer, while in Subsection 3.6.3 we allow the merging of traffic in the second stage multiplexer. Subsection 3.6.4 studies the case where the output traffic from a first stage multiplexer is split through several second stage multiplexers, with the merging of new traffic at the tagged second stage multiplexer. Finally, Subsection 3.6.5 considers the more general case where a tagged second stage multiplexer is fed by fraction of the output of several first stage multiplexers and new traffic sources.

### 3.6.1 Output Process for a Single Multiplexer

In this Subsection we study the output process for a single multiplexer. A single multiplexer can be modeled as a finite size queue with general arrivals and deterministic service time ( $G/D/1/K$ ).

Tran-Gia and Ahmadi [TGA88] solved a discrete-time finite size queue with deterministic service time and batch arrivals with general batch-size distribution ( $G^{[X]}/D/1/K$  queueing system). They use a numeric algorithm based on the Fast Fourier Transform to obtain the cell loss probability.

Louvion, Boyer, and Gravey [LBG88] solve analytically for the queue length distribution and loss probability of a discrete-time single server queue with Bernoulli arrivals and deterministic service time queue with finite or infinite buffer sizes

(Geo/D/1 and Geo/D/1/K, respectively).

However, we didn't find any references to the output process statistics of such systems.

Generally speaking, no matter what the input traffic statistics are, the server will generate (i.e. serve) cells at a constant rate while the queue is not empty, and will transmit cells at the input rate whenever the queue is empty. Therefore, in order to characterize the output process of a single queue we would have to know what is the distribution of the "busy" period of such a queue, defined as the amount of time in which the queue is not empty (see Figure 3.20).

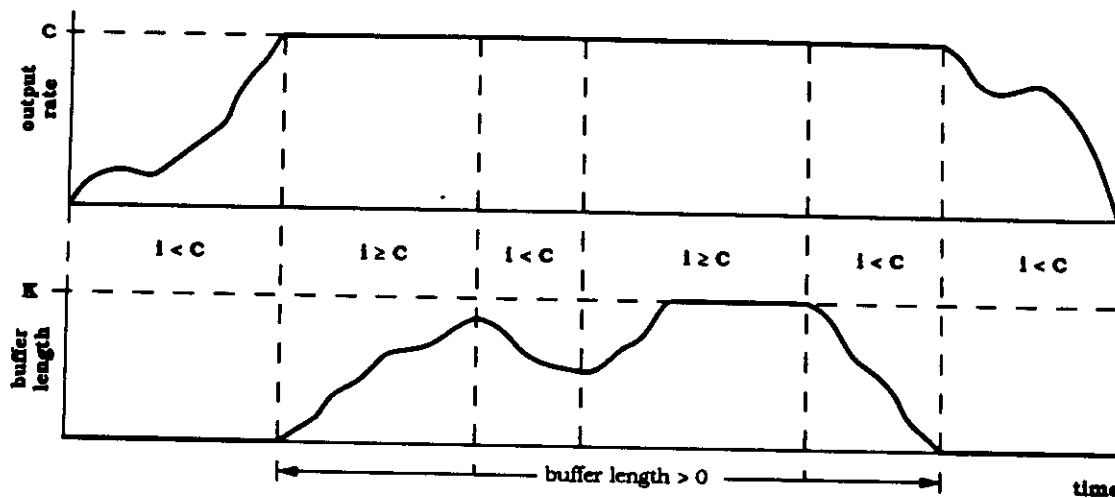


Figure 3.20: Multiplexer Busy Period.

The queue length increases whenever the aggregate input bit rate is larger than the output bit rate. It remains constant whenever the aggregate input bit rate is equal to the output bit rate. And, the queue length decreases whenever the aggregate input bit rate is smaller than the output bit rate.

Once the queue gets larger than 0, it can increase and decrease for a number of times before it gets empty again.

Since we are dealing with a finite queue, its length should always be in the range

from 0 to its maximum ( $K$ ). Losses in the multiplexer would occur, whenever the queue is at its maximum and the aggregate input bit rate is larger than the output bit rate. Obviously, this event must be kept as rare as possible in order to achieve the desired GOS.

### 3.6.1.1 Bursty Input Traffic

Bursty sources alternate between an “on” state, where they transmit at peak bit rate ( $B_p$ ), and an “off” state, where they keep silent. In the homogeneous case, let  $i$  be the number of active bursty sources, and  $C = W/B_p$  be the output bit rate relative to the peak bit rate of a single source (i.e.,  $C$  gives the maximum number of sources that can be simultaneously active, without increasing the queue length). Therefore, if  $i > C$  the queue length will increase unless the queue is at its maximum; if  $i = C$  the queue length will keep constant; and if  $i < C$  the queue will decrease unless it is already empty.

Even if we don’t solve exactly for the multiplexer output process, in the next sections we will use approximations based on the queue length distribution and probability of having  $i$  active sources.

Table 3.1 shows several parameters obtained using the UAS model with finite buffer size [Tuc88] for  $GOS = 10^{-5}$ ,  $W = 150$  Mbps, and homogeneous bursty sources with the following parameters:  $B_p = 10$  Mbps,  $b = 10$ , and  $L = 100$ .

Here,  $K$  is the buffer size;  $N_{max}$  is the maximum number of our bursty sources that can be multiplexed in a multiplexer with buffer size  $K$  and output rate  $W$ ;  $P$  (UAS) is the base 10 logarithm of the cell loss probability for the given MUX parameters and number of sources, using the UAS model, while  $P$  (Sim.) was obtained through simulation (95% confidence intervals);  $F(0)$  is the probability that the queue length is 0;  $\bar{Q}$  is the MUX average buffer size length;  $D_{max}$  is the

Table 3.1: Multiplexer Parameters for Bursty Traffic ( $GOS = 10^{-5}$ ).

| $K$  | $N_{max}$ | $P$ (UAS) | $P$ (Sim.)      | $F(0)$ | $\bar{Q}$ | $D_{max}$ | $\bar{D}$       |
|------|-----------|-----------|-----------------|--------|-----------|-----------|-----------------|
| 35   | 58        | -5.09     | $-5.08 \pm .12$ | .99946 | .002      | .099      | 56.5 nsec       |
| 350  | 77        | -5.02     | N/A             | .98852 | .274      | .989      | 774.5 nsec      |
| 3500 | 124       | -5.04     | $-5.06 \pm .17$ | .58712 | 103.193   | 9.890     | 291.7 $\mu$ sec |

maximum cell delay (in msec) for a MUX of buffer size  $K$  and output rate  $W$ , in mili-seconds; and,  $\bar{D}$  is the average cell delay for a MUX of buffer size  $K$  and output rate  $W$ .

A couple of comments. First, in order to achieve a low cell loss probability, as is the case in ATM networks, the average buffer size has to be well below its maximum, which, for short buffers, translates in having the buffers much of the time empty. Note that the results of Table 3.1 are for  $GOS = 10^{-5}$ ; therefore,  $F(0)$ ,  $\bar{Q}$ , and  $\bar{D}$  will be even smaller for  $GOS = 10^{-9}$ . Second, larger buffer sizes imply larger delays which can be intolerable for certain applications.

### 3.6.2 Multiplexers in Series

Figure 3.21 shows two multiplexers in series (tandem) where  $N_{max}$  sources are offered to the first stage multiplexer and the second stage multiplexer is fed only with the output of the first one.

In this case, since both MUX are identical, it is easy to see that no queue is formed at the second stage multiplexer since the input rate of the second queue will be always lower or equal to the output rate of the first one. This result is a particular case of Rubin's [Rub75] which states that "the distribution of the overall message waiting time in the path is equal to that obtained by presenting all the

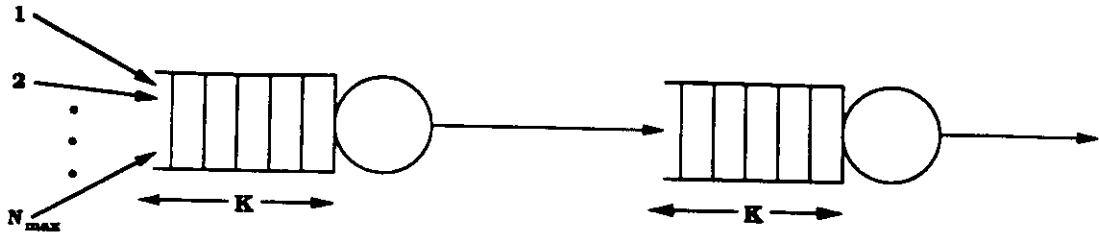


Figure 3.21: Multiplexers in Series.

messages to the channel with lowest capacity only.”

### 3.6.3 Multiplexers in Series with Merging Sources at Second Stage

Now in order to take advantage of the buffer available at the second stage, we allow that extra sources join the second stage multiplexer besides the output of the first stage multiplexer (see Figure 3.22). Let us call  $M$  this number of extra sources.

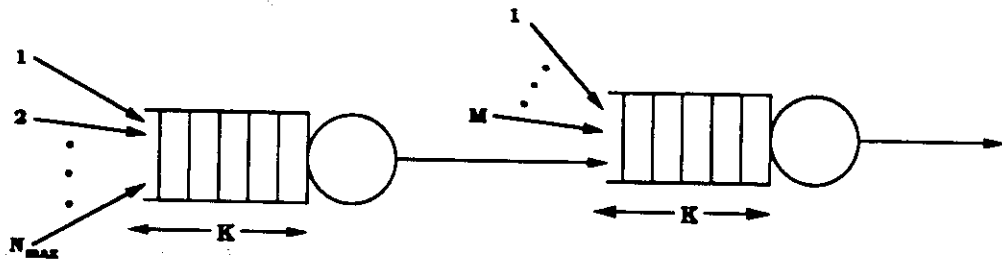


Figure 3.22: Multiplexers in Series with Merging Sources at Second Stage.

For multiplexers with short buffer lengths, practically no queue is formed (see Table 3.1) and therefore, most of the cells are transmitted by the first stage MUX as they are received. Thus, it is almost as if the aggregate traffic of the  $N_{max}$  sources offered to the first stage MUX was offered directly to the second stage MUX. In this case  $M \approx 0$ . Obviously, if  $M = 0$  all losses would occur only at the first stage MUX.

For multiplexers with  $\bar{Q} > 1$ , the number  $M$  of extra sources that joins the second stage multiplexer must be such that the same average queue length is maintained at the second stage MUX.

We are going to estimate  $M$  by

$$M \geq N_{max} - \left( \left[ \sum_{i=1}^C i \cdot b_i + C \sum_{i=C+1}^{N_{max}} b_i \right] \right), \quad (3.2)$$

where  $b_i$  is the probability that  $i$  bursty sources are simultaneously active and it is given by

$$b_i = \frac{\binom{\lambda/\mu}{i} \binom{N}{i}}{(1 + \lambda/\mu)^N}. \quad (3.3)$$

The second term on the right hand side of Equation 3.2 corresponds to a normalized approximate average output bit rate from the first stage multiplexer. It is approximate because it assumes that the second stage multiplexer transmits at maximum speed only if  $i \geq C$ , and also because it does not take into account the fact that the output process from a first stage multiplexer is less burstier than an equivalent number of bursty sources. While the first assumption underestimates the average output bit rate of the first stage multiplexer, and therefore, underestimates the equivalent number of bursty sources, the second assumption overestimates it since cells arrive sequentially and not in bulks as it may happen with independent sources.

Table 3.2 reports the number of extra sources ( $M$ ) obtained through the approximation given by Equation 3.2 and results obtained through simulation (95% confidence interval).

As we can see, our approximation is very conservative.



Table 3.2: Number of Extra Sources ( $M$ ) for Multiplexers in Series with Merging.

| $K$  | $N_{max}$ | $M$ (approx.) | $M$ (Sim.) | $\log_{10} P$ (Sim.) |
|------|-----------|---------------|------------|----------------------|
| 35   | 58        | 0             | 4          | $-4.93 \pm .12$      |
| 3500 | 124       | 4             | 7          | $-5.26 \pm .40$      |

### 3.6.4 Multiplexers in Series with Several Destinations and Merging

In this Section we study the case where the output traffic from a first stage multiplexer is split through several second stage multiplexers, with merging of new traffic at the tagged second stage multiplexer (see Figure 3.23).

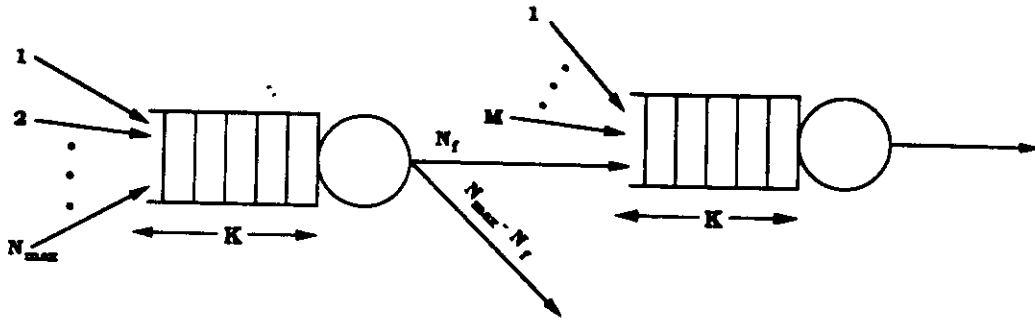


Figure 3.23: Multiplexers in Series with Several Destinations and Merging.

Let us call  $N_f$  the number of sources from the first stage MUX that go to our tagged second stage MUX. We are going to estimate  $M$  by

$$M \geq N_{max} - N_f. \quad (3.4)$$

Table 3.3 reports the number of extra sources ( $M$ ) obtained through the approximation given by Equation 3.4 and results obtained through simulation (95% confidence interval).

A reduction in the number of sources from the first stage MUX that are directed to the second stage one, have as a consequence the increase of the burstiness of

Table 3.3: Number of Extra Sources ( $M$ ) for Multiplexers in Series with Several Destinations and Merging.

| $K$  | $N_{max}$ | $N_f$ | $M$ (approx.) | $M$ (Sim.) | $\log_{10} P$ (Sim.) |
|------|-----------|-------|---------------|------------|----------------------|
| 35   | 58        | 29    | 29            | 29         | $-5.04 \pm .14$      |
| 3500 | 124       | 62    | 62            | 63         | $-5.08 \pm .23$      |

the traffic destined to our tagged second stage MUX. Therefore, we are basically restricted to have only  $N_{max}$  sources in the second stage MUX.

### 3.6.5 Multiplexers Fed by other Multiplexers with Several Destinations

Finally, this Subsection considers the more general case where a tagged second stage multiplexer is fed by fraction of the output of several first stage multiplexers and new traffic sources (see Figure 3.24).

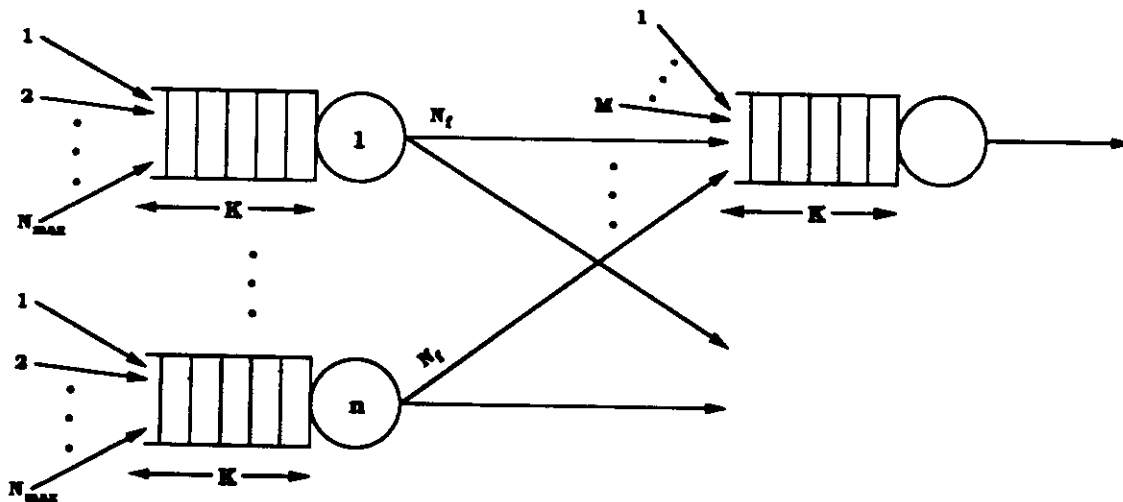


Figure 3.24: Multiplexers Fed by other Multiplexers with Several Destinations.

Again, calling  $N_f$  the number of sources from any of the first stage MUX that

go to our tagged second stage MUX, the number  $n$  of first stage multiplexers that can feed our tagged second stage multiplexer can be estimated by

$$n = \lfloor N_{max}/N_f \rfloor. \quad (3.5)$$

Furthermore, we can estimate  $M$  by

$$M = N_{max} - n \cdot N_f. \quad (3.6)$$

Table 3.4 reports the number of multiplexers obtained through Equation 3.5, the number of extra sources ( $M$ ) obtained through the approximation given by Equation 3.6 and results obtained through simulation (95% confidence interval).

Table 3.4: Number of First Stage Multiplexers ( $n$ ) and Extra Sources ( $M$ ) for Multiplexers Fed by other Multiplexers with Several Destinations.

| $K$  | $N_{max}$ | $N_f$ | $n$ | $M$ (approx.) | $M$ (Sim.) | $\log_{10} P$ (Sim.) |
|------|-----------|-------|-----|---------------|------------|----------------------|
| 35   | 58        | 29    | 2   | 0             | 0          | $-5.08 \pm .08$      |
| 3500 | 124       | 62    | 2   | 0             | 2          | $-5.03 \pm .38$      |

The more we split the traffic from first stage MUXes to second stage ones, the more the traffic offered to the second stage MUX looks like the traffic from independent sources. Therefore, it is no surprise that we are basically restricted to have only  $N_{max}$  sources in the second stage MUX.

### 3.7 Summary

In this Chapter we studied the statistical multiplexing gain in ATM networks for homogeneous and heterogeneous traffic sources. More specifically, we studied the required bandwidth, for bursty sources as well as variable bit rate sources in

a finite buffer multiplexer in order to achieve a given Grade Of Service (GOS), expressed by the cell loss probability.

Using the Uniform Arrival and Service (UAS) model we performed a sensitivity study for bursty traffic and for variable bit rate traffic. The required bandwidth for bursty sources was shown to depend on burst and buffer length only through their ratio. The required bandwidth for variable bit rate sources was compared with the required bandwidth for bursty sources. We concluded that the multiplexing gain for VBR sources is much more pronounced than the multiplexing gain for burst sources with similar burstiness factor.

We also studied the behavior of a mixture of bursty and variable bit rate sources through simulation, and compared it with earlier proposed approximations. The linear approximation was shown to be too optimistic, while the upper bound is too pessimistic. A non-linear approximation proposed elsewhere was shown to produce a good match. We conclude that more work is required in order to validate the above mentioned approximation and; to decide if it is adequate as a basis for admission control.

Finally, we studied the statistical multiplexing of bursty traffic in multiplexers in tandem, and showed that if we use an admission control based on the required bandwidth at the first stage multiplexer, we are being as conservative as the aggregate traffic follows a single path.

Further work is necessary in these areas: models for the traffic sources (in particular, relaxing the assumption that the active and silence periods for a bursty source are exponentially distributed); analytical evaluation of cell loss for mixed traffic, and; better approximations for the allowed number of sources in mixed traffic.

## Chapter 4

### Input Rate Control Mechanisms

#### 4.1 Introduction

In Chapter 3, we studied the statistical multiplexing gain in ATM networks for homogeneous and heterogeneous traffic sources at the cell level. Using the Uniform Arrival and Service (UAS) model [AMS82, DL86, Tuc88, MAS+87, MAS+88, DJ88], we obtained the bandwidth required to achieve a given Grade Of Service (GOS) (i.e., cell loss probability) for bursty sources as well as variable bit rate sources.

The UAS model can predict the effective bandwidth required by the addition of a single bursty or variable bit-rate source. This knowledge can be used by the admission control mechanism in order to decide whether or not a new call can be accepted, still guaranteeing the desired cell loss probability for all connections that share one or more links with the new call.

Evidently, the desired cell loss probability will be achieved only if the sources abide to the parameters specified at call set-up. Therefore, the network must implement some special mechanisms, called *Input Rate Control* (IRC) (also known as sentinel function, usage monitoring, bandwidth enforcement, or policing mechanisms), to assure that the sources abide by their specification. This policing function can be *repressive* if performed by the network, or *preventive* if performed by the terminal. Even if preventive policing is used, the network still needs to perform a policing function on its side of the user network interface (UNI) in order

to protect well behaved sources from dishonest ones. In this Chapter we will be studying the former.

Violating cells (i.e., cells that does not conform to the traffic specification) may be summarily deleted or they can be just marked as such and eventually be deleted by the network in case of congestion. The idea behind the latter approach is that not all violating cells will cause congestion. Therefore, there is a good chance that marked cells will make it through the network. Obviously, this scheme requires the use of cell priorities inside the network (i.e., marked cells are given lower priority than unmarked ones). The latter approach allows for the provision of *pre-marking* by the sources [JLL89].

In this Chapter we present some of the policing mechanisms proposed in the literature, as well as three new mechanisms called *delta*, *delta-2*, and *delta-3*. The performance of these mechanisms are studied through simulation and compared.

We conclude that the delta-2 mechanism, even though not conforming to the ideal marking probability behavior, nor being the simplest one, has the nice property of penalizing abusive traffic sources, by marking more than the abusive cells for large offending factors. As a result, the cell loss probability of well-behaved sources may even be reduced below the desired levels.

Section 4.2 introduces what we understand by an Ideal IRC Mechanism and how we will compare the proposed mechanisms with this ideal one. In Section 4.3 we present some of the policing mechanisms that have been proposed in the literature, namely, credit counter, jumping and moving windows, and leaky bucket, as well as the new mechanisms. We analyze the behavior of the IRC mechanisms when offered a bursty traffic in Section 4.4, while their behavior under VBR traffic is analyzed in Section 4.5. Based on these analyses we compare the IRC mechanisms in Section 4.6. Finally, we conclude in Section 4.6.5.

## 4.2 The Ideal IRC Mechanism

The ideal input rate control mechanism is one that marks/deletes all and only the violating cells (i.e., cells in excess of the average) [Rat90].

Therefore, an ideal policing mechanism would not mark nor delete any cell from sources with average rates up to the nominal, and would mark half the cells if the average rate was twice the nominal, two thirds of the cells if the average rate was three times the nominal, and so on. In general, a fraction  $(\sigma - 1)/\sigma$  of the cells would have to be marked or deleted, where  $\sigma$  is the normalized average rate (see Fig. 4.1).

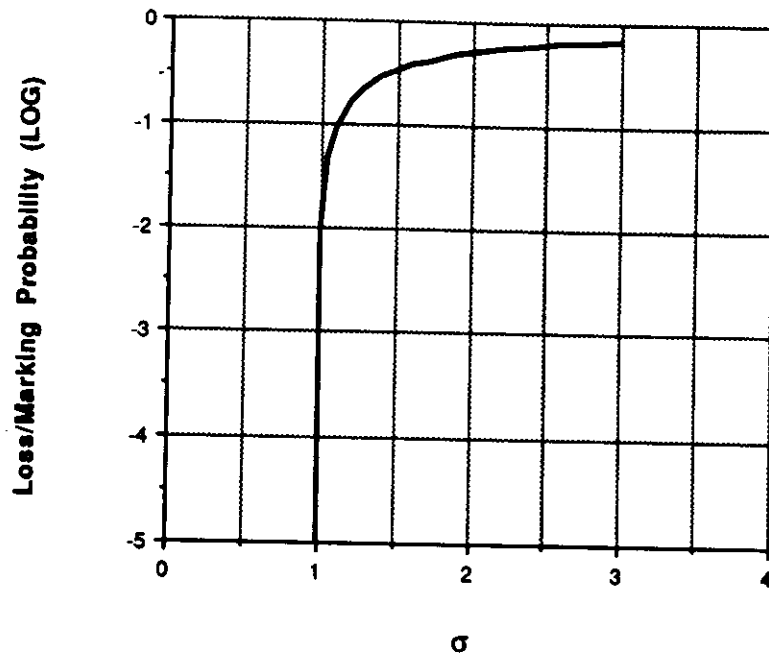


Figure 4.1: Ideal Policing Mechanism Behavior.

One of the criteria that will be used to measure the performance (quality) of a policing mechanism is its conformance to the ideal cell loss/marketing probability behavior (i.e., the ability to mark/delete all and only the cells in excess of the

average).

As we will see, not all the proposed input rate control mechanisms are appropriate for deleting abusive cells. They would either delete too many normal cells or let go too many abusive ones. In the other hand, due to the multiplexing effect, the multiplexer (and ultimately, the network) might be able to carry a percentage of the abusive cells without deteriorating the traffic of well behaved sources; increasing, therefore, network throughput. This suggests that marking cells instead of deleting them is a better bet for the input rate control. By only marking cells and not deleting them at the entry point, we can afford to mark a greater number of well behaved cells without the risk of deteriorating too much their loss probabilities. Still the ideal behavior would be that of not marking any of the average traffic cells.

However, one of the costs of this scheme is that it requires the implementation of priority handling by all network buffers. This priority scheme works as follows: an unmarked (high priority) cell is blocked only if there are only unmarked cells in the buffer. Furthermore, if there is at least one marked (low priority) cell in the full buffer when an unmarked cell arrives, the last marked cell is removed from the buffer (and lost) and the unmarked cell is accepted. In the other hand, a marked cell that finds the buffer full is lost.

Another way of looking at this performance measure is the effect that abusive traffic has on well behaved ones. In the ideal case there should be no sensible effect. However, with the use of proposed IRCs either a considerable amount of average traffic cells will get marked, which increases their probability of being deleted, or a considerable amount of abusive cells will go unmarked, interfering with the traffic of unmarked cells of well behaved sources.

In order to study the effect of violating traffic sources in the cell loss probability



of well behaved ones for the various input rate control mechanisms, the system reported in Figure 4.2 was simulated. The multiplexer (MUX), which represents one of the buffers in the network, behaves as described above. It has a buffer size of 35 cells, and its service rate is designed in order to achieve a  $GOS=10^{-4}$ , for the desired number of sources ( $n_{src}$ ). The choice of  $GOS=10^{-4}$  was made in order to reduce simulation running times; while  $n_{src}$  is the number of identical sources that can be multiplexed in a 150 Mbps channel in order to achieve  $GOS=10^{-9}$ . For a bursty source with parameters  $B_p=10$  Mbps,  $b=10$ , and  $L=100$ ,  $n_{src}=34$ . For our VBR source,  $n_{src}=22$ . Each source has an input rate control (IRC). All IRC's are assumed to be of the same type. The sources are divided into well-behaved and abusive sources. Well behaved sources have average bit rate  $B_m$ , as specified, while abusive sources, have an average bit rate  $B'_m = \sigma B_m$ .

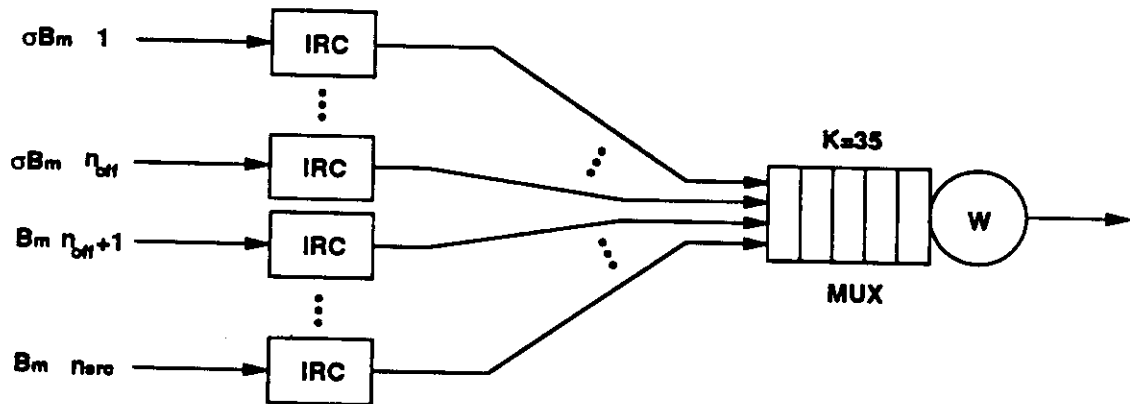


Figure 4.2: Simulation system for the study of IRC mechanisms.

In enforcing the average rate, some amount of fluctuation (due to burstiness) around this rate is tolerated. However, long, sustained bursts, at a rate much higher than the nominal rate, must be prevented, lest they create congestion inside the network. In order to characterize the burst filtering properties of the policing mechanisms, we introduce the notion of “reaction time”, i.e., time re-

quired for the policing mechanism to react (i.e., drop/mark cells) after the start of the burst. It turns out that accurate rate enforcement and prompt reaction to bursts are conflicting requirements, therefore a compromise must be reached by proper selection of the policing mechanism parameters.

### 4.3 Input Rate Control Mechanisms

In this section we present some of the policing mechanisms that have been proposed in the literature, namely, credit counter, jumping and moving windows, and leaky bucket, as well as 3 new mechanisms called delta, delta-2, and delta-3.

#### 4.3.1 Credit Counter

Joos and Verbiest [JV89] proposed an usage monitoring algorithm for VBR traffic based on credit counters which monitor the deviation of the traffic cell rate from predefined values. This deviation must be within predefined probabilities or else the cell will be deleted (or marked). In other words, each counter is associated with a given cell rate and is incremented or decremented according to whether the current measured cell rate is above or below its cell rate reference value.

The amount by which the counter is incremented or decremented is proportional to the desired probability that the cell rate be below or above the counter reference value. The parameters presented to the system are: cell rate average and deviation, and service time constant; if we assume a known distribution such as Gaussian.

Every time a cell arrives, its effect on the counter limits is checked. If the cell arrival makes the counter values exceed their limits, the cell is dropped (or possibly marked), otherwise the measured values are just updated. The more counters are used, the finer the approximation will be. In the other hand, an increase on the number of counters, adds on its complexity, and therefore, cost.

### 4.3.2 Jumping and Moving Windows

The *Jumping Window* (JW) mechanism basically consists in counting the number of cells generated during a given interval ( $W$ ) (the window size) and marking/deleting the ones that are above a desired maximum ( $N$ ). At the end of the interval, the counter is reset and a new window is started. Figure 4.3 shows the evolution of the counter in the jumping window scheme for a bursty traffic source. Note the marked cells after the counter reaches the maximum desired value  $N$ , and the counter reset at the end of the window (even if at the middle of a burst). The problem with JW is that with a reset the mechanism loses the past history (even immediate) of the traffic source. However, if the window is large enough, these losses in history should not cause any problem.

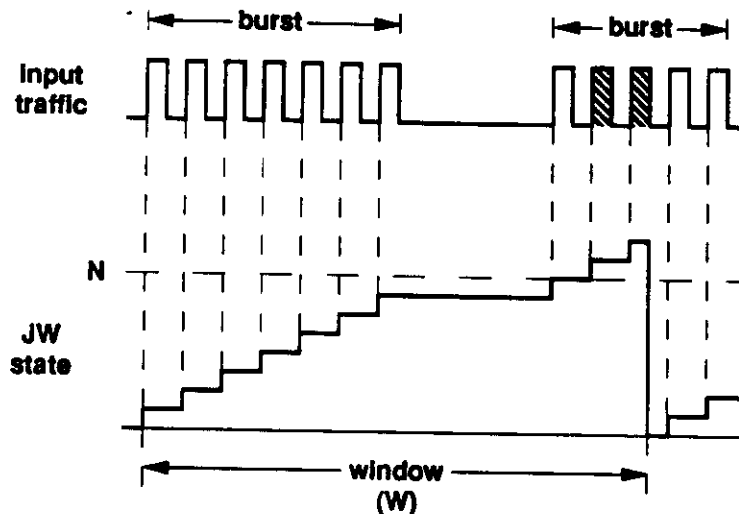


Figure 4.3: Example of evolution of a Jumping Window counter.

In the *Moving Window* (MW) mechanism the window as the name says moves in time keeping track of how many cells were transmitted during the past interval (the window size). Again cells that are found in excess of the desired maximum are marked/deleted. Figure 4.4 shows the evolution of the counter in the moving

window scheme for a bursty traffic source. Note that in the example, by the end of the examination period, the counter remains constant because at the arrival of a new cell, the window is advanced and a cell from the previous burst falls out the window.

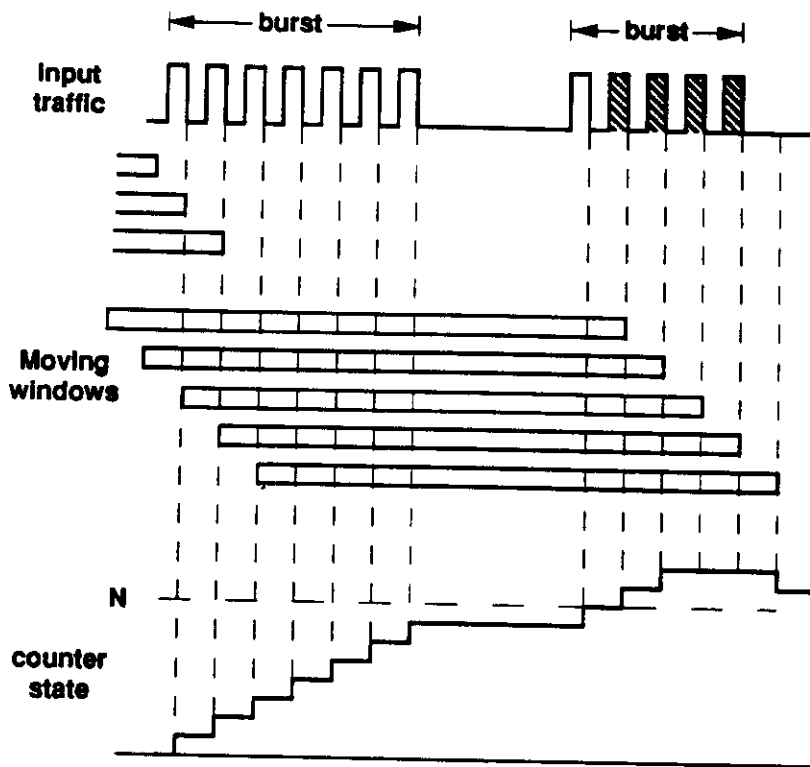


Figure 4.4: Example of evolution of Moving Window counter.

Implementation-wise, moving window is costlier because it requires that the mechanism remembers not only the number of cells in the window but also their arrival time, so that the window can be updated correctly. This requirement is specially costly for large windows.

Rathgeb [Rat90] compares leaky bucket (see Section 4.3.3) with jumping and moving window mechanisms. He makes a comparison based on Poisson sources. For bursty sources he just compares the loss probability as a function of the counter limits, noting that they are higher than in the Poisson case.

We analyze the Jumping Window mechanism under bursty traffic, in Section 4.4.1, and under VBR traffic in Section 4.5.1.

### 4.3.3 Leaky Bucket

The *Leaky Bucket* (LB) [Tur87, Tur86b] also known as *throughput-burstiness filter* [ELL89] is a well known ATM policing mechanism. Figure 4.5 shows the high level leaky bucket functional diagram. The heart of the leaky bucket is the pseudo queue. The pseudo queue is a counter which is incremented whenever a cell arrives from the monitored source and is decremented at a constant rate  $B_e$  ("depletion" rate). The counter has a maximum value  $Q$  (the pseudo queue maximum length). The cells that upon arrival find the counter at its maximum value are dropped.

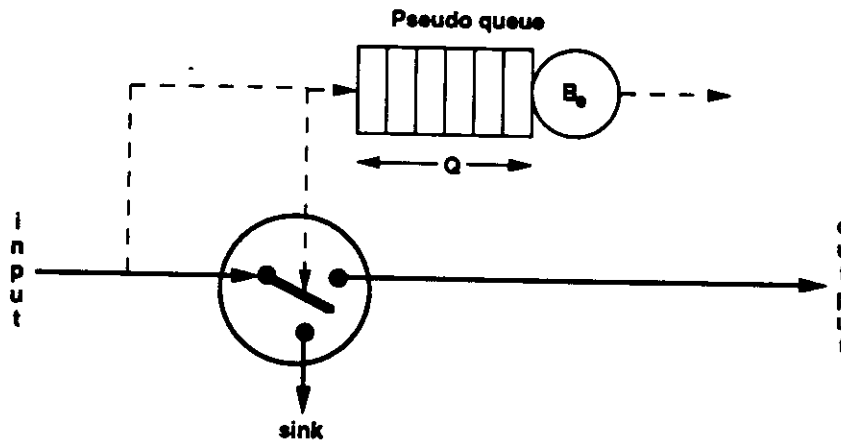


Figure 4.5: Leaky Bucket Functional Diagram.

Some variations of this basic approach have also been proposed in the literature. One such approach is the *Virtual Leaky Bucket* [GRF89] where the extra cells are not dropped but simply *marked* as they are forwarded into the network. If the network experiences any congestion, marked cells will be dropped in favor of non-marked ones.

Figure 4.6 shows the evolution of the state of a LB pseudo queue for a bursty

traffic source. In this figure we are representing bursts rather than individual cells. Note that cells get marked only when the pseudo queue length reaches the maximum value  $Q$ .

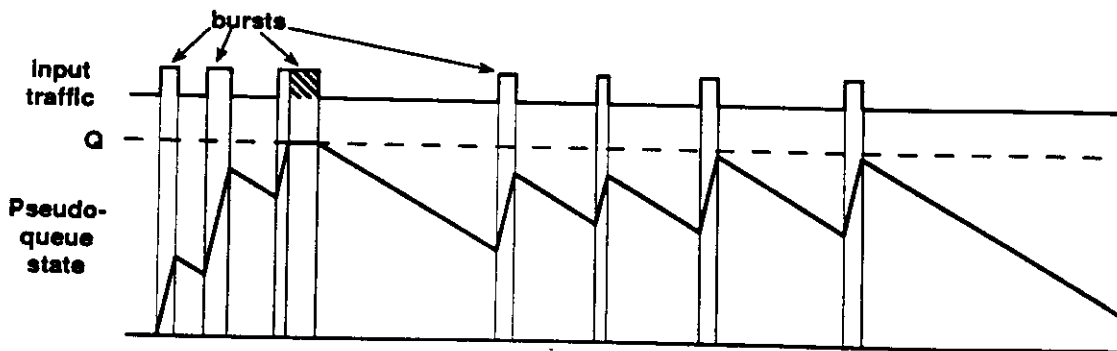


Figure 4.6: Example of evolution of the state of a LB pseudo queue.

Another approach is the *Generalized Leaky Bucket* [BCS90] which has a limit on the number of allowed violation packets (in order to avoid net overload), and also has a (cell) spacer to smooth the traffic.

In Section 4.4.2, we analyze Leaky Bucket mechanism under bursty traffic, while in Section 4.5.2 we analyze LB under VBR traffic.

#### 4.3.4 Delta

The *delta* input rate control is a mechanism where the decision to mark/delete is based on the period of time that a input source has been operating above its nominal (predefined) average rate.

This mechanism is implemented with two counters. The first one (the average counter) is similar to the LB's pseudo queue and it is used to capture the average behavior of the input source. The second counter (the peak counter) basically keeps track of how long the first counter is kept above its threshold, i.e., how long has the input source been operating above the nominal rate. The range of the

second counter is between 0 and a maximum value ( $T_{pk}$ ). Its value is increased at a constant rate while the average counter is above the threshold; otherwise it is decreased at the same constant rate. A cell that upon arrival finds the second counter at its maximum, is marked. Marked cells are not counted in the average counter.

With this scheme we have basically 4 parameters:

- the threshold of the average counter ( $T_a$ );
- the depletion rate of the average counter ( $B_a$ );
- the maximum value of the peak counter ( $T_{pk}$ ); and
- the insertion/depletion rate of the peak counter ( $B_{pk}$ ).

$T_{pk}$  and  $B_{pk}$  are dependent, since an increase in the insertion/depletion rate of the peak counter ( $B_{pk}$ ) can be compensated by an increase in the maximum value of the peak counter ( $T_{pk}$ ) and vice-versa. Let us call  $D_{pk}$  the marking delay, i.e., the amount of time that the average counter is allowed to stay above its threshold ( $T_a$ ) before it starts marking cells.  $D_{pk}$  is given by

$$D_{pk} = \frac{T_{pk} \cdot n_{cell}}{B_{pk}} \quad (4.1)$$

Figure 4.7 shows an example of the evolution of the state of a delta average and peak counters for the same input traffic of Figure 4.6. Note that cells get marked only when the peak counter is at its maximum value  $T_{pk}$ . Note also that marked cells are not counted by the average counter.

While LB marks cells only when the pseudo queue overflows, the delta mechanism can monitor how long did the input source generate cells above its nominal

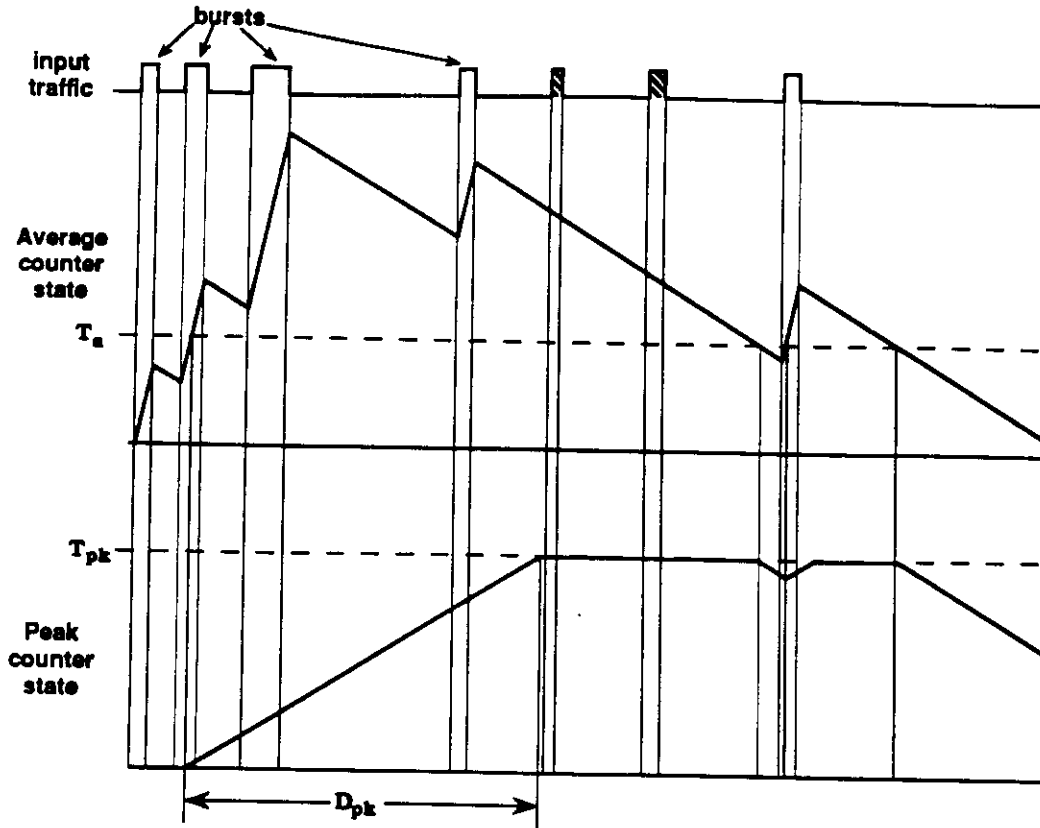


Figure 4.7: Example of evolution of the state of the Delta counters.

rate, and it can mark the cells that exceed the allowed maximum duration. Therefore, we claim that the delta mechanism is more flexible than LB. This flexibility comes from the freedom of choosing the marking delay.

In Section 4.4.3, we analyze the delta mechanism under bursty traffic.

### 4.3.5 Delta-2

The *delta-2* input rate control is a variation of the delta mechanism presented in Subsection 4.3.4.

It is implemented with two counters, but the first one (the average counter) beside having a threshold that affects the state of the second counter, has also a maximum value, and marked cells are still counted if the average counter is below



its maximum. The second counter (the peak counter) was not modified. A cell that upon arrival finds the second counter at its maximum is marked.

With this scheme we have basically 5 parameters:

- the maximum value of the average counter ( $Q$ );
- the threshold of the average counter ( $T_a$ );
- the depletion rate of the average counter ( $B_a$ );
- the maximum value of the peak counter ( $T_{pk}$ ); and
- the insertion/depletion rate of the peak counter ( $B_{pk}$ ).

Again,  $T_{pk}$  and  $B_{pk}$  are dependent, since an increase in the insertion/depletion rate of the peak counter has the same effect as decreasing the maximum value of the peak counter and vice-versa. Again we can define  $D_{pk}$  as given by Equation 4.1.

Figure 4.8 shows an example of the evolution of the state of a delta-2 average and peak counters for the same input traffic of Figure 4.6. Note that cells get marked only when the peak counter is at its maximum value  $T_{pk}$ .

The main difference between delta and delta-2 mechanisms is that while the average counter on delta tends to work around its threshold ( $T_a$ ) even for abusive traffic cells; in delta-2 this is no longer true, and abusive traffic will push the average counter to its maximum value ( $Q$ ). This effect can be explained by the fact that the delta mechanism counts only the unmarked cells, while the delta-2 mechanism counts all cells until the limit  $Q$  is reached.

### 4.3.6 Delta-3

The main disadvantage of the delta-2 mechanism, as far as implementation is concerned, is the need to compute the optimal value of the average threshold  $T_a$

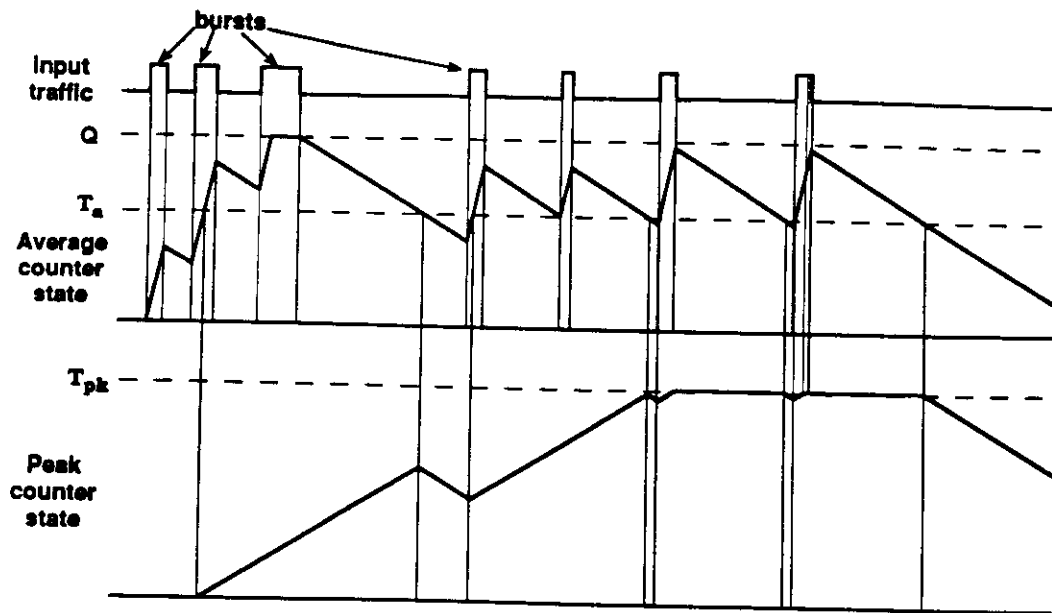


Figure 4.8: Example of evolution of the state of the Delta-2 counters.

which should depend on the statistics of the input traffic. The scheme proposed in this Section, called *delta-3* is an attempt to overcome this problem.

The delta-3 mechanism is a variation of the Moving Window and as the previous delta mechanisms, also has two counters. Each time a cell arrives the first counter (the average counter) is incremented, and it is depleted at a rate greater than the average nominal bit rate (i.e.,  $B_e > B_m$ ). If the source transmits at nominal average rate, the average counter should be idle on average during a fraction  $\frac{B_e - B_m}{B_e}$  of the time. The second counter (the gap counter) is used to “check” if this average idle/busy period ratio is being kept. The value of the gap counter is increased at a rate  $B_i$  (i.e., accumulates credits) when the average counter is idle (pseudo queue empty); and it is depleted at rate  $B_d$ . A cell that upon arrival finds the gap counter empty (i.e., without credits) is marked. Marked cells are not counted by the average counter.

The gap counter has also a maximum value  $S_{max}$  which limits the accumulation

of credits from excessively long silence periods. Furthermore, the ratio among  $B_i$  and  $B_d$  is such that  $S_{max}$  be depleted during an observation interval  $W$ .

$S_{max}$  and  $B_d$  are given by

$$S_{max} = \left( \frac{B_e - B_m}{B_e} \right) W \frac{B_i}{n_{cell}} \quad (4.2)$$

$$B_d = \left( \frac{B_e - B_m}{B_e} \right) B_i \quad (4.3)$$

where  $B_e$ ,  $B_m$ , and  $B_i$  are given in bits/sec;  $W$  is given in seconds;  $n_{cell}$  is given in bits/cell; and, finally,  $S_{max}$  is given in cells.

Figure 4.9 shows an example of the evolution of the state of the two counters on the delta-3 mechanism. Note that cells get marked only when the gap counter is at zero. Note also that marked cells are not counted by the average counter.

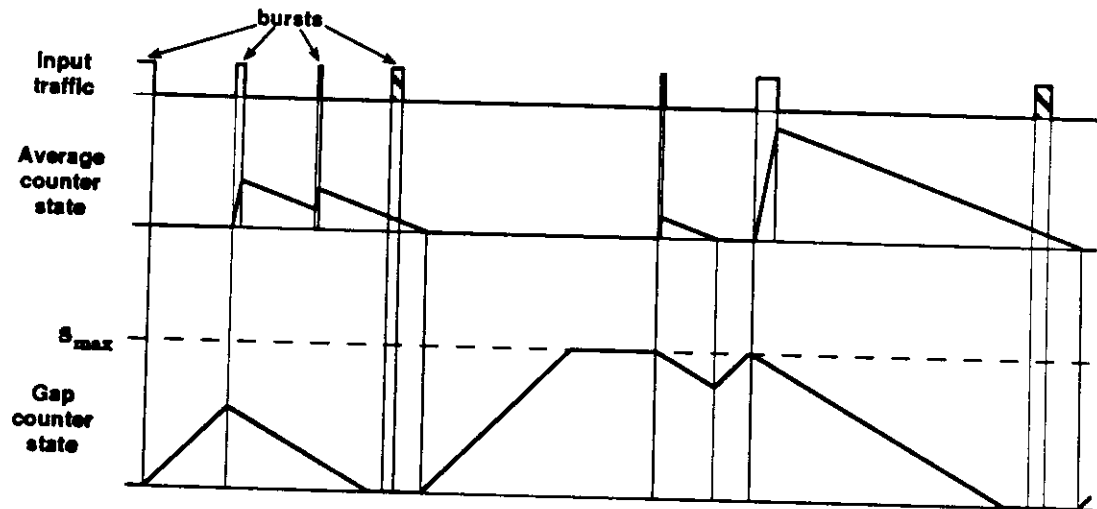


Figure 4.9: Example of evolution of the state of the Delta-3 counters.

In Section 4.4.5, we analyze the delta-3 mechanism under bursty traffic, while in Section 4.5.4 we analyze the delta-3 mechanism under VBR traffic.

#### 4.4 Analysis of IRC mechanisms under bursty traffic

In order to study the effectiveness of the several IRC mechanism under bursty traffic, we increase the source average bit rate by a factor  $\sigma$ . In other words, the modified average bit rate ( $B'_m$ ) is given by  $B'_m = \sigma B_m$ . We then obtain the cell “marking/loss” probability at the modified average bit rate.

The modified average bit rate is obtained by modifying the average active and silence periods keeping the average cycle period constant (i.e.,  $T + S = T' + S'$ ), as follows:

$$T' = \sigma T \quad (4.4)$$

$$S' = (1 - \sigma)T + S \quad (4.5)$$

As in subsection 3.2.2, we characterize a bursty traffic source by the following parameters: peak bit rate  $B_p$ , average bit rate  $B_m$  (or alternatively, burstiness  $b$ ), and average burst length  $L$ . Unless otherwise specified, the bursty source has the following parameters:  $B_p = 10$  Mbps,  $B_m = 1$  Mbps,  $b = 10$ , and  $L = 100$ .

Jumping window is analyzed in Subsection 4.4.1, leaky bucket in Subsection 4.4.2, delta mechanism in Subsection 4.4.3, delta-2 mechanism in Subsection 4.4.4, and delta-3 mechanism in Subsection 4.4.5.

##### 4.4.1 Jumping Window analysis under bursty traffic

###### 4.4.1.1 Marking/Loss probability

Figure 4.10 shows the jumping window marking probabilities for a bursty source with parameters  $B_p=10$  Mbps,  $b=10$ , and  $L=100$ . Several window sizes ( $W$ ) were used, keeping the ratio with the maximum number of cells ( $N$ ) constant, according to the source burstiness. A reasonable match with the ideal marking probabilities

was found for  $N=100K$ , and  $W=1M$ , which corresponds to 42.4 seconds.

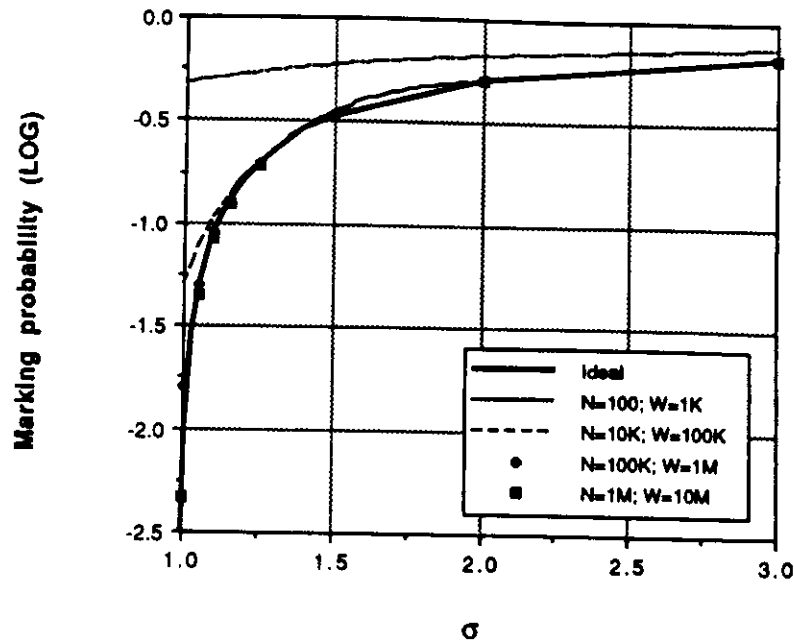


Figure 4.10: Jumping Window Marking Probabilities for a Bursty Source — 2 hours equivalent simulated time.

The marking probabilities just slightly changed when equivalent simulated time was reduced from 2 hours to 1,000 seconds.

#### 4.4.1.2 Marking effect on well behaved sources

Figure 4.11 shows the deleterious effect of marking on well behaved sources. As shown in Figure 4.10, even though the marking probability of abusive traffic sources (i.e.,  $\sigma > 1$ ) matches very well with the ideal curve, still a large proportion of legitimate cells gets marked as well. In the multiplexer buffer, marked cells from either well behaved or abusive sources are equally treated (i.e., have the same priority). Therefore, if the buffer is filled with marked cells from the abusive sources, the marked cells of the well behaved sources will be deleted. This effect

is more evident for a large number of abusive sources, and for large values of  $\sigma$ . These results shown in Figure 4.11 were obtained for 17 abusive sources out of 34.

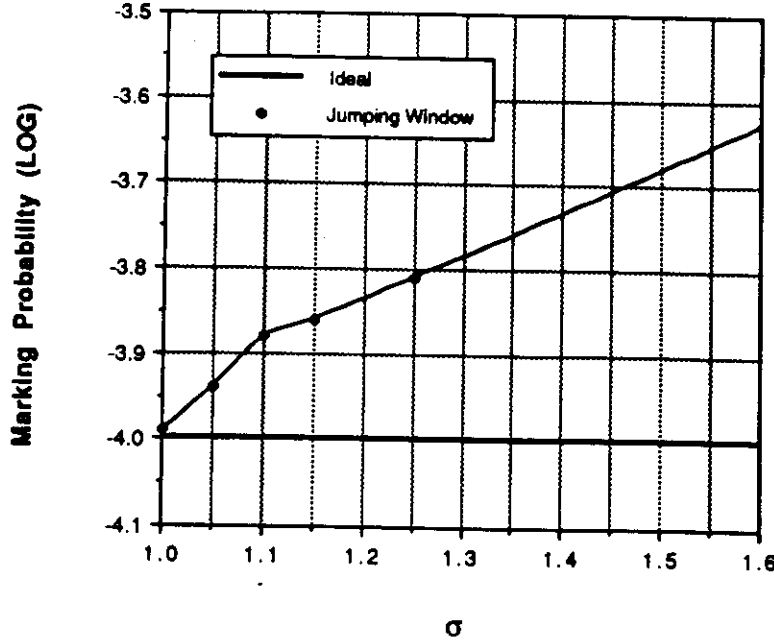


Figure 4.11: Effect of Marking on well behaved sources (Jumping Window).

#### 4.4.1.3 Lower bound on Reaction Time

In order to obtain a lower bound on reaction time, we assume that the source would continuously generate cells at peak bit rate, at least until the maximum number of cells ( $N$ ) is reached. In other words,

$$RT_{JW}^b = \frac{N \cdot n_{cell}}{B_p} \quad (4.6)$$

This expression also assumes that the counting is not interrupted by an end of window. On average the counter should be at  $N/2$ , therefore,

$$\overline{RT}_{JW}^b = \frac{N \cdot n_{cell}}{2B_p} = \frac{RT_{JW}^b}{2} \quad (4.7)$$

Table 4.1 reports the values of  $RT_{JW}^b$  and  $\overline{RT}_{JW}^b$  for several values of  $N$ .

Table 4.1: Jumping Window Reaction Time (lower bounds)

| $N$  | $RT_{JW}^b$ | $\overline{RT}_{JW}^b$ |
|------|-------------|------------------------|
| 100  | 0.004       | .002                   |
| 10K  | 0.4244      | .212                   |
| 100K | 4.240       | 2.120                  |
| 1M   | 42.4        | 21.200                 |

#### 4.4.2 Leaky Bucket analysis under bursty traffic

##### 4.4.2.1 Marking/Loss probability

In the analysis which follows, we use the UAS model with finite buffer (Appendix B) [Tuc88] and simulation to obtain the “marking/deletion” probability of cells for a range of leaky bucket parameters.

Initially, for a given maximum counter value  $Q$ , the depletion rate  $B_e$  is chosen so that the desired GOS is achieved as an equality, at nominal average bit rate. This strict requirement is needed in case LB deletes any detected abusive cell.

Turner [Tur87] suggests to choose  $B_e = B_m$  and  $Q = L_{max}$ , where  $L_{max}$  is the maximum burst length. However, for these values, the utilization factor of the pseudo-queue is 1 and the cell loss probability is unacceptably high. To show this, Table 4.2 reports the cell loss/marking probability ( $P$ ) and the average leaky bucket size ( $\overline{Q}$ ) as a function of  $Q$  for our bursty source and leaky bucket depletion rate  $B_e = 1,000,001$  bps. (Note that  $B_e$  was chosen slightly higher than  $B_m$  so that the utilization factor would be less than one, and the UAS model could be used to obtain the cell loss/marking probability). As one can see, even for large leaky bucket sizes (which lead to high reaction times), the loss/marking probability

is still much higher than the desired GOS ( $10^{-9}$ ).

Table 4.2: Cell loss/marking probability for  $B_e \approx B_m(L = 100)$ .

| $Q$ (cells) | $\log_{10} P$ | $\bar{Q}$  |
|-------------|---------------|------------|
| 100         | -0.370        | 26.58      |
| 200         | -0.554        | 69.31      |
| 1,000       | -1.129        | 459.17     |
| 3,000       | -1.581        | 1,456.79   |
| 5,000       | -1.798        | 2,456.26   |
| 10,000      | -2.095        | 4,955.79   |
| 50,000      | -2.791        | 24,953.00  |
| 100,000     | -3.092        | 49,945.24  |
| 1,000,000   | -4.094        | 498,926.57 |

Therefore,  $B_e$  must be chosen larger than  $B_m$ , i.e.,  $B_e = CB_m$ , where  $C > 1$ .

The question now becomes: which values of  $C$  and  $Q$  should be used in order for a well behaved source to experience a loss/mark probability of the same order as its GOS (e.g.,  $10^{-9}$ )?

Due to the oversize factor  $C$ , we expect that some sources may exceed the declared  $B_m$  and yet go undetected. However, we want to keep this abuse as rare as possible.

Figure 4.12 shows the cell marking/loss probability as a function of  $\sigma$  for several values of  $Q$ . For each  $Q$ ,  $B_e$  was chosen so that the GOS would be achieved at nominal average bit rate. The choice of GOS =  $10^{-5}$  was made so that it would be feasible to compare analytical with simulation results. The analytic model was



also run at  $GOS=10^{-9}$ . The results are not reported here, but confirm the trends observed with  $GOS=10^{-5}$ .

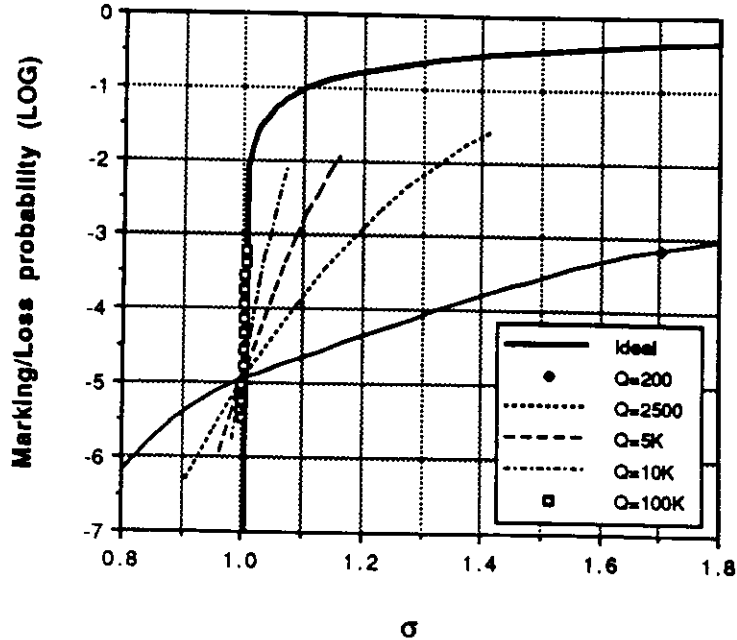


Figure 4.12: Leaky Bucket Behavior for a Bursty Source ( $GOS=10^{-5}$ ).

Table 4.3 presents the leaky bucket parameters corresponding to each curve in Figure 4.12. As we can see from Figure 4.12 and Table 4.3, the LB approaches the ideal behavior as  $Q$  increases and  $B_e$  decreases. In the limit,  $B_e \rightarrow B_m$  as  $Q \rightarrow \infty$ . At the other end of the spectrum, we note that a leaky bucket with  $Q = 2L = 200$ , as suggested in [GRF89], with  $B_e$  chosen in this way provides almost no control.

Thus, the bigger the counter size  $Q$  the better the control. However, this comes at a cost: the reaction time also increases. In other words, the lag time between the jump in average bit rate and its detection (and correction by means of the loss/marking mechanism) becomes larger.

Note that these results corresponds to sources with steady average bit rates of  $B'_m = \sigma B_m$ . A temporary jump in average bit rate with same magnitude would

Table 4.3: Leaky Bucket Parameters for Bursty Traffic (GOS =  $10^{-5}$ )

| $Q$ (cells) | $B_e$ (bits/sec) | $C$   | $\log_{10} P$ | $\bar{Q}$ (cells) |
|-------------|------------------|-------|---------------|-------------------|
| 200         | 8,012,745        | 8.013 | -4.96         | 3.2               |
| 2,500       | 1,464,733        | 1.465 | -5.01         | 168.7             |
| 5,000       | 1,180,517        | 1.181 | -5.01         | 443.7             |
| 10,000      | 1,076,946        | 1.077 | -5.01         | 1,047.1           |
| 100,000     | 1,005,099        | 1.005 | -5.02         | 15,693.3          |

have a lower probability of being detected.

#### 4.4.2.2 Reaction Time

In the previous section we have mainly focused on the ability to enforce a given average input rate, and have investigated the parameters that influence such property. Here, we investigate another important property, that is, the ability to detect and prevent long bursts. We will use reaction time as a performance measure, and will evaluate its dependence on system parameters.

The reaction time may be defined as the average time  $T_{fill-up}$  required to fillup the bucket after the start of the burst. An approximate expression of  $T_{fill-up}$  is derived as follows. At nominal average bit rate, the average LB counter size  $\bar{Q}$  (see Table 4.3) is obtained from the queue length distribution given by the UAS model with finite buffer size.  $T_{fill-up}$  is then given by

$$T_{fill-up}(\sigma) = \begin{cases} \frac{(Q-\bar{Q})n_{cell}}{B'_m - B_e} = \frac{(Q-\bar{Q})n_{cell}}{(\sigma-C)B_m}, & \sigma > C \\ \infty, & \text{otherwise} \end{cases} \quad (4.8)$$

Figure 4.13 shows the reaction time for the parameters of Table 4.3.

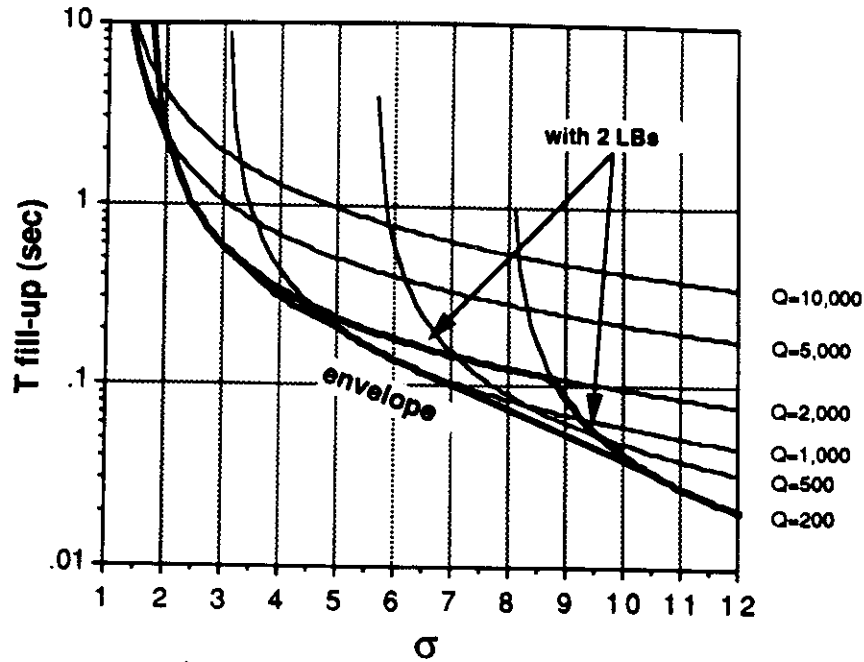


Figure 4.13: Reaction Time for several counter sizes.

We note that, based on our approximate model, the LB will not detect a burst (no matter how long it lasts) if  $B'_m \leq B_e$ , i.e.,  $\sigma \leq C$ . If we want to trap low level bursts, we should thus use small values of  $C$  (i.e., close to unity). This, in turn, implies the use of large values of  $Q$ , as shown in Table 4.3. However, a large value of  $Q$  (and, consequently,  $\bar{Q}$ ) leads to a large reaction time. In other words, it will take a long time to detect the burst.

These considerations suggest that there is a trade off between sensitivity to small, sustained increases in average rate and prompt reaction to big jumps in rate. To illustrate this trade off with an example, consider the two cases  $Q = 200$  and  $Q = 10,000$ , and assume that the system is subjected to two different rate jumps (i.e., prolonged bursts) of size  $\sigma = 2$  and  $\sigma = 10$  respectively. With  $Q = 200$ , a  $\sigma = 2$  step increase goes unnoticed (see Fig. 4.14); a  $\sigma = 10$  step increase, on the

other hand, is detected within 50 msec (see Fig. 4.13). With  $Q = 10,000$ , a  $\sigma = 2$  step increase will be very effectively stopped (see Fig. 4.12); a  $\sigma = 10$  rate jump will be detected only after a 500 msec (see Fig. 4.13).

In general, the choice of the best compromise between rate enforcement and prompt reaction will be determined by the application at hand. In some cases, however, both properties are required. To this end, multiple LB's in parallel can be used, with different  $Q$  values. The multiple LB operation is very straightforward: a cell is dropped/marked whenever one of the buckets overflows (i.e., OR operation). For the case depicted in Figure 4.13, if the two buckets with  $Q = 200$  and  $Q = 2,000$  are used in parallel, the resulting reaction time curve is the minimum of the two curves (i.e., the bucket with the smallest reaction time prevails). In the ideal case, one could maintain an infinite number of parallel buckets, with different counter sizes covering the entire feasible range. In this case, the response time would be the lower envelope of the family of curves thus generated.

It would be obviously impractical to maintain a very large number of buckets, because of the associated processing O/H at the entry node. However, it makes practical sense to maintain two buckets (e.g.,  $Q = 200$  and  $Q = 10,000$  in the previous application). The small bucket is used to detect big, sudden jumps in rate; the large bucket is used to trap small, but prolonged step increases in rate.

#### 4.4.2.3 Lower Bound on Reaction Time

The fluid-flow approximation for the reaction time expressed by  $T_{fill-up}$  in Equation 4.8 is a very pessimistic one. It tells us that the reaction time is infinity if  $B'_m = B_e$  as in the case when  $B_e = B_m$  and  $\sigma = 1$ , which is not true.

On the other hand, a lower bound on the reaction time can be obtained by assuming that the source would continuously generate cells at peak bit rate until

filling-up the leaky bucket. Actually, we could have two measures. One would assume an empty initial state, while the other would assume the average state as the initial one. Let us call them  $RT_{LB}^b$  and  $\overline{RT}_{LB}^b$ , respectively. Therefore, we have:

$$RT_{LB}^b = \frac{Q \cdot n_{cell}}{B_p - B_e} \quad (4.9)$$

and

$$\overline{RT}_{LB}^b = \frac{(Q - \overline{Q}) \cdot n_{cell}}{B_p - B_e} \quad (4.10)$$

Table 4.4 shows the values of  $RT_{LB}^b$  and  $\overline{RT}_{LB}^b$  obtained by applying Equations 4.9 and 4.10 to some LB parameters reported in Tables 4.2 and 4.3.

Table 4.4: Leaky Bucket Reaction Time (lower bound)

| $Q$ (cells) | $B_e$ (bits/sec) | $\overline{Q}$ (cells) | $RT_{LB}^b$ (msec) | $\overline{RT}_{LB}^b$ (msec) |
|-------------|------------------|------------------------|--------------------|-------------------------------|
| 200         | 8,012,745        | 3.2                    | 42.67              | 41.99                         |
| 2,500       | 1,464,733        | 168.7                  | 124.20             | 115.80                        |
| 5,000       | 1,180,517        | 443.7                  | 240.40             | 219.05                        |
| 10,000      | 1,076,946        | 1,047.1                | 475.20             | 425.42                        |
| 100,000     | 1,005,099        | 15,693.3               | 4,713.78           | 3,974.03                      |
| 50,000      | 1,000,001        | 24,953.0               | 2,356.00           | 1,180.00                      |
| 100,000     | 1,000,001        | 49,945.2               | 4,711.00           | 2,358.00                      |

Note that  $RT_{LB}^b$  has approximately a linear growth with  $Q$  in the range  $2,500 \leq Q \leq 100,000$ .

#### 4.4.2.4 Simulation Results

Figures 4.14, 4.15 and 4.16 compare analytical with simulation results for a leaky bucket with  $Q = 200$ ,  $Q = 2,500$ , and  $Q = 10,000$ , respectively, for the same bursty source of Figure 4.12.

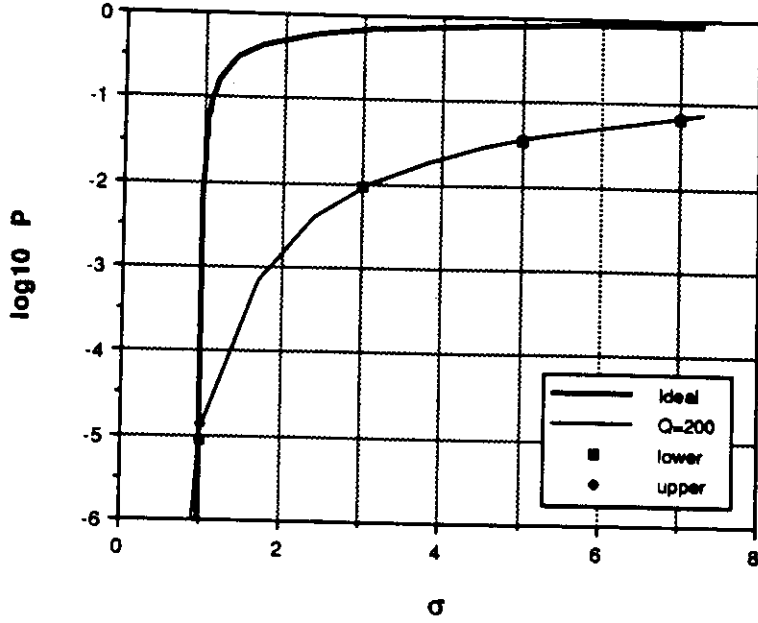


Figure 4.14: Analytical versus Simulation Results for a Bursty Source Leaky Bucket with  $Q=200$ .

At each simulation point, the dots correspond to a 95% confidence interval of the cell loss/marking probability ( $P$ ). In comparing analytic and simulation results for  $Q = 200$ ,  $Q = 2,500$  and  $Q = 10,000$ , we note that the accuracy of the analytical model depends on the queue size. The larger the queue, the less accurate are the analytical results. Tucker [Tuc88] had already observed this phenomenon, which is related to the amount of time the queue spends at its extremities. The reason being that the talker activity model (see Appendix B) is only needed to model the queue behavior. Neither the amount of packet (cell) loss when the

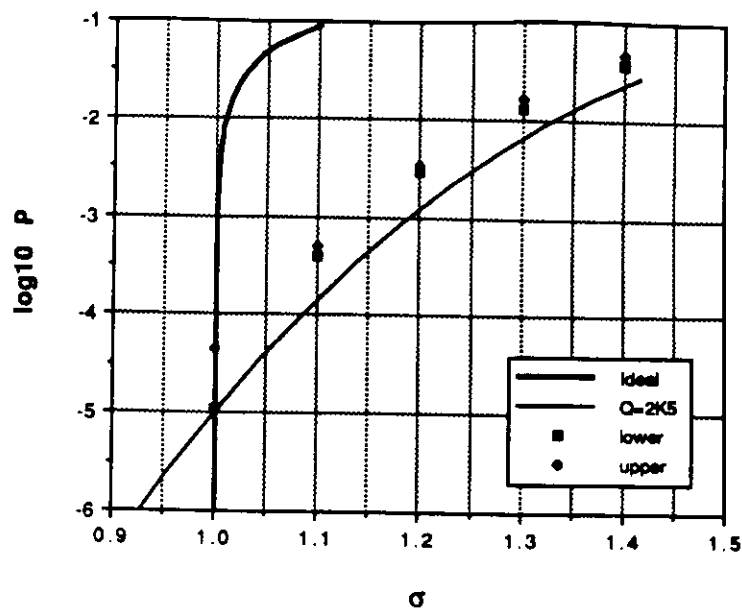


Figure 4.15: Analytical versus Simulation Results for a Bursty Source Leaky Bucket with  $Q=2,500$ .

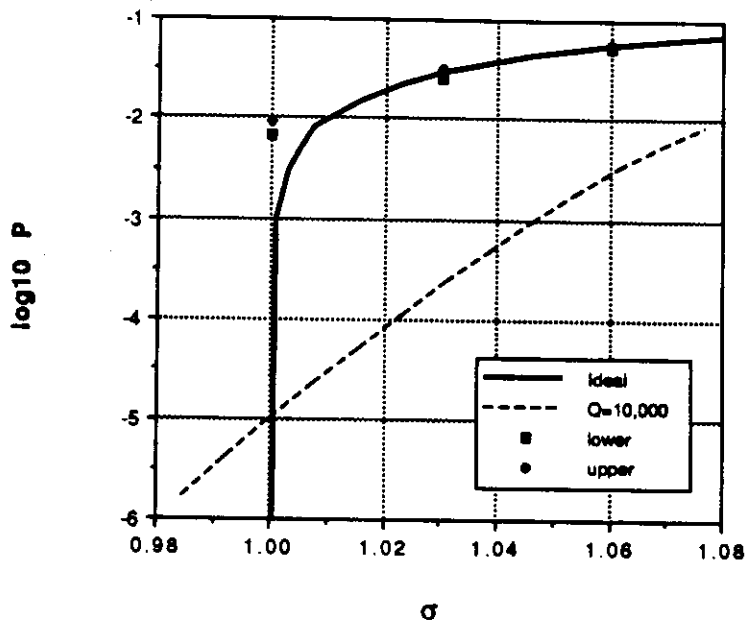


Figure 4.16: Analytical versus Simulation Results for a Bursty Source Leaky Bucket with  $Q=10,000$ .

queue is at its limit, nor the amount of available capacity when the queue is empty, depends on the talker activity model. Therefore, the longer the queue spends at its extremities, the less the inaccuracies of the talker activity model affect the final result.

Note that since we are working with very low loss probabilities, the queues will be empty most of the time. Recall that smaller queues imply more accurate results. Thus, for the same queue size, the analytical model is more accurate for lower loss probabilities, which comes to our advantage since the desired loss probability is at the order of  $10^{-9}$ .

#### 4.4.2.5 Marking/Loss probability revisited

In Subsection 4.4.2.1 we imposed the strict requirement that cells be deleted by the leaky bucket at GOS levels, exactly because we were deleting cells. However, if the marking scheme is used instead, we can afford to mark legitimate cells as long as this does not significantly affects the cell loss probability of well-behaved sources.

If this is the case, we could even have  $B_e = B_m$ . Figure 4.17 compares the cell marking probabilities for the LB working at nominal average bit rate (i.e.,  $B_e = B_m$ ), with the ideal marking probability. The simulation results with 95% confidence intervals are reported also in Table 4.5. These results were obtained through simulation except for the values at  $\sigma = 1$  for the curves  $Q=50K$ ,  $Q=100K$ , and  $Q=1M$ , in which we used the analytical results reported in Table 4.2. The simulations were run for a fixed amount of simulated time equivalent to 1.85 hours.

From Figure 4.17 and Table 4.5 we observe that as the bucket size  $Q$  increases, the marking probabilities approaches the ideal one from above, and then starts to get further away. This is due to reaction and simulated times. If we let the



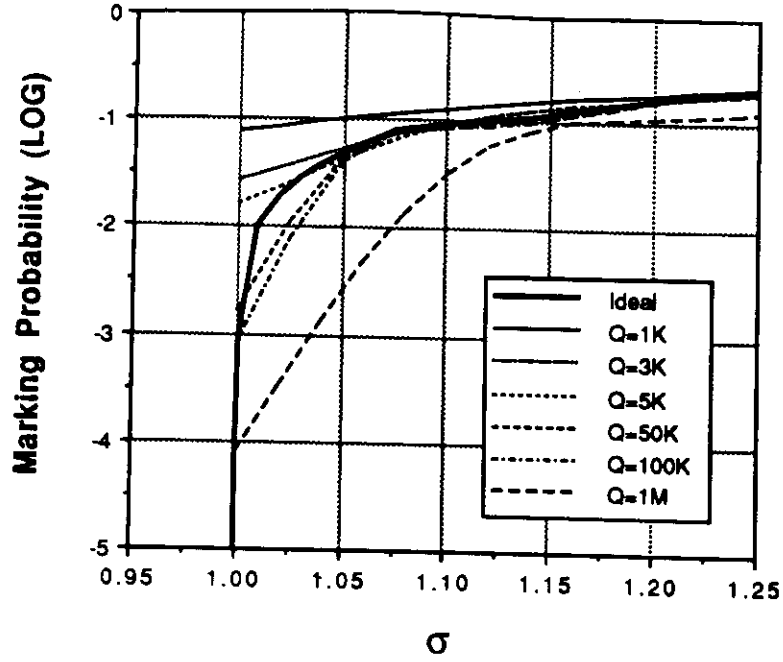


Figure 4.17: Leaky Bucket Marking Probabilities for a Bursty Source ( $B_e = B_m$ ).

Table 4.5: LB Cell marking probabilities for  $B_e \approx B_m (L = 100)$ .

| $Q$ (cells) | Normalized average rate ( $\sigma$ ) |                 |                 |                 |                |
|-------------|--------------------------------------|-----------------|-----------------|-----------------|----------------|
|             | 1.00                                 | 1.05            | 1.10            | 1.20            | 1.50           |
| IDEAL       | $-\infty$                            | -1.32           | -1.04           | -0.78           | -.48           |
| 1K          | $-1.12 \pm .01$                      |                 | $-0.89 \pm .01$ | $-0.73 \pm .00$ | -.47           |
| 3K          | $-1.58 \pm .03$                      |                 | $-1.03 \pm .01$ | $-0.78 \pm .01$ | -.48           |
| 5K          | $-1.80 \pm .05$                      |                 | $-1.04 \pm .01$ | $-0.78 \pm .00$ | -.48           |
| 50K         | $-3.35 \pm 1.06$                     | $-1.35 \pm .04$ | $-1.07 \pm .02$ | $-0.79 \pm .01$ | -.48           |
| 100K        | " $-\infty$ "                        | $-1.39 \pm .03$ | $-1.08 \pm .02$ | $-0.80 \pm .01$ | $-.49 \pm .01$ |
| 1,000K      | " $-\infty$ "                        | " $-\infty$ "   | $-1.50 \pm .05$ | $-0.95 \pm .01$ | -.54           |

simulation go further, the curves that are below the ideal would tend to approach it.

As a result, in order to be able to mark all the abusive cells in a reasonable amount of time, we will need to tolerate the marking of some legitimate cells, hoping that the cell loss probability of well behaved sources would not be significantly affected.

#### 4.4.2.6 Marking effect on well behaved sources

In this subsection we will study the effect of marking legitimate cells, on the cell loss probability of well behaved sources.

Due to the nonconformity of the LB cell marking probability with the ideal one, a certain percentage of the abusive cells will go undetected or a number of legitimate cells will be marked. If the amount of undetected abusive cells is large in comparison with the number of generated cells of a well behaved source, the cell loss probability of the well behaved sources will quickly deteriorate.

Figure 4.18 shows the cell marking probabilities for a LB working at  $B_e = B_m$ , and with bucket sizes  $Q=50K$  cells, and  $Q=100K$  cells, during an observation period of 1,000 seconds. Notice that the marking probabilities are lower than the ones obtained for a large observation period (Figure 4.17). In the majority of cases, no cells from well behaved sources were marked.

However, due to the marking of abusive sources (i.e., sources for which  $\sigma > 1$ ) below the ideal marking, the cell loss probability of well behaved sources will suffer an increase, as reported in Figure 4.19. This deleterious effect becomes more significant as the number of abusive sources increases.

In Section 4.6 we compare these results with the ones obtained for other IRC mechanisms.

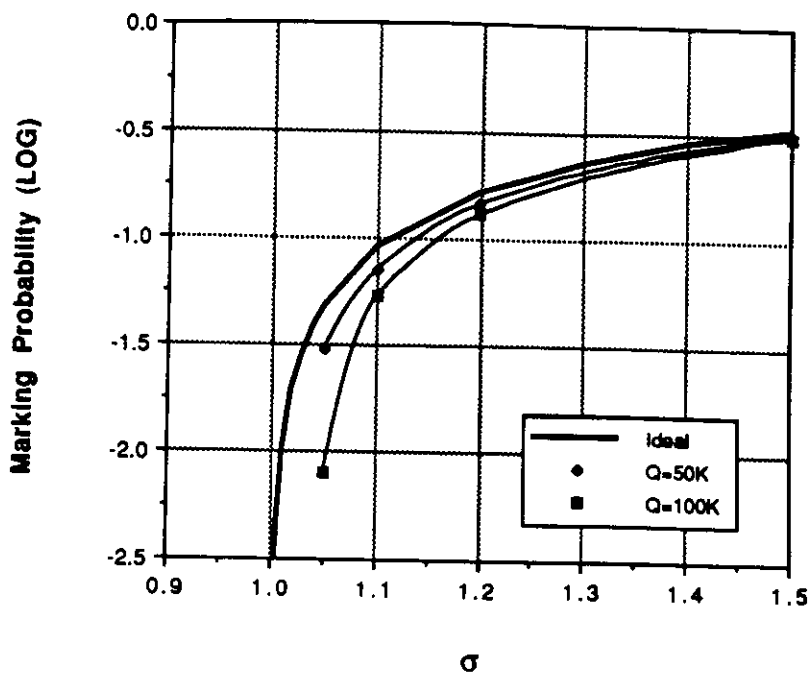


Figure 4.18: Leaky Bucket Marking Probabilities (1,000 sec. observation period).

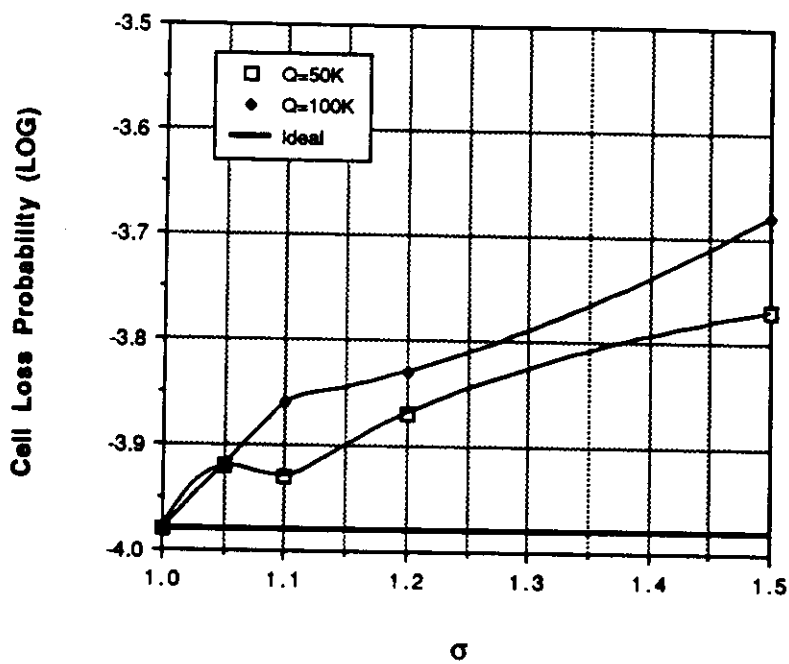


Figure 4.19: Effect of Marking cells on well behaved sources (Leaky Bucket).

### 4.4.3 Delta mechanism analysis under bursty traffic

As we mentioned in Section 4.3.4, this scheme has basically four parameters:

- the threshold of the average counter ( $T_a$ );
- the depletion rate of the average counter ( $B_a$ );
- the maximum value of the peak counter ( $T_{pk}$ ); and
- the insertion/depletion rate of the peak counter ( $B_{pk}$ ).

Even though the average counter does not have an explicit maximum value, its value will not go to infinity (or overflow) even if the source average bit rate is greater or equal to  $B_a$ . In fact, for  $\sigma \geq B_a/B_m$  the average counter will always work around  $T_a$ . The reason being that when the average counter goes above  $T_a$ , the peak counter starts counting, and if the average counter remains above  $T_a$ , it will reach  $T_{pk}$  after  $D_{pk}$  seconds. At this point, cells will be marked, and therefore, not counted by the average counter. This will bring the average counter down to  $T_a$ . When it goes below  $T_a$ , the peak counter will get down from  $T_{pk}$ , cells will start being counted again by the average counter, bringing it up above  $T_a$ . This will make the peak counter increase again, reaching  $T_{pk}$ , cells will not be counted by the average counter, and so on, and so forth.

If we choose  $B_a > B_m$  we could set the value of the average counter threshold ( $T_a$ ) to the average queue length obtained for example with the UAS method.

In the other hand, the value of  $D_{pk}$  must be chosen so that the marking probability for well behaved sources be as low as possible, yet not making the reaction time excessively high.

For  $1K \leq T_a \leq 4K$ , and  $D_{pk} = 2.12$  sec, marking probabilities are comparable to LB's with bucket size  $Q = T_a$ .

Since working at  $\rho = 1$  does not give us a good control on the state of the average counter for nominal traffic rates, let us choose a value for  $B_a$  which gives us a stable average counter size for nominal traffic sources. We used the UAS with finite buffer size model to obtain the average queue length for several values of  $B_a$  and buffer sizes  $Q = 1M$ , and  $Q = 100K$ . Starting from  $B_a \approx B_m$  we stopped when the average queue length of both buffers was the same (i.e., the result was independent from the buffer size).

For our bursty source we thus obtained  $T_a = 5,781.65$  for  $B_a = 1,014Kbps$ , or  $\rho = .986$ . However, simulation results produced a longer average queue length, with a high variability for a 2 hours equivalent simulated time:  $29,344 \pm 12,848$ .

Figure 4.20 shows the marking probability for a delta mechanism with parameters:  $T_a = \{30,000; 50,000\}$ ,  $B_a = 1,014Kbps$ , and  $D_{pk} = 1$  sec. We note that at this high utilization there is little difference in the marking probabilities with different values of the average threshold. Furthermore, we observe that the marking probability tends to 0 around  $\sigma = (B_a/B_m) = 1.014$ . This suggests that if we use  $B_a = B_m$ , the marking probabilities behavior will be similar to leaky bucket's.

#### 4.4.3.1 Lower Bound on Reaction Time

As in Section 4.4.2.3, we will obtain a lower bound on the reaction time by assuming that the source would continuously generate cells at peak bit rate until start marking (i.e., average counter goes above its threshold and the peak counter reaches its maximum value).

In other words,

$$RT_{DT}^b = \frac{T_a \cdot n_{cell}}{B_p - B_a} + D_{pk} \quad (4.11)$$

Since the average counter tends to work around its threshold  $T_a$ , we will assume that in the average state the average counter is already at the threshold. Therefore,

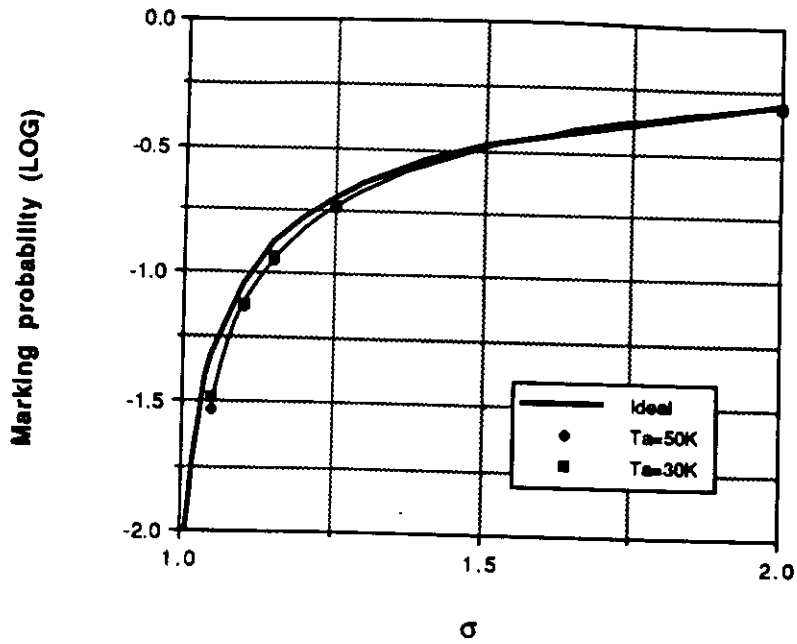


Figure 4.20: Delta-Marking Probabilities ( $D_{pk} = 1$  sec.).

the reaction time from the average state will be just the time required to fill the peak counter, which is given by  $D_{pk}$ . Therefore, we have:

$$\overline{RT}_{DT}^b = D_{pk} \quad (4.12)$$

#### 4.4.4 Delta-2 mechanism analysis under bursty traffic

As we mentioned in Section 4.3.5, this scheme has basically five parameters:

- the maximum value of the average counter ( $Q$ );
- the threshold of the average counter ( $T_a$ );
- the depletion rate of the average counter ( $B_a$ );
- the maximum value of the peak counter ( $T_{pk}$ ); and
- the insertion/depletion rate of the peak counter ( $B_{pk}$ ).

In order to avoid the variability of average queue length associated with operating the average counter at utilization  $\rho = 1$ , we will choose  $B_a$  so that the average counter works at a more stable region (e.g.,  $\rho = .5$ ) for nominal traffic sources. We will call  $\rho_n$  the utilization of the delta-2 mechanism at nominal average bit rate (i.e.,  $\rho_n = \frac{B_m}{B_a}$ ).

We can therefore use some analytical or approximate method (such as the UAS model) to obtain the average counter value for a nominal traffic source. This value should be used as the threshold  $T_a$ .

The maximum value of the average counter ( $Q$ ) has to be chosen so that abusive traffic cells be marked at desired rates. In the other hand, it has to be large enough so that it will not interfere with the average counter value at nominal traffic. The closer  $Q$  is from  $T_a$ , the lower is the probability that the average counter is above  $T_a$  and, therefore, the lower is the cell marking probability even for abusive sources.

In the other hand, the choice of the peak counter maximum delay ( $D_{pk}$ ) must be made so that the probability of marking/deleting nominal traffic cells be as low as desired.

As it will be shown in the next Subsection, this mechanism severely penalizes sources which work in the region  $\rho \geq 1.0$  (i.e, sources whose average bit rate are at least  $B_m/\rho_n$ ).

#### 4.4.4.1 Marking/Loss probability

In this Subsection, we compare the marking/loss probability of the delta-2 mechanism with that of an ideal policing mechanism. As we will see, the general behavior of the delta-2 mechanism is that of marking abusive cells at a probability smaller than the ideal until the utilization approaches one. After this point, the marking probability is much higher than the ideal, therefore, severely penalizing

sources that generates abusive traffic.

We want to avoid the deleterious effect that undermarking abusive cells may cause on the cell loss probability of well behaved ones. Therefore, we must choose  $B_a$  so that the region in which the marking probability is below the ideal be as small as possible, still leaving some space for small variations in the rate of the sources, before entering the penalty region. Another important factor in the choice of  $B_a$  is that the average length of the average counter be as stable as possible so that we can use it as the threshold  $T_a$ .

Our bursty source has the following parameters:  $B_p=10$  Mbps,  $b=10$ , and  $L=100$ . Initially, let us choose  $B_a = 2$  Mbps (i.e.,  $\rho_n = .5$ ). In this case we find, using the UAS model, that the average value of the average counter is approximately 75. Therefore, we make  $T_a = 75$ . Figure 4.21 shows the marking probabilities for the delta-2 mechanism with  $T_a = 75$ ,  $D_{pk} = 1.06$  sec, and  $Q=\{1K, 2K, 3K\}$ .

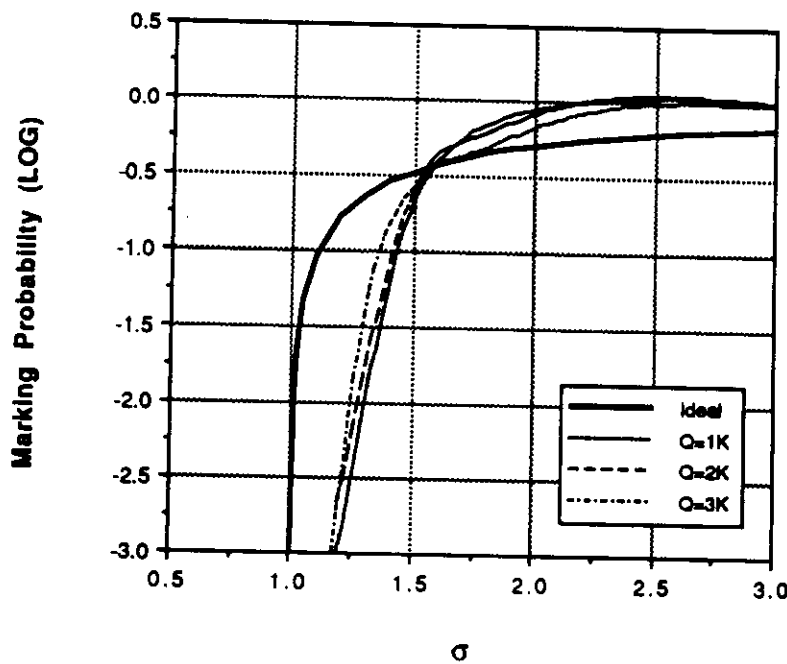


Figure 4.21: Delta-2 Marking Probabilities ( $\rho_n = 0.5$ ).



Note that in Figure 4.21, all three Delta-2 mechanisms mark less cells than the ideal for up to  $\sigma \approx 1.6$ , and from that point on they mark more cells than the ideal, reaching a point where they mark all cells.

Now, we obtain the marking probabilities at  $\rho_n = 0.8$ . The UAS model with a single source is not very precise, specially at high utilizations. For these parameters it gave an average counter length of 319 with  $Q \geq 5K$ , while simulation produced the value  $421 \pm 10$ . In order to work safely, we chose  $T_a = 450$ . Figure 4.22 shows the marking probabilities for the delta-2 mechanism with  $T_a = 450$ ,  $Q=5K$ , and  $D_{pk} = \{4.24, 6.36, 8.48, 10.60\}$ sec.

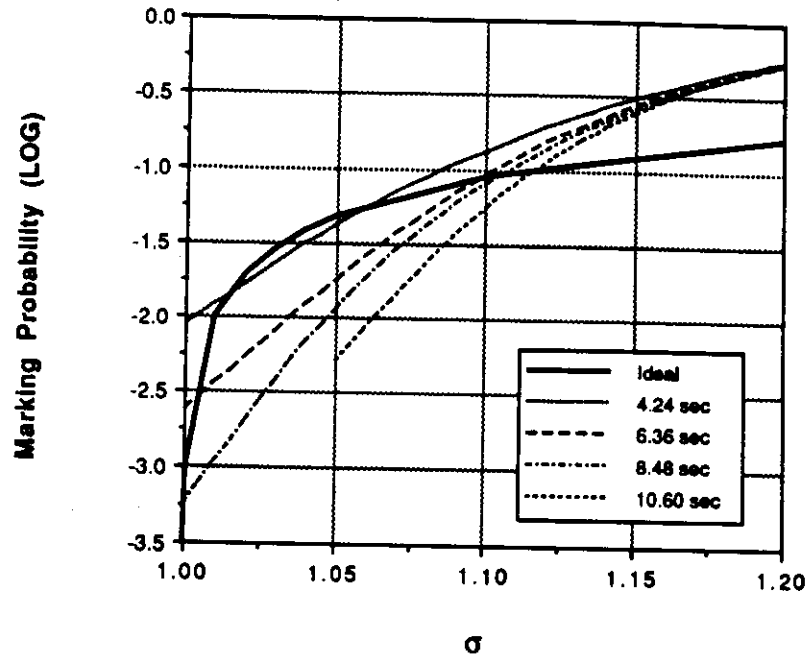


Figure 4.22: Delta-2 Marking Probabilities ( $\rho_n = 0.8$ ).

By decreasing  $B_a$  (increasing  $\rho_n$ ) the value of  $\sigma$  for which the Delta-2 marking probabilities are superior than the ideal, is reduced. This reduces the number of abusive cells that gets by unmarked, but it also increases the marking probability at nominal average bit rate.

#### 4.4.4.2 Marking effect on well behaved sources

Due to the marking characteristics of the delta-2 mechanism, well behaved sources will have their cell loss probability increased in the region in which the marking probability of abusive sources is below the ideal, and they will be reduced (even below GOS levels) for  $\sigma$  greater than the ideal curve crosspoint (see Figure 4.23).

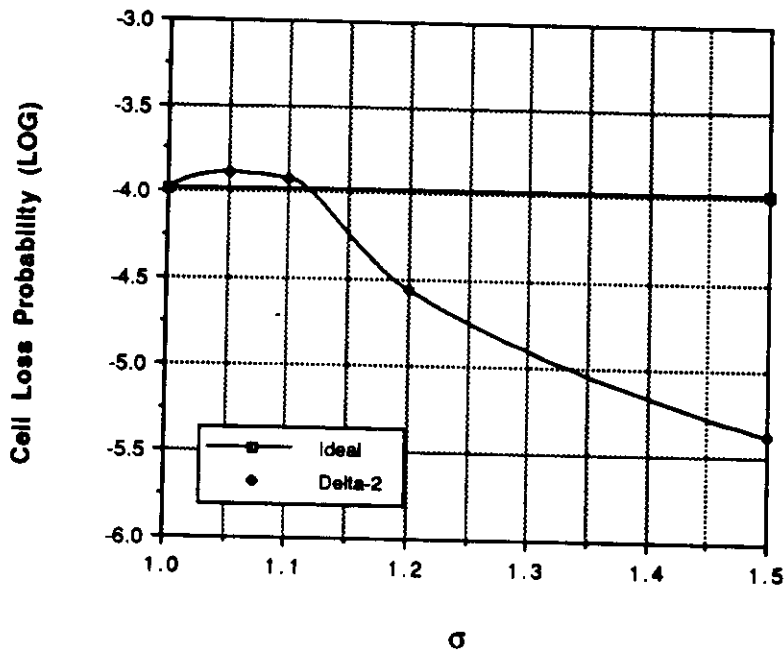


Figure 4.23: Marking effect on well behaved sources for Delta-2 ( $\rho_n = 0.8$ ).

Figure 4.23 was obtained for 17 abusive sources and 17 well-behaved ones. The reduction on well behaved sources cell loss probability is less evident for a reduced number of abusive sources. We can expect this reduction to revert to an increase as we increase the number of abusive sources or  $\sigma$ . While the first effect was caused by the excessive marking of cells for abusive sources, the later is caused by the excessive marking of cells for well-behaved sources. This can be explained by the fact that the well-behaved sources marked cells will have to compete with

the large number of marked cells from the abusive sources, eventually loosing the contention.

#### 4.4.4.3 Lower Bound on Reaction Time

The expressions for the lower bound on reaction time for the delta-2 mechanism are basically the same as the ones for the delta mechanism (Equations 4.11 and 4.12). In fact, the main distinction among the mechanisms are the values of threshold and average counter's average length.

Therefore,

$$RT_{DT-2}^b = \frac{T_a \cdot n_{cell}}{B_p - B_a} + D_{pk} \quad (4.13)$$

and

$$\overline{RT}_{DT-2}^b = \frac{(T_a - \overline{Q})n_{cell}}{B_p - B_a} + D_{pk} \quad T_a > \overline{Q} \quad (4.14)$$

Table 4.6 reports the reaction times for a Delta-2 with the same parameters as used for Figure 4.22 (i.e.,  $T_a = 450$ ,  $B_a = 1.25$  Mbps,  $Q=5K$ ,  $\overline{Q} = 421$ , and  $D_{pk} = \{8.48, 12.72, 16.96, 21.20\}$ sec).

Table 4.6: Delta-2 Reaction Time (lower bound)

| $D_{pk}$ (sec) | $RT_{DT-2}^b$ (sec) | $\overline{RT}_{DT-2}^b$ (sec) |
|----------------|---------------------|--------------------------------|
| 8.48           | 8.502               | 8.481                          |
| 12.72          | 12.740              | 12.721                         |
| 16.96          | 16.980              | 16.960                         |
| 21.20          | 21.220              | 21.200                         |

From Equations 4.13 and 4.14 and from Table 4.6, the time to fill-up the peak counter is the major factor on delta-2 reaction times.

#### 4.4.5 Delta-3 mechanism analysis under bursty traffic

As we mentioned in Section 4.3.6, this scheme has basically five parameters:

- the depletion rate of the average counter ( $B_e$ );
- the increasing rate of the gap counter ( $B_i$ );
- the depletion rate of the gap counter ( $B_d$ );
- the maximum value of the gap counter ( $S_{max}$ ); and
- the depletion interval ( $W$ ).

$B_e$  must be chosen slightly higher than  $B_m$  so that in average, the average counter is zero (empty) during a fraction  $\frac{B_e - B_m}{B_e}$  of the time. Furthermore, the ratio among  $B_i$  and  $B_d$  is such that  $S_{max}$  be depleted during the interval  $W$ .

##### 4.4.5.1 Marking/Loss probability

Figure 4.24 reports the marking probabilities of the delta-3 mechanism for  $B_e = CB_m$ ,  $C = \{1.01, 1.05, 1.1, 1.15, 1.25, 2\}$ ,  $B_i = 100\text{Mbps}$ ,  $W = 42.4$  seconds (=1 Mcells).  $S_{max}$  and  $B_d$  were obtained by using Equations 4.2 and 4.3. These results were obtained through simulation for a simulated time equivalent to two hours.

From Figure 4.24 we conclude that for these parameters, we should use  $B_e = 1.1B_m$ .

For the same cell interarrival distribution, an increase on the depletion rate of the average counter ( $B_e$ ) increases the amount of time the average counter is empty between bursts. As a consequence, the probability of finding the gap counter empty is reduced, and, therefore, the cell marking probability is reduced.

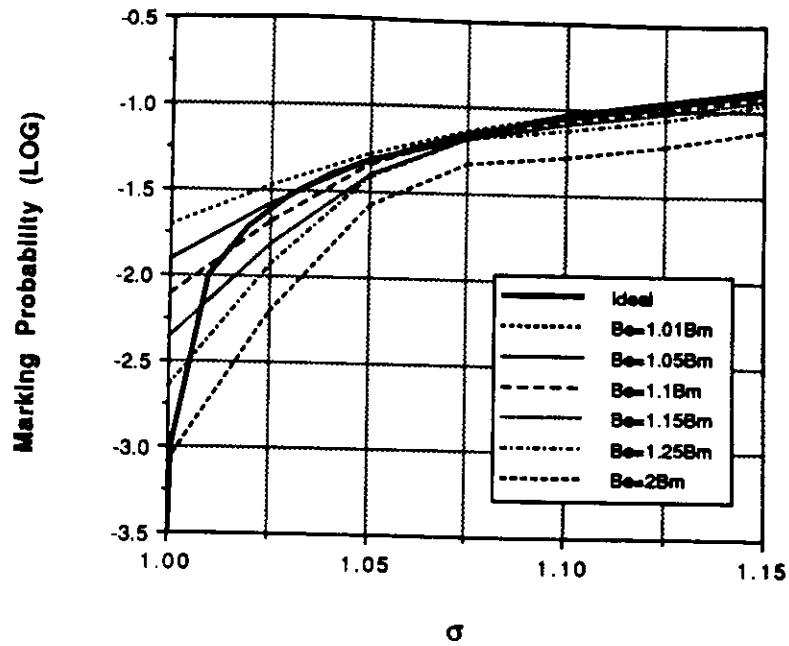


Figure 4.24: Delta-3 Marking Probabilities — 2 hours equivalent simulated time.

The ideal value of  $B_e$  is a function of the traffic burstiness. To confirm this assertion, compare Figure 4.24 for a source with  $b = 10$ , with Figure 4.36 for a VBR source with burstiness 2.71.

#### 4.4.5.2 Marking effect on well behaved sources

As we did with leaky bucket, we simulated the system of Figure 4.2 for 17 well-behaved sources, and 17 abusive ones, for an equivalent simulated time of 1,000 seconds. Marking probabilities are reported in Figure 4.25. Contrary to the previous mechanisms, the marking probabilities were just slightly changed when we reduced the simulated time. This may be explained by the fact that unless the traffic is too light, the gap counter will be at 0 or close to it, thus being almost ready to mark any abusive cell in a short term.

Figure 4.26 reports the corresponding cell loss probabilities for well-behaved

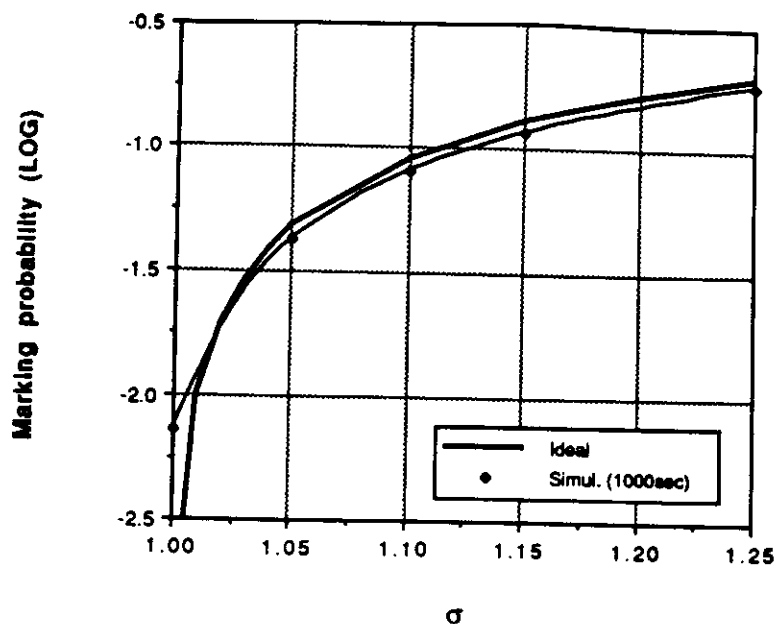


Figure 4.25: Delta-3 Marking Probabilities ( $\rho_n = 0.91$ ) - 1000 sec equivalent simulated time.

sources. Two factors contribute to the increase in cell loss probability with the increase in average rate of the abusive sources. The first is the relatively high percentage of nominal average rate cells that are marked, which will have to compete with the marked cells from the abusive sources for a space on the MUX buffer. The second is that not all abusive cells are being marked.

#### 4.4.5.3 Reaction Times

Contrary to other IRC mechanisms that before start marking have to fill-up a "bucket" or have a counter reach a maximum value, delta-3 starts marking as soon as the accumulated credits are not enough for the size of the arriving burst. We may assume that at the beginning the delta-3 mechanism accumulates credits correspondent to the period until the first talkspurt is generated which in average is

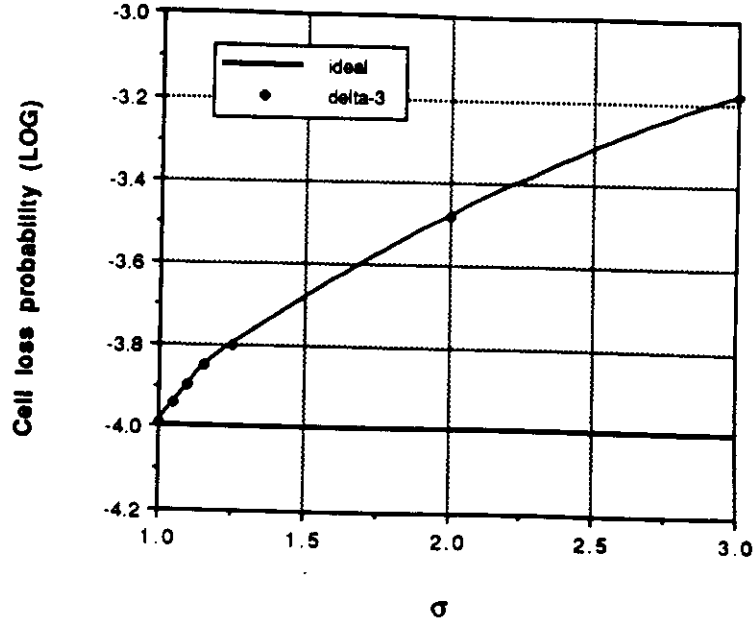


Figure 4.26: Marking effect on well behaved sources for Delta-3 ( $\rho_n = 0.91$ )

equal to the average silence period (talkspurt average interarrival time). Therefore,

$$RT_{DT-3}^b = S \frac{B_e}{B_e - B_m} \quad (4.15)$$

Even though we can make up scenarios where reaction times are longer, we will consider as an upper bound on reaction time, the time required by the gap counter to decrease from  $S_{max}$  to 0, which is given by

$$RT_{DT-3}^{b-UB} = S_{max} \cdot n_{cell} / B_d \quad (4.16)$$

which by construction (Equations 4.2 and 4.3) gives

$$RT_{DT-3}^{b-UB} = W \quad (4.17)$$

## 4.5 Analysis of IRC mechanisms under VBR traffic

### 4.5.1 Jumping Window analysis under VBR traffic

#### 4.5.1.1 Marking/Loss probability

Figure 4.27 shows the jumping window marking probabilities for our VBR source, and  $W = 1$  Kframes, which corresponds to 33.3 seconds. Note that once again, the marking probabilities just slightly changed when equivalent simulated time was reduced from 2 hours to 1,000 seconds.

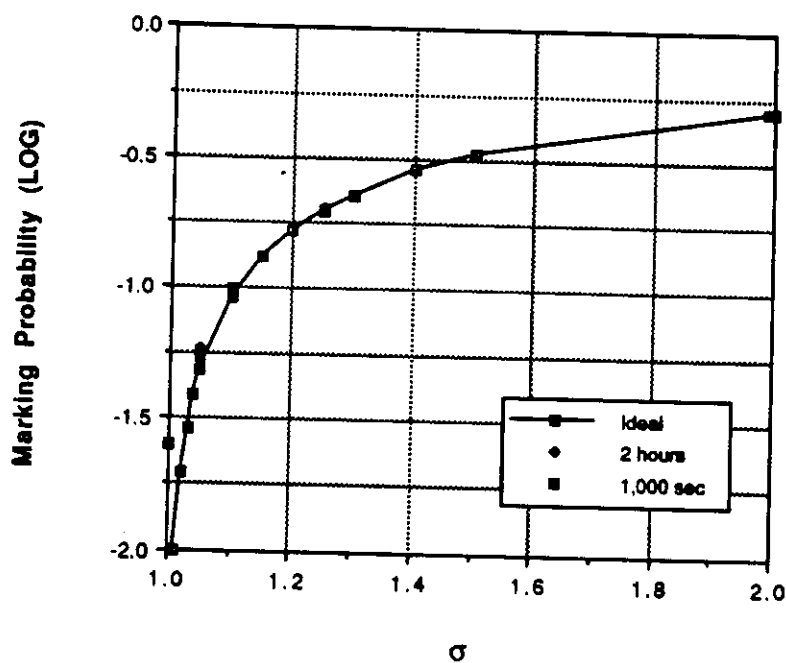


Figure 4.27: Jumping Window Marking Probabilities for a VBR Source.

#### 4.5.1.2 Marking effect on well behaved sources

Figure 4.28 shows the deleterious effect of marking cells on well behaved sources. These results were obtained for 1 abusive source out of 22.



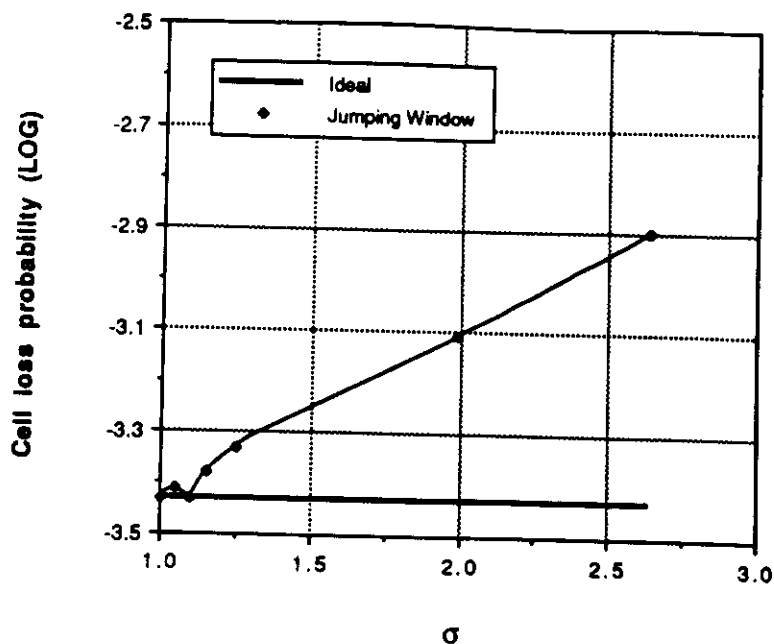


Figure 4.28: Effect of Marking cells on well behaved sources (Jumping Window).

#### 4.5.2 Leaky Bucket analysis under VBR traffic

In this Subsection we consider the Variable Bit Rate (VBR) Traffic described in Section 3.2.3.

As we did in the bursty traffic case, we will start our study by considering the strict requirement that the marking/loss probability be achieved as an equality, at nominal average bit rate. Again we use the UAS method to obtain the “marking/loss” probability of cells for a given set of leaky bucket parameters. For a given maximum counter value  $Q$ , the depletion is chosen so that the desired GOS is achieved as an equality, at nominal average bit rate.

Figure 4.29 shows the variation of the cell marking/loss probability with  $\sigma$  for several maximum counter values  $Q$ , for a VBR source.

Similarly, Table 4.7 gives the depletion rate  $B_e$ , the overdimensioning factor  $C$ , and the average counter size  $\bar{Q}$  for each of these curves.

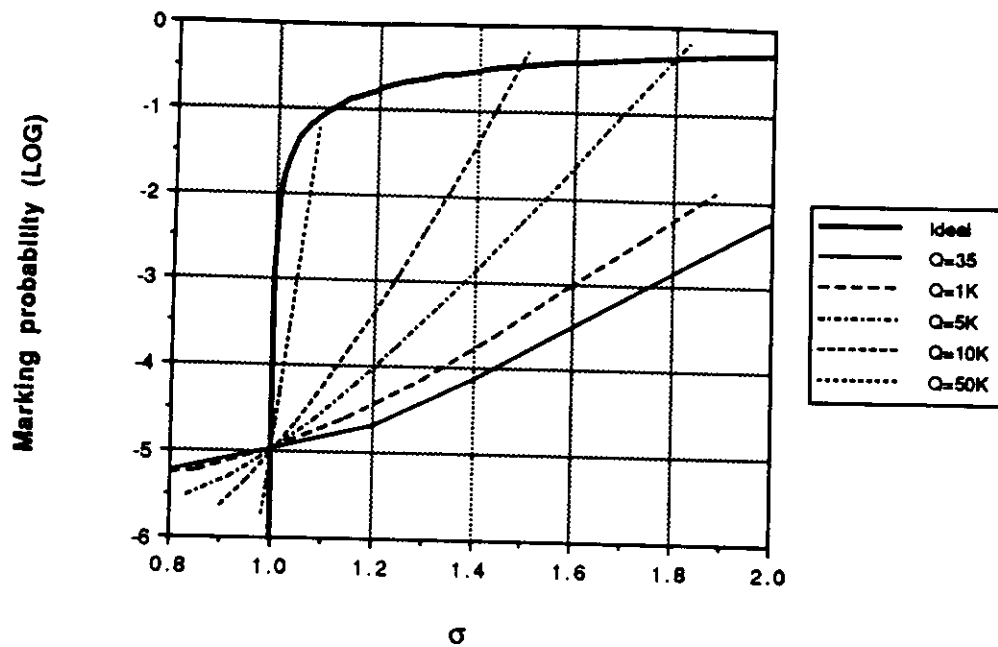


Figure 4.29: Leaky Bucket Behavior for a VBR Source ( $GOS = 10^{-5}$ ).

Table 4.7: Leaky Bucket Parameters for VBR Traffic ( $GOS = 10^{-5}$ )

| $Q$ (cells) | $B_e$ (bits/sec) | $C$   | $\bar{Q}$ (cells) |
|-------------|------------------|-------|-------------------|
| 35          | 11,871,010       | 3.044 | 0.0               |
| 1,000       | 9,675,616        | 2.481 | 0.2               |
| 5,000       | 7,125,012        | 1.827 | 16.4              |
| 10,000      | 5,856,749        | 1.502 | 124.9             |
| 50,000      | 4,225,325        | 1.083 | 2,986.5           |

Comparing these results with the bursty source results, we note that in order to achieve the same cell loss behavior we need larger pseudo queue sizes, and higher depletion rates. We also observe that the average queue lengths are shorter. This can be explained by the fact that in our examples while the bursty source had a burstiness  $b = 10$ , the VBR source has a burstiness of only  $b = 2.71$ .

Table 4.8 shows the cell loss/marking probability ( $P$ ) for our VBR source and leaky bucket depletion rate  $B_e = 3,900,001$  bps. (Note that  $B_e$  was chosen slightly higher than  $B_m$  so that the utilization factor would be less than one, and the UAS model could be used to obtain the cell loss/marking probability). As one can see, in order to achieve the desired GOS ( $10^{-9}$ ), we have to use leaky buckets with sizes around 400,000 cells (which corresponds to a reaction time of 24.94 seconds).

Note that these results correspond to sources with steady average bit rate  $B'_m = \sigma B_m$ . In the Variable Bit Rate case, however, the bit rate will vary around a certain long term average value. So, the leaky bucket scheme cannot be directly applied (as it was in the bursty traffic case) to enforce a predefined input rate. We recall that in a VBR session we can identify two critical rates: the long term average rate, and the peak rate. Accordingly, we propose to use two LB's: one small bucket to promptly cut off input rates which exceed the peak rate, and a very large bucket to keep track of the long term average. The latter bucket should be large enough to absorb the rate fluctuations of the VBR signal around the long term average.

#### 4.5.2.1 Analytical versus simulation results

Figures 4.30 and 4.31 compare analytical with simulation results for a leaky bucket with  $Q = 1,000$  and  $Q = 10,000$ , respectively, for our VBR source.

At each simulation point, the dots correspond to a 95% confidence interval of

Table 4.8: Cell loss/marking probability for  $B_e \approx B_m$  (VBR source).

| $Q$ (cells) | $\log_{10} P$ | $\bar{Q}$ |
|-------------|---------------|-----------|
| 35          | -0.029        | 0.08      |
| 100         | -0.032        | 0.58      |
| 1,000       | -0.060        | 29.12     |
| 5,000       | -0.153        | 520.01    |
| 10,000      | -0.268        | 1,721.99  |
| 50,000      | -1.182        | 12,843.64 |
| 100,000     | -2.325        | 16,820.18 |
| 200,000     | -4.610        | 17,378.48 |
| 300,000     | -6.895        | 17,383.81 |
| 400,000     | -9.181        | 17,383.85 |
| 500,000     | -11.466       | 17,383.85 |
| 1,000,000   | -22.894       | 17,383.85 |

the cell loss/marking probability ( $P$ ). We can see that the accuracy of the UAS model for the VBR sources is reasonably good.

The same phenomenon as with bursty sources happens here, due to the inaccuracies of the model for long buffers.

#### 4.5.2.2 Marking probability revisited

As in Subsection 4.4.2.5 we now remove the strict requirement that average rate cells be deleted by the leaky bucket at GOS levels. We are able to do this by using a marking scheme that can even mark legitimate cells, as long as this does

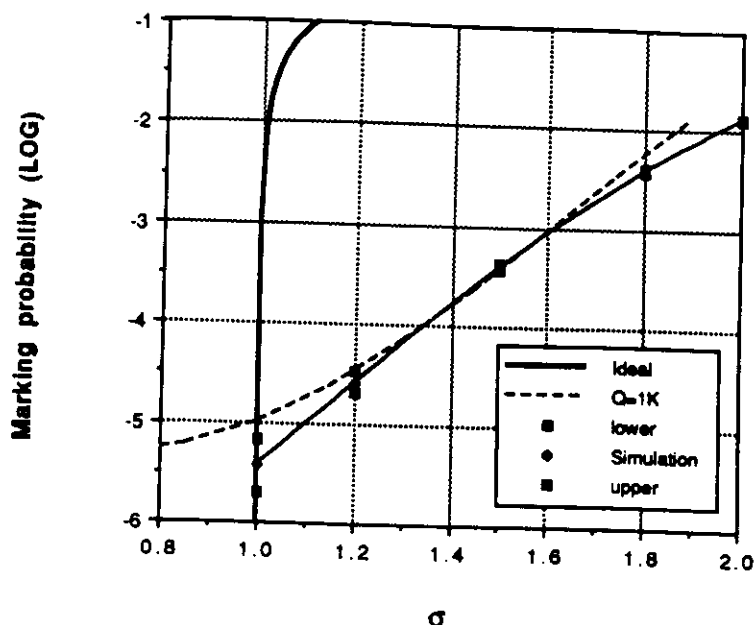


Figure 4.30: Analytical versus Simulation Results for a VBR Source Leaky Bucket with  $Q=1K$ .

not significantly affects the cell loss probability of well-behaved sources.

Figure 4.32 compares the cell marking probabilities for the LB working at nominal average bit rate (i.e.,  $B_e = B_m$ ), with the ideal marking probability, for VBR sources. These results were obtained through simulation. The simulations were run for a fixed amount of simulated time equivalent to two hours.

Even though the marking probabilities for these two LBs show a remarkable match with the ideal curve for  $\sigma > 1$ , it is still too high for the legitimate traffic. Therefore, a marking scheme is a better solution than a deletion one.

#### 4.5.2.3 Marking effect on well-behaved sources

In this subsection we study the deleterious effect of a single abusive source on the cell loss probability of 22 well-behaved ones.

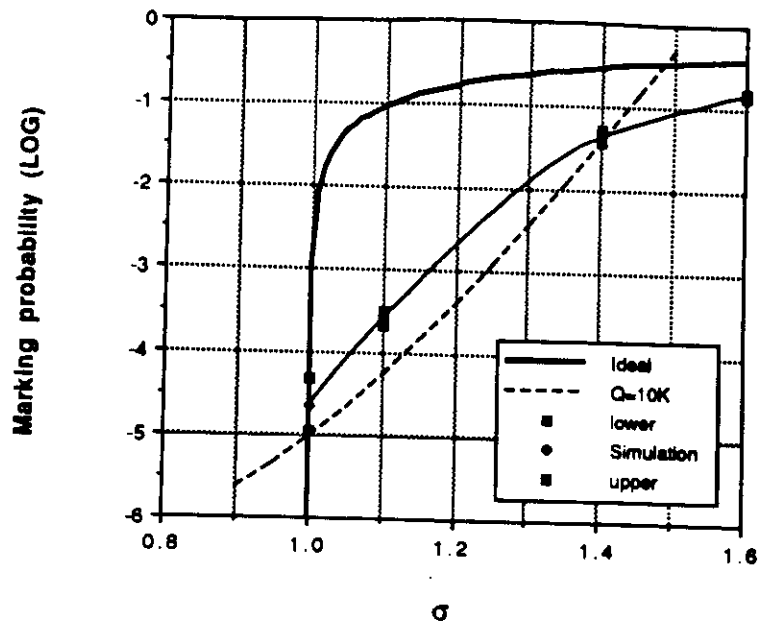


Figure 4.31: Analytical versus Simulation Results for a VBR Source Leaky Bucket with  $Q=10K$ .

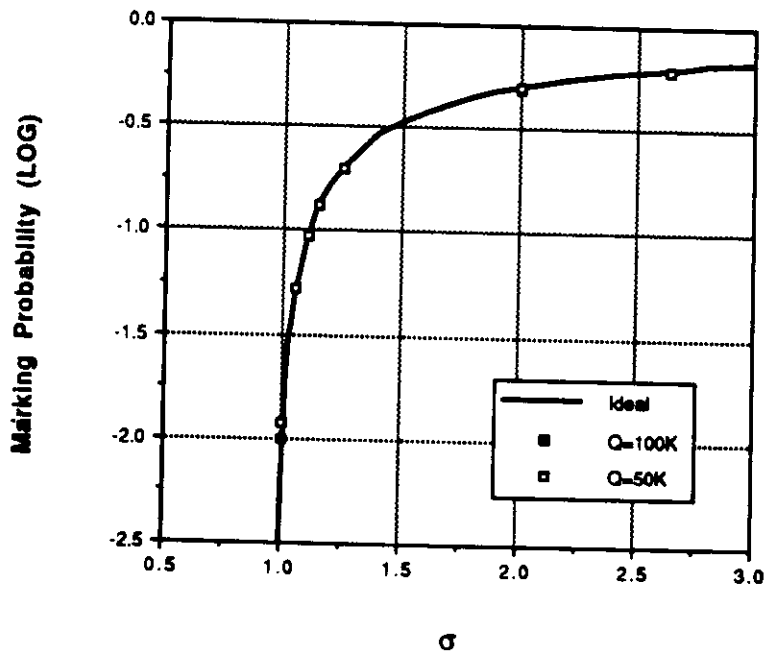


Figure 4.32: Leaky Bucket Marking Probabilities for a VBR Source ( $B_e = B_m$ ).

Figure 4.33 shows the marking probability for our VBR source for a reduced observation period of 1,000 seconds, for LBs of sizes  $Q = 50K$  and  $Q = 100K$ . Even though the marking probability is lower than the ideal, it is still very good. This is probably due to the lower burstiness (as compared to our bursty source).

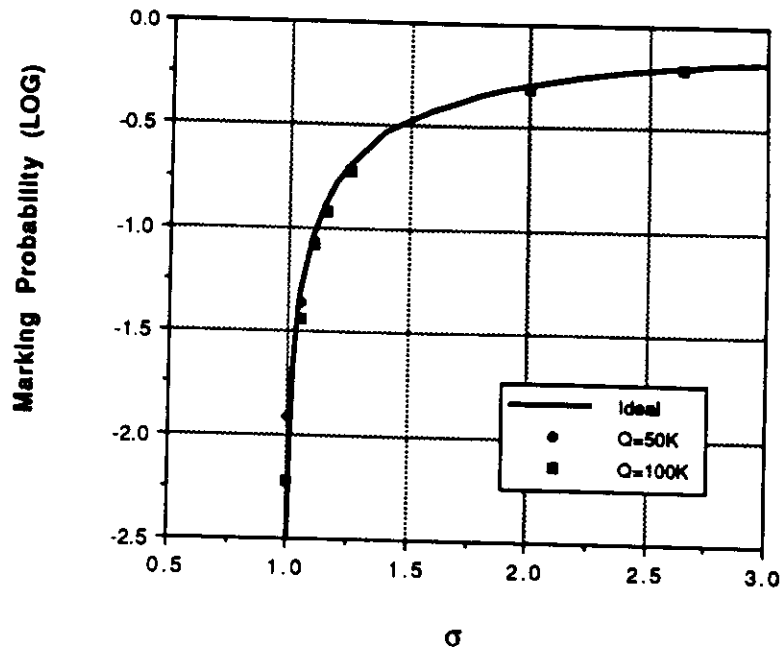


Figure 4.33: Leaky Bucket Marking Probabilities for a VBR Source ( $B_e = B_m$ ) - 1,000 seconds equivalent simulation time.

In Figure 4.34 we show the deleterious effect of marking cells on well behaved VBR sources. This effect becomes more significant as the number of abusive sources increases.

#### 4.5.2.4 Reaction Times

A lower bound on the reaction time can be obtained by assuming that the source would continuously generate cells at maximum bit rate until filling-up the leaky bucket. Actually, we could have two measures. One would assume an empty

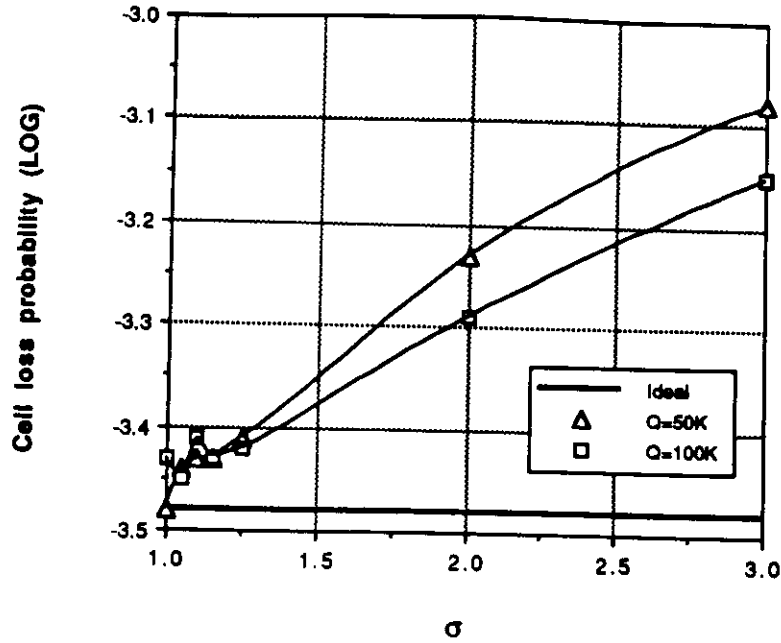


Figure 4.34: Effect of Marking cells on well behaved VBR sources (Leaky Bucket).

initial state, while the other would assume the average state as the initial one. Let us call them  $RT_{LB}^{ubr}$  and  $\overline{RT}_{LB}^{ubr}$ , respectively. Therefore, we have:

$$RT_{LB}^{ubr} = \frac{Q \cdot n_{cell}}{B_{max} - B_e} \quad (4.18)$$

and

$$\overline{RT}_{LB}^{ubr} = \frac{(Q - \overline{Q}) \cdot n_{cell}}{B_{max} - B_e} \quad (4.19)$$

#### 4.5.3 Delta-2 mechanism analysis under VBR traffic

In this Subsection, we compare the marking/loss probability of the delta-2 mechanism with that of an ideal policing mechanism, for our VBR source. Again, the general behavior of the delta-2 mechanism is that of marking abusive cells at a probability smaller than the ideal until the utilization approaches one. After this point, the marking probability is much higher than the ideal, therefore, severely penalizing sources that generate abusive traffic.



### 4.5.3.1 Marking probability

Now, we obtain the marking probabilities at  $\rho_n = 0.8$ . The UAS model with a single source is not very precise, specially at high utilizations. For these parameters it gave an average counter length of 658 with  $Q \geq 30K$ , while simulation produced the value  $846 \pm 35$ . Based on the simulation result, we chose  $T_a = 850$ . Figure 4.35 shows the marking probabilities for the delta-2 mechanism with  $T_a = 850$ ,  $Q=30K$ , and  $D_{pk} = \{10, 20\}$ sec.

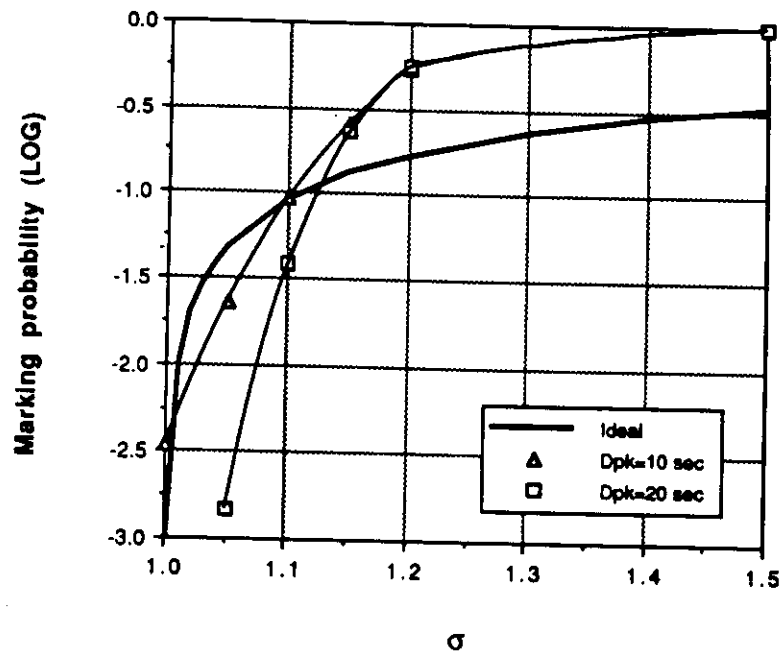


Figure 4.35: Delta-2 Marking Probabilities (VBR source;  $\rho_n = 0.8$ ).

Comparing Figure 4.35 with Figure 4.22, we note that the delta-2 marking probability for our VBR source with  $D_{pk} = 10$  seconds basically have the same behavior as the delta-2 marking probability for the bursty source with  $D_{pk} = 6.36$  seconds. This can be explained by the fact that being the VBR source less bursty than the bursty one ( $b = 2.71$  as opposed to  $b = 10$ ), it is more likely to have the average counter above its threshold for longer periods than in the bursty case.

#### 4.5.3.2 Marking effect on well-behaved sources

Even though we did not run simulation to obtain the effect of marking cells on the cell loss probability of well-behaved sources, we can expect that the same results for bursty sources should be observed, since the marking probabilities follow the same pattern.

#### 4.5.3.3 Lower Bound on Reaction Time

Again we use the same expressions as for the bursty case, except that we substitute  $B_p$  by  $B_{max}$ :

$$RT_{DT-2}^{vbr} = \frac{T_a \cdot n_{cell}}{B_{max} - B_a} + D_{pk} \quad (4.20)$$

and

$$\overline{RT}_{DT-2}^{vbr} = \frac{(T_a - \overline{Q})n_{cell}}{B_{max} - B_a} + D_{pk} \quad (4.21)$$

#### 4.5.4 Delta-3 mechanism analysis under VBR traffic

Figure 4.36 reports the marking probabilities of the delta-3 mechanism for our VBR source and parameters  $B_e = CB_m$ ,  $C = \{1.01, 1.05, 1.1, 1.15, 1.25, 2\}$ ,  $B_i = 430.625\text{Mbps}$  ( $=100 B_m$ ),  $W = 33.3$  seconds ( $=1$  Kframes).  $S_{max}$  and  $B_d$  were obtained by using Equations 4.2 and 4.3. These results were obtained through simulation for a simulated time equivalent to two hours.

Again, from Figure 4.36 we conclude that for those parameters, we should use  $B_e = 1.1B_m$ .

Even though we did not run simulation to obtain the effect of marking cells on the cell loss probability of well-behaved sources, we can expect that the same results for bursty sources should be observed, since the marking probabilities follow the same pattern.

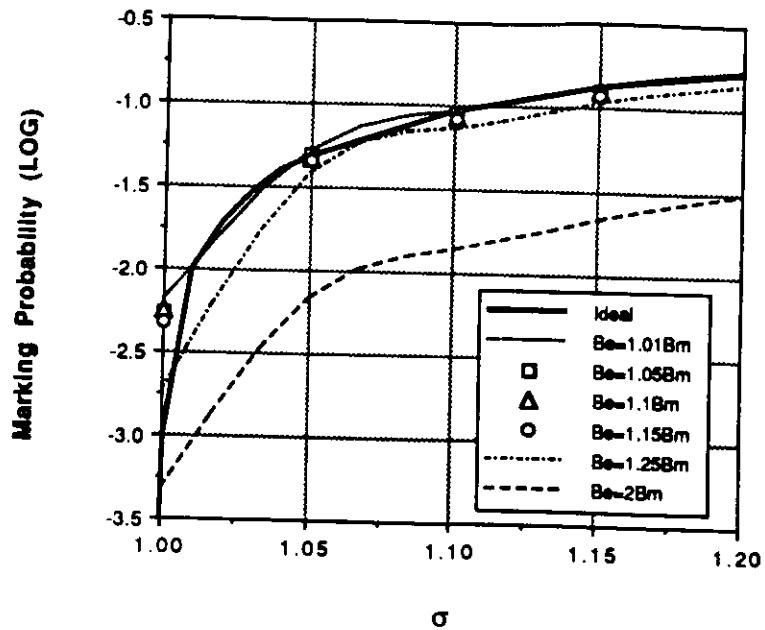


Figure 4.36: Delta-3 Marking Probabilities for a VBR source — 2 hours equivalent simulated time .

#### 4.6 IRC Mechanisms Comparison

In this Section we compare the effectiveness of the policing mechanisms studied in previous Sections.

The criteria for our comparison are:

- Conformity to the ideal marking probabilities;
- Effect on well-behaved sources;
- Reaction time; and
- Implementation complexity.

### 4.6.1 Conformity to the ideal marking probabilities

Figure 4.37 compares the marking probabilities of jumping window, leaky bucket, delta, delta-2, and delta-3 mechanisms with the marking probability of the ideal IRC mechanism, for a bursty source.

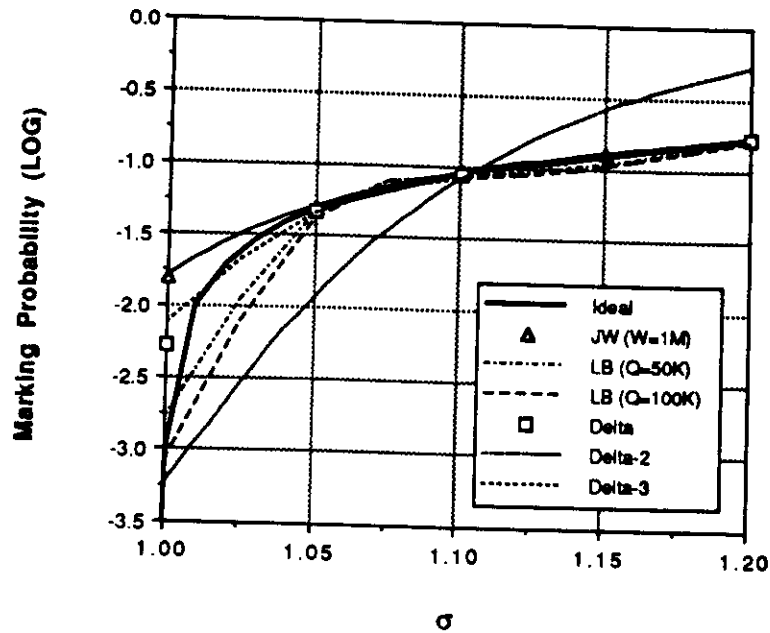


Figure 4.37: Marking probabilities comparison for IRC mechanisms (bursty source).

From all the considered mechanisms, leaky bucket is the one that better conforms to the ideal IRC behavior. It combines a good match for average bit rates above nominal, with a relatively low marking probability at nominal average bit rate.

### 4.6.2 Effect on well-behaved sources

Figure 4.38 compares the effect of marking probabilities on the cell loss probabilities of well-behaved sources of the ideal IRC mechanism with jumping window,

leaky bucket, delta-2, and delta-3 mechanisms, for a bursty source. These results were obtained through simulation where the number of well-behaved and abusive sources were both seventeen, and the simulated time was equivalent to 1,000 seconds.

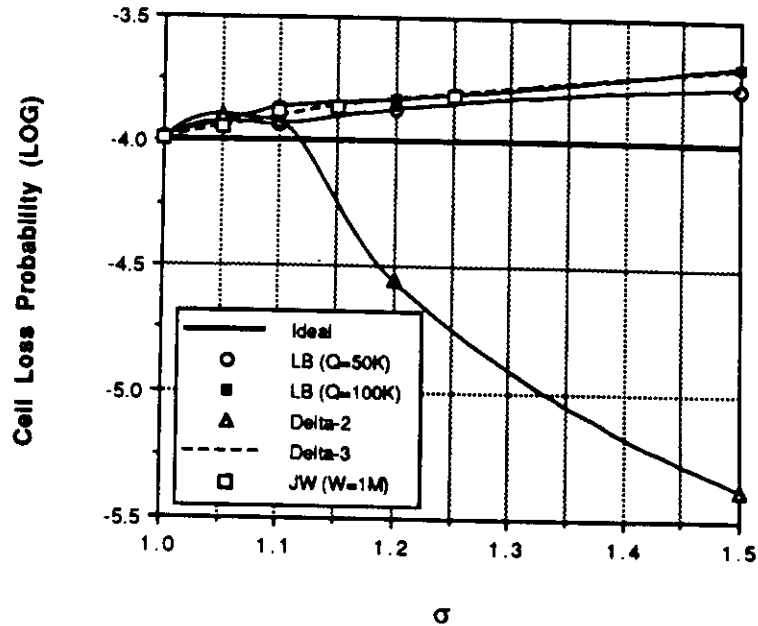


Figure 4.38: Effect on well behaved sources comparison for IRC mechanisms (bursty source).

From Figure 4.38 we can conclude that jumping window, leaky bucket, and the delta-3 mechanism have an equivalent effect on the cell loss probability of well behaved sources, while delta-2 although worse for  $1.0 < \sigma < 1.1$ , presents even a reduction with respect to the ideal cell loss probabilities for  $\sigma > 1.2$ .

Therefore, we conclude that the delta-3 mechanism is the better among them as far as the effect on well-behaved sources is concerned.

### 4.6.3 Reaction time

Figure 4.39 compares the lower bound on reaction times for jumping window, leaky bucket, delta, delta-2, and delta-3 mechanisms. Let us recall that we defined reaction time as the amount of time that the given mechanism requires to go from a predefined state (“empty”, or average) to the marking state. In order to obtain a lower bound, we assume that cells arrive at peak bit rate until the marking state is reached.

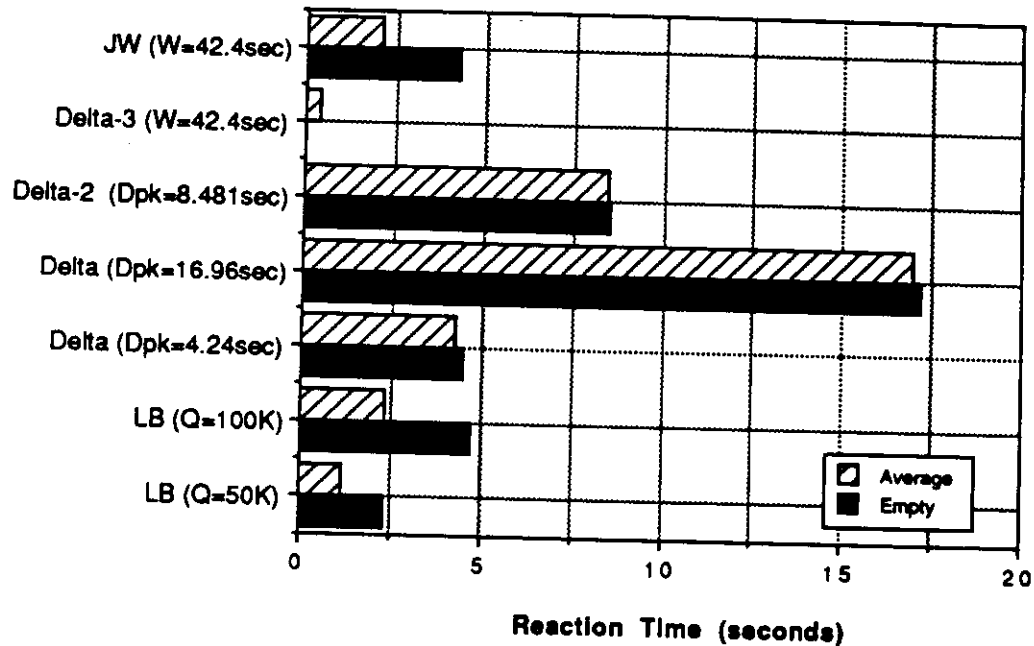


Figure 4.39: Reaction times comparison for IRC mechanisms (bursty source).

Reaction time for leaky bucket is linearly dependent on the bucket size. Both queue sizes (i.e., 50K and 100K) presents almost the same marking characteristics (see Figure 4.37), and therefore, we can use the 50K one for the comparison. The same applies to the delta mechanism. Therefore, we can rate the IRC mechanisms according to reaction times as:  $\text{delta-3} < \text{LB} < \text{jumping window} < \text{delta} < \text{delta-2}$ .

#### 4.6.4 Implementation complexity

In this Section we evaluate the mechanism complexity by the number of “basic” hardware elements they require. The “basic” hardware elements are: counters, comparators, and rate generators.

The counters are registers that hold the counter value, and are updated according to the arrival of a cell, or a given bit rate. The comparator compares counter values to values stored in registers. These values represent thresholds or maximum counter values, and have to be set according to the traffic currently being monitored. The rate generators which may be only programmable divisors, are responsible for generating the counter bit rates.

Here, we are not concerned with the implementation details of these “basic” elements. They could be even implemented just in software, as long as time constraints are not violated.

Table 4.9 list the number of such “basic” elements for each of the studied IRC mechanisms. The Delta-3 mechanism practically has only 1 comparator, since the comparison with 0 does not need to be programmed. In the other hand, it needs 3 rate generators. Therefore, we can rate their relative complexity as: Jumping Window = Leaky Bucket < Delta < Delta-3 < Delta-2. The exact order depends on the relative costs of comparators and rate generators.

#### 4.6.5 Summary

Table 4.10 summarizes the comparisons made in previous subsections.

Even though leaky bucket gets a good rating in most of the criteria, including implementation complexity, which is crucial in obtaining its cost, we argue that still the most important criteria is the effect on well-behaved sources, as far as

Table 4.9: "Basic" Elements for IRC Mechanisms.

| Mechanism      | Counters | Comparators            | Rate Generators       |
|----------------|----------|------------------------|-----------------------|
| Jumping Window | 1        | 1 ( $N$ )              | 1 ( $window$ )        |
| Leaky Bucket   | 1        | 1 ( $Q$ )              | 1 ( $B_e$ )           |
| Delta          | 2        | 2 ( $T_a, T_{pk}$ )    | 2 ( $B_e, B_{pk}$ )   |
| Delta-2        | 2        | 3 ( $T_a, Q, T_{pk}$ ) | 2 ( $B_e, B_{pk}$ )   |
| Delta-3        | 2        | 2 ( $0, S_{max}$ )     | 3 ( $B_e, B_i, B_d$ ) |

Table 4.10: IRC Mechanisms Comparison Summary.

| Mechanism      | Conformity | Effect<br>on WBS | Reaction<br>Time | Implement.<br>Complexity |
|----------------|------------|------------------|------------------|--------------------------|
| Jumping Window | Good       | Poor             | Good             | Excellent                |
| Leaky Bucket   | Excellent  | Poor             | Good             | Excellent                |
| Delta          | Fair       | Poor             | Fair             | Fair                     |
| Delta-2        | Poor       | Excellent        | Poor             | Poor                     |
| Delta-3        | Fair       | Poor             | Excellent        | Fair                     |



the design of broadband networks is concerned. In this case, a mechanism with a marking probability like delta-2 is more desirable than leaky bucket.

#### 4.7 Conclusions

In broadband networks built using ATM (Asynchronous Transfer Mode) technology, we need not only to allocate enough bandwidth to each connection, in order to achieve a desired grade of service (GOS). We need also to ensure that the sources abide by the declared rates or else well-behaved traffic will suffer.

Input Rate Control (IRC) mechanisms were devised to enforce such predefined maximum rates. Even though, the sources themselves can monitor their traffic and deliver only at the predefined rates, the network still must perform a repressive function to protect well-behaved sources.

The most popular input rate control for broadband networks reported in the literature, namely, leaky bucket, was shown not to be adequate for deletion of abusive traffic cells. A "virtual" marking scheme proposed elsewhere was adopted. In this scheme, abusive traffic cells are not deleted but simply marked. In case of congestion, marked (low priority) cells have to defer to unmarked (high priority) ones. The performance of leaky bucket and three new mechanisms (delta, delta-2, and delta-3) were studied and compared, both for a bursty source and for a variable bit rate (VBR) traffic.

We conclude that the delta-2 mechanism, even though not conforming to the ideal marking probability behavior, nor being the simplest one, has the nice property of "overpenalizing" abusive traffic sources, by marking more than the abusive cells for large offending factors. As a result, the cell loss probability of well-behaved sources may even be reduced below the normally expected levels.



## Chapter 5

### B-ISDN Design Problem

Previous work on design of communication networks concentrated into two areas according to the switching technology: circuit or packet switching.

In circuit switching (C/S) networks, channel bandwidth is dedicated to a connection while it lasts. Since the required bandwidth is dedicated to the connection the only cause of data loss are either noise or equipment failure. Circuit switched networks are usually designed in order to minimize call blocking probabilities (i.e., the probability that a new call be rejected because not enough bandwidth is available). In the case of multiservice networks, either a fixed amount of bandwidth is reserved for each service, or a minimum amount is reserved for each service and the remaining bandwidth is shared among all services. This reservation is necessary in order to assure a minimum availability, and avoid *fairness* problems.

In the other side of the spectrum, there are packet switched (P/S) networks, where all channels are shared, and packets are routed through the channels either in a packet-by-packet basis (*datagram* service), or a *virtual circuit* is established at call set-up and all packets that belongs to the same connection follow the same path (the virtual circuit). This circuit is called "virtual" because no actual bandwidth is dedicated to it. Since in principle a large number of connections (or packets in connectionless services) can be accepted without blocking, the design of packet switched networks has usually as a goal to minimize average packet delay. The larger the congestion, the larger the average packet delay. In case of congestion packets can eventually be lost.

Packet switched networks are more flexible than circuit switched ones, because there is no need to dedicate bandwidth to a connection (allowing for statistical multiplexing), and furthermore, it is not tied to the use of multiples of some basic rates (hierarchies).

In Section 5.1 we present the broadband network architecture that we are assuming in this work. Section 5.2 formulates and sketches solutions for some of the design problems associated with ATM networks. In Section 5.3 we consider one of these problems: the reconfiguration of the network embedded in a facility network; which is solved in Section 5.4 as a P/S design problem. In Section 5.5 we explore some of the aspects involved in the design of ATM networks as C/S nets. Finally, Section 5.6 summarizes the Chapter and gives directions for further work.

## 5.1 Broadband Network Architecture

Batorsky et al. [BST88] proposed a long-term architecture as a double star topology consisting of the following elements (refer to Figure 2.5):

**Interworking unit (IWU)** - located at customer's premises, provides for the conversion between various customer interfaces and the integrated user-network interface (UNI).

**Remote Multiplexer (RM)** - (T1S1.1 contribution) used in order to increase sharing in the distribution plant. It is connected to the customers in a star topology.

**Remote Electronics (RE)** - aggregates traffic to the Central Office.

**Central Office (CO)**

**Hub Offices** - provides the interconnection between CO locations.

Table 5.1 shows which kind of topology can be used at each interconnection level. Basically, it is possible to have a star, a bus, or a ring at any level but in the distribution level, where a star topology is expected.

Table 5.1: Interconnection topology per level

| level                 | star | bus | ring |
|-----------------------|------|-----|------|
| distribution (IWU-RM) | x    | -   | -    |
| sub-feeder (RM-RE)    | x    | x   | x    |
| feeder (RE-CO)        | x    | -   | x    |
| interoffice (CO-Hub)  | x    | x   | x    |

As far as cost is concerned, at any level there is a trade-off between the cost of laying down another fiber and the cost of using WDM on a single one. Short distances favors parallel fibers, while long distances, may favor a single fiber.

Other trade-offs must also be considered such as moving the active electronics in the RE back to the CO and use multiple wavelengths on the RE-CO fibers.

We propose an architecture in which a hub office is not necessary. All ATM switches (CO's) are at the same level. We can use another ATM switch in the same metropolitan area as a relay to a long distance destination in cases where the CO traffic to that destination does not justify an extra channel. But there is no need that this CO will always be the same (the hub).

In this architecture we are ignoring the fact that due to a high community of interest (COI) traffic from a given set of users, could be homed directly to the correspondent CO, with a reduction in the required number of ports.

## 5.2 Design Problems

In this Section we formulate and sketch solutions for some of the design problems associated with ATM networks.

As for traditional P/S networks, we assume that a cost-effective structure for the ATM network is a multilevel hierarchical one consisting of a *backbone* network and a number of *distribution networks* attached to the ATM switches.

In order to reduce the design complexity we can partition the problem into two: the backbone network design and the distribution network design. Where the distribution network design is in fact a family of problems, one per each distribution network that emanates from an ATM switch.

The distribution network design problems are:

- “Concentrators” (RM’s & RE’s) location problem.
- “Terminals” layout problem.

Backbone network design problems:

- Switches location problem.
  - goals:
    - \* reduce feeder and backbone plant costs.
    - \* keep switches at reasonable sizes (non-linear cost).
  - tradeoff: single large switch vs. several smaller ones.
- DCS location problem.
  - goal: reconfiguration flexibility.
  - tradeoff: flexibility vs. DCS cost.

- potential locations:
  - \* potential switch locations.
  - \* fiber route bifurcations.

- Constrained Trunk layout (reconfiguration) problem.

Given:

- switch locations and capacities.
- link topology and capacities.
- traffic matrix(ces).

Minimize: cost.

Variables:

- trunk capacities and routing.

Constraints:

- trunk capacities do not exceed link capacities.

- Unconstrained Trunk layout problem.

Given:

- switch locations and capacities.
- potential traffic routes.
- traffic matrix(ces).

Minimize: cost.

Variables:

- trunk capacities and routing.

- Switch capacity assignment and trunk layout problem.

As above, except that switch capacities are not fixed.

### 5.3 Reconfiguration

The ability to reconfigure a customer network dynamically is a well known advantage of Digital Cross Connect Systems and has been reported extensively in the literature [YH88, Zan88, FP88]. Most of the previous studies, however, have been based on transparent, circuit-switched type networks in which channels of various rates are established end-to-end between pairs of user sites. The main goals in the design of such systems were the dynamic network reconfiguration following trunk failures, and the reassignment of trunks to applications following a predefined time schedule, or on a reservation basis, or in response to sudden traffic changes.

In this study, we are not concerned with the configuration of the transparent, circuit switched type network. Rather, we are interested in the ATM network built on top of the facility network. We want to exploit the DCS flexibility in order to obtain a more efficient design and operation of the ATM network.

To illustrate the point, consider the network shown in Fig. 5.1. From the original (backbone) topology, several embedded topologies can be derived. The embedded topology of Figure 5.2 is identical to the backbone topology, whereas the topology of Figure 5.3 has introduced a number of “express pipes” between remote nodes. Express pipes reduce the number of intermediate hops along the path and thus reduce store and forward delay and nodal processing overhead.

Express pipes also reduce the number of packet switch terminations. Note that the topology of Figure 5.2 requires 192 packet switch terminations while the



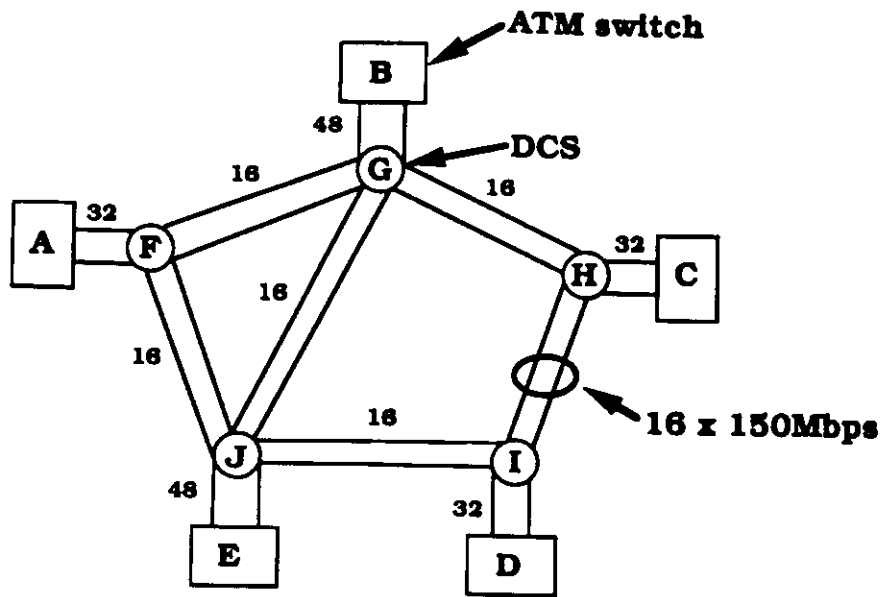


Figure 5.1: Backbone topology.

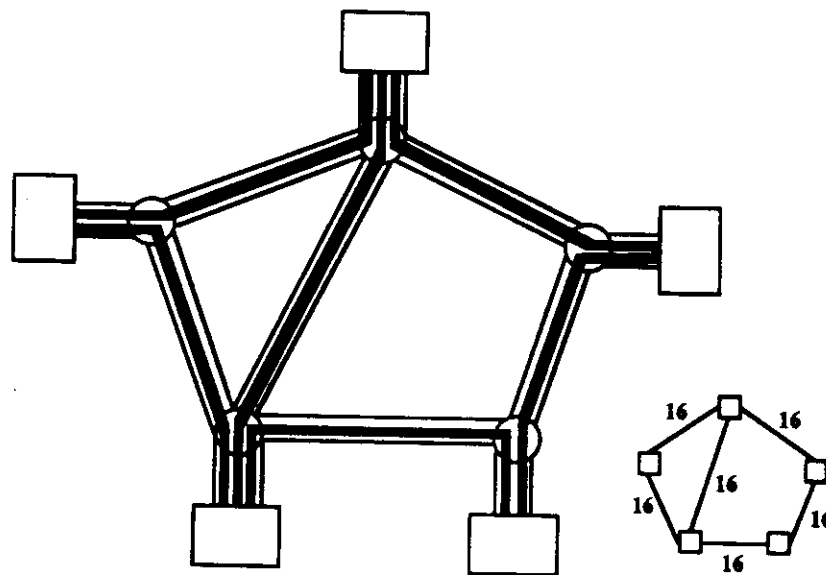


Figure 5.2: Embedded topology A.

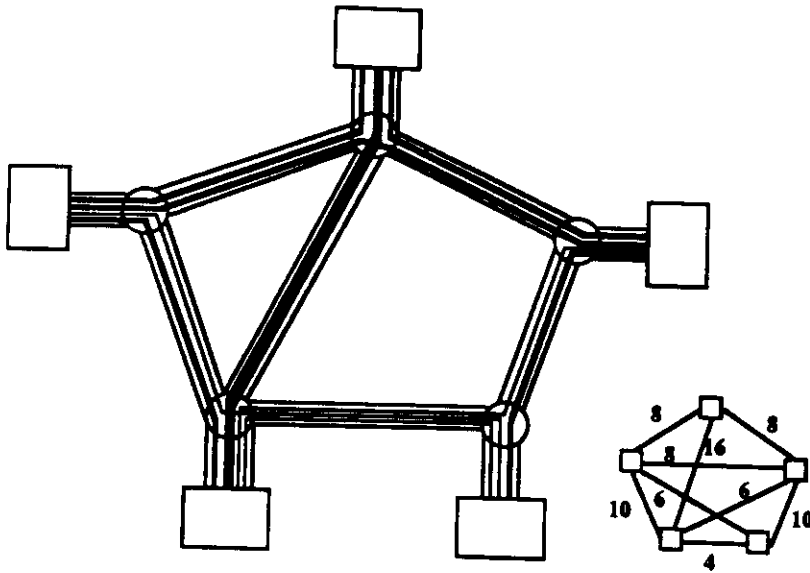


Figure 5.3: Embedded topology B.

topology of Figure 5.3 (which has more express pipes) requires 152 terminations. Using a fully piped network with a direct pipe (of capacity 4, say) also between nodes B and D will further reduce the number of terminations to 144. This is a very important point, since current trends in transmission and processing costs indicate that terminations costs will soon dominate the cost of fiber trunks. Thus, adding more express pipes will imply reducing overall costs.

Another important advantage of express pipes is that of simplifying the congestion control problem. In fact, when a pipe becomes congested, the offending source(s) can be immediately identified and slowed down. In contrast, in a purely meshed packet network, it is often very difficult to trace the sources that cause internal congestion, let alone control them.

There are also drawbacks in the configuration of fully piped (i.e. fully connected) embedded topologies. For example, in networks with a large number of nodes the bandwidth may become too fragmented, and the advantages of statis-

tically multiplexing several sessions on the same trunk may be lost. Thus a good balance must be sought between express pipes and large trunks.

From the network management point of view, the DCS provides added flexibility in that it permits to dynamically tailor the topology to traffic demands. This "topology tuning" is of particular interest in broadband networks implemented using ATM techniques. In fact, ATM nets are stripped of most of the congestion and flow control procedures found in conventional networks, in order to improve switch throughput. If there is a mismatch between offered traffic pattern and network topology, congestion would be inevitable. The problem can be alleviated by dynamically tuning the topology to traffic pattern.

### 5.3.1 Reconfiguration Time Scales

We will distinguish between two types of reconfiguration procedures: *medium term reconfiguration* and *topology tuning*. We envision that medium term reconfigurations may involve a major change in embedded network topology, and may not be transparent to users (i.e., some connections may get interrupted and later resumed). Medium term reconfiguration may take place with a frequency of months or weeks.

Topology tuning, on the other hand, should involve only minor perturbations in the embedded topology, and should be user transparent. It could be carried out on an hourly basis or even more frequently, to overcome short term traffic imbalances and in general render the network more robust to congestion.

The basic methodology that we propose for embedded network design is the same for both medium term configuration and topology tuning. The latter in fact can be viewed as an incremental implementation of the basic algorithm.

In the sequel, we first formulate the static (i.e., medium term) design problem,

define the performance measure and discuss solution approaches. We also discuss the topology tuning problem, illustrating it with a simple example.

### 5.3.2 Topology Tuning

Topology tuning refers to the adjustment of bandwidth allocation and routes in the embedded network in order to overcome congestion caused by an unfavorable traffic pattern or by a failure. This activity is carried out on-line and is closely related to network monitoring and traffic measurement functions. Since topology tuning may be carried out fairly frequently, user transparency is an important consideration.

Ideally, one would like to immediately open the new links and reassign bandwidth from the old to the new links. However, for transparency, a gradual expansion is required. This can be accomplished by opening initially a link of small capacity and adding new calls to this link. As calls are cleared on the old path, bandwidth is reallocated to the new path.

The entire procedure could be automated and in fact run as part of Network Management. It should be clear in fact that in a large network the dynamic adjustment of bandwidth is far too complex to be carried out manually.

## 5.4 Design as a P/S Network

We assume that the backbone facility network has already been defined (i.e., number and location of DCS switches, interoffice fiber trunks, etc.). Generally, this facility will be partitioned among many services, public and private, operational and experimental (e.g., telephony, video distribution, private P/S nets, ISDN, B-ISDN, etc.). In particular, one may envision the presence of several ATM nets (public and private) sharing these facilities.

In our study, we will assume that for the ATM net under consideration a number of interoffice trunk facilities have been reserved a priori. Also, the number and location of the ATM switches has been defined (the ATM switches may or may not be colocated with DCS switches). We refer to the reserved set of interoffice fiber trunks and associated DCS's as the *backbone* topology, as opposed to the *embedded* topology which is derived from the backbone one using DCS switching (see Figures 5.1-5.3 for an example).

The first step in the design of an ATM net is the design of the backbone topology. This will be carried out at network installation. It will take into account current and future traffic demands, availability of fiber facilities, charges for such facilities, fairness in the allocation of fiber resources among various services and user subnets, etc. This design problem is very complex since it must account for many factors, some of which are not known very accurately a priori (e.g., traffic demands are difficult to forecast before network implementation). Thus, this design cannot be expected to be very accurate and should be periodically reviewed. However, corrections to the backbone topology design require reallocation of facilities from one service to another, and, possibly, installation of new fiber facilities, all of which involve substantial lead times. Thus, we may refer to this design phase as *long term planning*.

In the remainder of this Section, we assume that the backbone topology is given, and thus a set of trunks is reserved to a particular ATM net. We concern ourselves with the problem of mapping the embedded topology into the backbone topology. This mapping may need to be revised very frequently, in part to correct the imprecisions inherent in the long term plan, and in part to overcome traffic fluctuations. As pointed out before, the embedded topology design is very different from the conventional packet switching net topology design in that we have a new

set of capacity constraints stating that the sum of the capacities of the embedded links multiplexed on a backbone trunk cannot exceed the capacity of the trunk itself. This new feature makes it necessary to develop new network design tools, as reported here.

#### **5.4.1 Embedded Network Design**

##### **5.4.1.1 Problem Formulation**

Let us consider an embedded network like the one depicted in Figure 5.3. Let us assume that the topology and the trunk capacities of the backbone network are known. Suppose also that an initial embedded topology is defined. This is clearly a major assumption which will be later relaxed. The external traffic offered to the embedded network is also known.

Since the initial embedded topology has been already defined, the key design variables are bandwidth allocation and routing. We can, therefore, generically state the bandwidth allocation and routing problem as follows:

**Given:**

- Topology and capacities of backbone network.
- Topology of embedded network.
- Offered traffic.

**Objective:** Optimize a given performance measure.

**Variables:** Embedded link (pipe) capacities and routing on the embedded network.

**Constraints:**

- Flows should not exceed pipe capacities.
- Backbone trunk capacities are not exceeded by the aggregate capacity of all

the pipes that use the trunk.

- ATM switch capacities are not exceeded.
- Offered traffic is satisfied.

Among the performance measure candidates that we might want to optimize are: average delay, total network throughput, cell loss probability, and call set-up blocking probability.

Even though average delay might not be as critical a design criterion in ATM nets as in conventional networks, delay is still important as an indirect measure of utilization [GF88].

#### 5.4.1.2 Performance Criteria

In conventional packet switched networks, average packet delay is the key performance measure (to be minimized) when the total budget is fixed. In ATM networks, however, the main contribution to packet delay is propagation. For example, for a 53 byte packet size (i.e., 48 data + 5 header, the proposed B-ISDN standard) and for a 150 Mbps channel speeds, the sum of average transmission, queueing and switching delays at an ATM node will be in the order of 10-20  $\mu$ sec, much less than the propagation delay (about 20 msec on a cross country connection). If anything, propagation delays should be minimized; however, propagation delays depend on the geographical distribution of user sites, and are only marginally affected by network topology layout.

More important than delay in ATM nets is the buffer overflow probability at switching nodes. Very strict requirements on packet loss (in the order of  $10^{-6}$  to  $10^{-9}$ ) are often imposed since some applications (e.g., compressed video) are extremely sensitive to packet loss and cannot rely on end to end retransmissions. Packets, however, will occasionally be lost even at relatively low levels of utilization

because of the bursty nature of the traffic and the limited size of nodal buffers. Thus, we would like to configure network topology and routing so as to minimize buffer overflow (i.e., packet loss). Unfortunately, buffer overflow models in typical switching fabrics (buffered and/or unbuffered) under general traffic distributions are very complex and do not yield closed form results.

Given the complexity of the packet loss models on the one hand, and the need for a simple formulation to use in our network optimization model on the other, we propose the following approximation. Arguing that buffer overflow probability is related to average channel queue lengths, and the latter are related to average delay, we will use average delay as the indirect measure of buffer overflow probability (to be minimized). Recalling that each link consists of a bundle of independent 150Mbps channels in parallel, we model a link with  $m$  fixed capacity channels as an  $m$ -M/M/1 queue (this is an approximation since packet length is fixed rather than exponentially distributed). This technique was used before by Ng and Hoang [NH87] and by Lee and Yee [LY89]. The validity of these approximations is supported by experimental evidence (in the design and optimization of conventional packet switched networks) that the optimal routing and topology solution is rather insensitive to the particular shape of the delay vs. link load curve, and is only affected by its asymptotic value, i.e., the link capacity (which is generally the same for all models) [GK77].

The DCSs introduce a delay that is fixed for each traversal. While an express pipe crosses a given DCS just once, an indirect path crosses intermediate DCSs twice. However, since we are just looking for a congestion measure, we can also neglect the DCS delay.

In carrying out this analysis, the usual assumptions for P/S networks are made [Kle72, Kle76]. These assumptions permit us to model ATM nets as a network of



independent M/M/1 queues.

### 5.4.1.3 Solution Approach

As mentioned before, our performance measure is based on link queueing delay where each link consists of an integer number ( $m$ ) of fixed capacity channels. Assuming that the link total flow is uniformly distributed among the  $m$  channels, and analyzing each channel as an independent M/M/1 queue, we obtain the following expression for the average delay  $T$ :

$$T = \frac{1}{R} \sum_{k=1}^{\bar{M}} \frac{m_k \bar{f}_k}{m_k C - \bar{f}_k} \quad (5.1)$$

where  $R$  is the total input rate,  $\bar{M}$  is the number of embedded links (pipes),  $m_k$  is the number of channels at link  $k$ ,  $C$  is the fixed channel capacity, and  $\bar{f}_k$  is the aggregate flow on link  $k$ .

Since the  $m_k$ 's are integral variables while the  $\bar{f}_k$ 's are continuous variables, this is a non-linear mixed integer programming problem. If we relax the integer problem and allow  $m_k$  to assume continuous values, we can show that (5.1) is convex in both  $m$  and  $\bar{f}$  [LY89, NH87]. This property is extremely useful in that it permits us to define an efficient (suboptimal) solution technique and to generate lower bounds.

Our approach consists of a non-linear (continuous) optimization phase followed by a discretization step. Since the problem is convex, we use a steepest descent method [Mar75] in order to obtain the optimal solution to the relaxed problem. Given an initial feasible set of pipe capacities and flows, at each iteration we first linearize the problem with respect to incremental capacities and flows, we then obtain the best set of capacities for the given flow using the simplex method [Mur76], and; the best set of improving flows by solving a shortest-path problem

where the pipe lengths are given by the gradient of  $T$  with respect to the flows. After obtaining these new sets of incremental capacities and flows, we deviate capacities and flows in that direction using a line search method (such as the golden section technique [BS79]). The iterative algorithm terminates (to the global minimum) when the delay improvement falls below a given tolerance. A detailed description of the algorithm is given in Appendix E.

The solution of the relaxed problem provides us with a lower bound to the original problem. To obtain a (suboptimal) solution to the original problem, we round off the continuous capacities to the nearest, feasible integer values, and subsequently optimize the flows using a “flow deviation” iteration [GK77].

#### 5.4.2 Topology Tuning

Our proposed approach is to use an incremental version of the previous algorithm. To illustrate the procedure, assume that there is a sudden surge of traffic between nodes  $i$  and  $j$ , which creates congestion at some of the intermediate switches. Let us further assume that there is no current direct embedded link between  $i$  and  $j$ . Prompted by network congestion alarms, the topology tuning procedure is run at the NCC (Network Control Center). First, the NCC will examine the current topology and traffic to identify the “pressure points” ( $i$  and  $j$ ). Then, the topology tuning algorithm is run using as a starting solution the current traffic flows, and a topology augmented by one or more embedded links between pressure points. The algorithm will iterate and will redistribute bandwidth and traffic from the existing links to the newly introduced link(s), in order to reduce delays and thus eliminate congestion.

### 5.4.3 Example

The algorithm outlined in section 5.4.1.3 was implemented and tested on several representative cases.

In this section we want to illustrate the proposed topology tuning procedure applied to a simple example. Let's consider the backbone network of Figure 5.1. The label attached to each trunk correspond to the number of 150Mbps channels. The DCS nodes are assumed to be able to connect the input channels to any permutation of output channels.

For this example a total of 28 potential pipes and 102 distinct paths that use these pipes were taken into account. For example, to directly connect switches A and C we considered 3 potential pipes described by the following paths: A-F-G-H-C, A-F-J-G-H-C, and A-F-J-I-H-C.

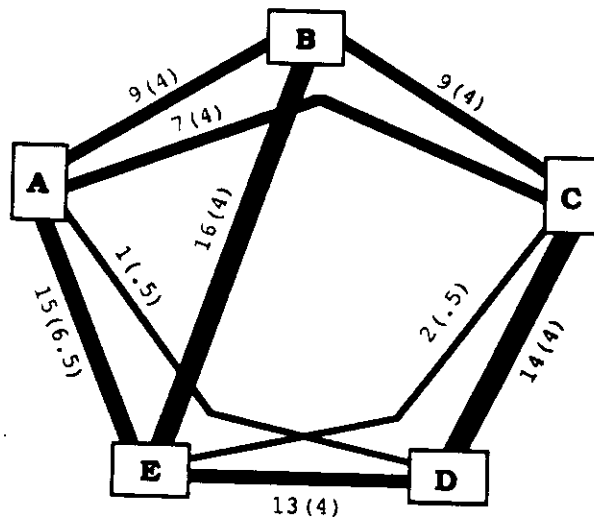
Figure 5.4 shows the best pipe capacities and flows for the traffic pattern of Table 5.2, where the flow is expressed in multiples of the basic channel rate (i.e., 150Mbps). This embedded network has a delay of 11.34  $\mu$ sec that is just 0.3% higher than the lower bound (i.e., the continuous solution). Each pipe and switch is labeled with its capacity and flow (in parentheses), while pipe widths are proportional to their capacities. Note that even though the traffic in the pipes are bidirectional, labels and delays correspond only to one direction.

Now, suppose that the traffic between nodes B and D is increased from 0.0 to 3.0 while the remaining traffic is kept fixed. Simply routing the resulting traffic through the existing pipes would increase the delay to 13.37  $\mu$ sec (or a 17.9% increase).

A complete optimization for the new traffic pattern produced the embedded network of Figure 5.5 that has a 12.24  $\mu$ sec delay, which is 0.6% higher than the

Table 5.2: Original Traffic Matrix

|   | A   | B   | C   | D   | E   |
|---|-----|-----|-----|-----|-----|
| A | -   | 4.0 | 4.0 | 0.5 | 6.5 |
| B | 4.0 | -   | 4.0 | 0.0 | 4.0 |
| C | 4.0 | 4.0 | -   | 4.0 | 0.5 |
| D | 0.5 | 0.0 | 4.0 | -   | 4.0 |
| E | 6.5 | 4.0 | 0.5 | 4.0 | -   |



Delay = 11.34  $\mu$ sec

Figure 5.4: Best network for original traffic.

lower bound. Note that in this solution a direct pipe between B-D was created and all the new B-D traffic was routed through it.

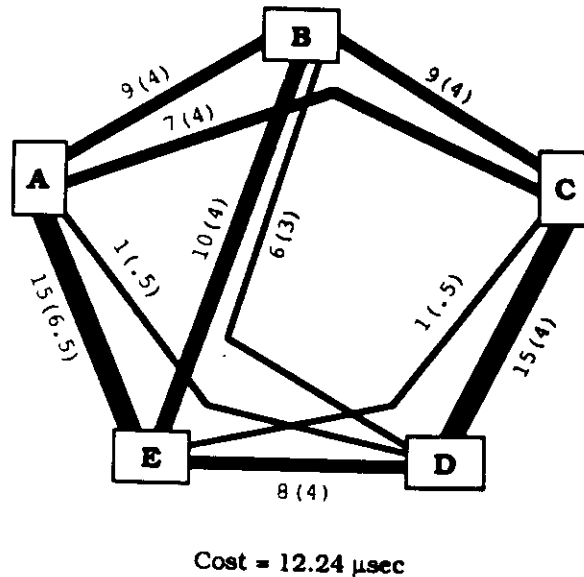
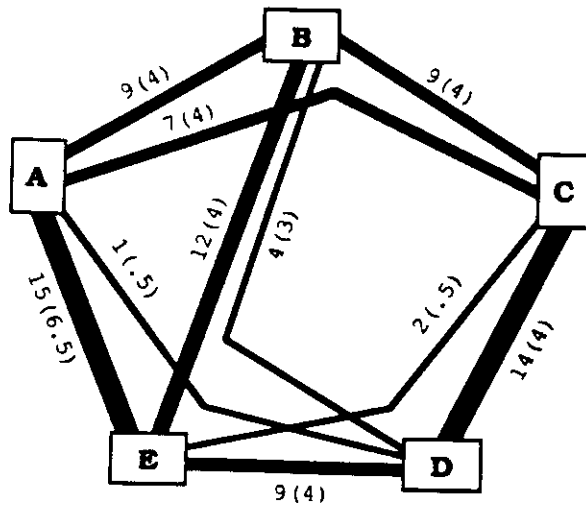


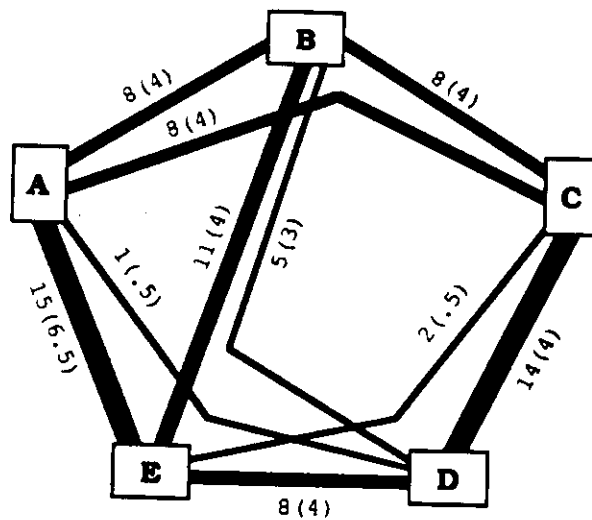
Figure 5.5: Best network after traffic change.

Finally, Figures 5.6–5.7 illustrate the topology tuning procedure presented in section 5.4.2. Since there was no direct connection between nodes B and D, we introduced one, following path B-G-J-I-D (see Figure 5.1) with capacity 4 (Fig. 5.6). Using this configuration as initial solution, a single step of the optimization procedure is applied. Namely, an initial routing for the new traffic is found, in this case through the newly introduced pipe. The resulting configuration has a 13.08  $\mu$ sec delay. After one iteration of the optimization algorithm, followed by a pipe discretization, the network of Figure 5.7 was obtained. This configuration has a delay of 12.47  $\mu$ sec which is just 1.9% higher than the optimal solution of Figure 5.5.



Cost = 13.08  $\mu$ sec

Figure 5.6: Topology Tuning: initial solution.



Cost = 12.47  $\mu$ sec

Figure 5.7: Topology Tuning: Discrete solution.

#### 5.4.4 Summary

In future B-ISDNs, the network manager will have the ability to reconfigure the ATM network topology fairly dynamically and to “tune” it to time varying traffic patterns by using DCS switches and relying on an existing fiber trunk infrastructure. This reconfiguration is particularly important in ATM nets because of the lack of efficient internal flow control mechanisms. Network reconfiguration implies the definition of an embedded topology, the assignment of bandwidth to embedded links (pipes), and the choice of routing. These decisions cannot be carried out manually in a large net; thus, appropriate network design tools are required.

An optimization algorithm that solve this problem can be run off-line (medium term planning); or, it can be run on-line to overcome short term traffic fluctuations (topology tuning). Experimental results show that this technique can lead to substantial delay improvements over non-optimized topologies.

In this Section, we have dealt only with bandwidth allocation and routing. It should be clear, however, that these problems are the first step towards more complex design problems involving the optimal sizing and location of ATM switches.

#### 5.5 Design as a C/S Network

An alternative way to look at the design of ATM networks is by its similarity with C/S networks. In ATM networks slots are not allocated to particular connections or services, as in C/S networks. However, from the analysis of statistical multiplexing of homogeneous and heterogeneous traffic sources (see Chapter 3), we know what is the allowed mix of traffic that still achieves the desired cell loss probability (GOS).

In other words, while in C/S each service type just requires a given bandwidth

which is independent from the other sources that share the same channel, in ATM networks a given source effective bandwidth will depend on the type and number of the other sources that share the same channel <sup>1</sup>.

The combination of admission control and input rate control (policing) mechanisms assure us that the cell loss probability for a given connection, will be below the desired maximum (at least for well behaved ones). Therefore, we can use these results to design a B-ISDN network and route calls such that call blockings are minimized still satisfying the connection GOS.

In this Section, we explore some of the aspects involved in the design of ATM networks as C/S nets. In particular, we will consider the optimal connection routing problem.

### 5.5.1 Assumptions

- All traffic class parameters are given.
  - Cell loss probability requirement.
  - Expansion factor.
  - Call arrival process (e.g., Poisson).
  - Holding time distribution (e.g., exponential).
  - GOS (Call blocking probability).
- Alternate path dynamic routing.
- Reconfiguration capability through DCS.
- Bandwidth (trunk) reservation for each traffic class.

---

<sup>1</sup>If we just use a linear approximation for the traffic mix, the problem is the same as for multiservice C/S. The only difference is that the required bandwidth per source is less in ATM nets because of the multiplexing gain.



### **5.5.1.1 Cell loss probability requirement**

Not all services require the same cell loss probability. For some real time services, delay requirements are more strict than loss requirements. We could design logically distinct networks for traffic classes with distinct cell loss probability requirements and later on join links of such networks based on cost savings, provided that the cell loss probability requirement of the more strict traffic is not violated.

### **5.5.1.2 Expansion factor**

Some of the traffic classes are or will be well known, such as voice, video-phone, and video distribution. However, new traffic classes corresponding to new applications will come up whose behavior the network would not know.

For the former, exact expansion factor (defined in Section 3.3) for a number of traffic sources can be obtained off-line; while for the later, approximations must be used based on the traffic parameters specified at call set-up time.

We envision the existence of standard traffic profiles for typical services, and customized profiles. Customized profiles give all the flexibility to the user, and enough time to the network to analyze its effective bandwidth requirement under homogeneous and heterogeneous traffic mixes.

### **5.5.1.3 Call arrival process**

Poisson call arrival process can be used. It assumes an infinite population what may not be the case for certain services (traffic classes).

### **5.5.1.4 Holding time distribution**

The use of exponential distribution for telephone call holding time was shown to be appropriate, but we can expect that some services will have a more deterministic

holding times, such as video distribution where a connection would usually last for the duration of the program being transmitted.

#### **5.5.1.5 GOS (Call blocking probability)**

Closed form expressions or approximations for call blocking probabilities will depend on the assumed call arrival process and holding time distribution.

#### **5.5.1.6 Alternate path dynamic routing**

In conventional C/S networks, each origin-destination (OD) pair has a set of alternate paths. When a connection is requested for a given OD, the first path in the set is checked. If that path is blocked, the next path in the set will be checked, until either a non-blocked path is found or else a busy signal is returned to the caller.

#### **5.5.1.7 Reconfiguration capability through DCS**

Gopal et al. [GkKW90] showed that for C/S networks, the combination of reconfiguration with a dynamic alternate routing scheme gives the best performance (i.e., lower blocking with traffic pattern changes).

#### **5.5.1.8 Bandwidth (trunk) reservation for each traffic class**

A minimum amount of bandwidth must be reserved for each traffic class in order to guarantee a certain GOS for that class. Without such reservation traffic classes that requires larger amounts of bandwidth would be penalized.

However, to avoid bandwidth waste, the reserved amount should change according to the traffic pattern, and a certain amount of bandwidth could be shared by all traffic classes.

### 5.5.2 Integrated Network Routing Strategy

The Integrated Network Routing (INR) strategy proposed by Ash [ABS88, Ash87] is an extension of the Unified Algorithm (UA) and the Dynamic Nonhierarchical Routing (DNHR) scheme proposed by Ash et al. [ACM81] for uniservice C/S networks.

Bandwidth is allocated to each traffic class at design time according to the traffic load forecast, and at operation time according to the usage. For each connection, there is a fixed maximum allowed utilization (no matter what the traffic mix is).

They assume an integrated network with segregated transport. Therefore, packet switched traffic are routed through virtual trunks (VTs). The utilization on the VTs are limited based on the packet delay constraint imposed on these services. In the other hand, the network controller allocates bandwidth to services in order to minimize blocking for circuit switched services and delay for packet switched services.

The INR net design has as a goal to minimize global cost while providing an objective Origin-Destination (OD) blocking and service quality levels between all OD pairs in all load set periods (LSP) [Ash87].

The solution is obtained by iteratively solving for the routing, capacity and link blocking variables. Blocking in a path is computed using the Erlang-B formula. Optimal blockings for direct and alternate paths are obtained using Truitt method [Tru54]. Routing is formulated as a Linear Programming but solved using an heuristic method.

### 5.5.3 Application of INR to ATM Networks

In ATM networks we have an integrated transport environment where all kind of services including typically circuit switched traffic may share transmission facilities.

As discussed in Chapter 3, in ATM networks we are more concerned about cell loss probabilities than packet delay (even though it is still present as a constraint). Therefore, the allowed utilization (or expansion factor) at each link is a function of the desired cell loss probability and the forecasted mix of traffic.

We propose two improvements in the INR method in order to be used for ATM networks: The use of a variable bandwidth requirement per call (Expansion factor); and Reconfiguration.

#### 5.5.3.1 Variable bandwidth requirement per call

In the original INR strategy, the maximum number of VTs for service type  $i$  on link  $k$  on time  $h$ ,  $VT_{ik}^h$  is given by [ABS88, Equation 1]:

$$VT_{ik}^h = \rho_i \times \frac{LBW_{ik}^h}{r_i} \quad (5.2)$$

where  $LBW_{ik}^h$  is the bandwidth allocated to service type  $i$  on link  $k$  on hour  $h$ ,  $\rho_i$  is the maximum allowed utilization of such bandwidth, and  $r_i$  is the average bandwidth per VC for service type  $i$ .

As we can see, for a given service  $i$  with average bandwidth  $r_i$ , there is a linear relation between allocated bandwidth  $LBW_{ik}^h$  and the number of VTs. In our case, however, due to the statistical gain from multiplexing, a linear relation is too pessimistic. Let us call  $\omega_i$  the function which gives the required bandwidth from the number of sources, i.e.  $LBW_i = \omega_i(VT_i)$ . The function  $\omega_i$  can be obtained by

the use of the UAS model. Therefore,

$$VT_{ik}^h = \omega_i^{-1}(LBW_{ik}^h). \quad (5.3)$$

The above equation does not take into account the statistical gain obtained by multiplexing the different traffic types that share the same channel. In the optimistic case of linear mixing, the number of virtual trunks is given by

$$VT_{ik}^h = LBW_{ik}^h \times \frac{N_{max}^{ik}}{W_k} \quad (5.4)$$

where  $W_k$  is the capacity of link  $k$ , and  $N_{max}^{ik}$  is the maximum number of sources of type  $i$  that can be multiplexed in link  $k$  still guaranteeing the desired cell loss probability.

Note that Equation 5.4 is similar to Equation 5.2. A more complicated expression or algorithm would have to be derived if a non-linear approximation is used instead.

### 5.5.3.2 Reconfiguration

In terms of reconfiguration, an initial network can be obtained by designing according to the estimated traffic. In order to obtain a new topology we can make use of the Unified Algorithm (UA) keeping capacities fixed and minimizing average call blocking probabilities.

### 5.5.4 Multiservice C/S Network Design

Pióro et al. [PLK89] have a different approach. A distinct network is designed for each service type. At the end, partially heuristic methods are used to superpose these networks.

Blocking on an isolated link is obtained for several *link access control* (LAC) strategies: complete sharing, class limitation, trunk reservation, and positive revenue. For each strategy a Poissonian call arrival distribution is assumed. Links are optimized based on a two parameter version of the classical Truitt method. LAC mechanisms are introduced in the last, superposition step of the design.

## 5.6 Summary and Future Work

In this Chapter we considered the problem of designing a B-ISDN network built using ATM technology. Initially, we identified the correspondent design problems, and in particular, we introduced the *topology tuning* problem.

These problems are studied as P/S network design problems where the link (excluding propagation) and node delays are used as an indirect measure of congestion. The problem of bandwidth allocation and routing of “express-pipes” on top of a facility network is formulated and a solution approach is sketched. The Section concludes with an example of the use of such algorithm for the topology tuning of a simple network.

There is opportunity to extend this work in many directions. First, a more accurate model for buffer overflow must be developed, based on a more realistic model of user traffic, such as bursty or VBR traffic. Regarding the “topology tuning” version of the algorithm, more work is required in the development of the interface between the network monitoring facility and the network optimization program (e.g., how to measure traffic requirements; how to measure node/link congestion, etc). Also, more work is needed on the transparent implementation of the dynamic reconfiguration (i.e., bandwidth allocation and routing) in an operational network. Furthermore, if several ATM nets (public and/or private) are embedded in the same fiber facility, the issue of how to fairly partition (and, perhaps, dynamically

share) the common facility among subnets must be investigated.

Departing from the fact that with the use of admission and input rate control mechanisms, the effective bandwidth of each connection is known, we characterized the B-ISDN network problem as a (multiservice) C/S design problem. An Integrated Network Routing (INR) strategy proposed elsewhere was presented, followed by the proposal of some improvements. An alternative approach was also briefly described.

As future research we suggest the implementation of the proposed methods, possibly the development of new ones, and their comparison.





## Chapter 6

### Conclusions

In this concluding Chapter, we summarize the work done and its major contributions (Section 6.1), as well as suggestions for further work (Section 6.2).

#### 6.1 Contributions

Our first contribution was the *bandwidth allocation* study using the Uniform Arrival and Service (UAS) analytical model for homogeneous (all traffic sources are of the same type) bursty and variable bit rate (VBR) traffic. We obtained the effective required bandwidth per source in a finite buffer multiplexer in order to achieve a given Grade-Of-Service (GOS), expressed by the cell loss probability. For both bursty and VBR traffic sources, we performed a sensitivity analysis with significant parameters. The required bandwidth for bursty sources was shown to depend on burst and buffer length only through their ratio.

The mix of continuous bit rate (CBR), bursty, and VBR traffic was later considered. We compared the results obtained through simulation with approximations proposed in the literature. The linear approximation was shown to be usually too optimistic, while the class related rule (CRR) was usually too conservative. A non-linear approximation proposed elsewhere was shown to give the best match with simulation results.

These results were encouraging, because they demonstrated that analytical models to obtain the effective required bandwidth, combined with mixing rules

can be used by an admission control strategy, in order to decide whether or not a new call can be accepted based on its effective required bandwidth and effect on the mix of traffic.

Still on the statistical multiplexing arena, we studied the multiplexing of bursty sources in the network internal buffers (i.e., buffers in tandem) through simulation. The question we posed ourselves was: *Is the effective bandwidth requirement obtained from a first stage buffer, valid also for internal ones?* Obviously, if all the traffic followed a single path, without new traffic joining at intermediate nodes, buffer would be required only at the first stage (entry point). We have shown that an admission control based on the bandwidth required at the first stage multiplexer is conservative in the measure by which the aggregate traffic follows a single path. In other words, the more the traffic splits inside the network, the more its burstiness is preserved.

Evidently, the desired cell loss probability will be achieved only if the sources abide to the parameters specified at call set-up. Therefore, the network must implement some special mechanisms, called *Input Rate Control* to assure that the sources abide by their specification.

We studied this policing function on its side of the user network interface in order to protect well behaved sources from dishonest ones. The most popular input rate control for broadband networks in the literature, leaky bucket, was shown not to be adequate for deletion of excessive traffic cells. A marking scheme proposed elsewhere was adopted. In this scheme, excessive traffic cells are not deleted but simply marked. In case of congestion, marked (low priority) cells have to defer to unmarked (high priority) ones. The performance of jumping window, leaky bucket, and three new mechanisms (delta, delta-2, and delta-3) were studied and compared, both for a bursty source and for a variable bit rate (VBR) traffic. The comparison

was based on four factors: conformity to the ideal marking probabilities, effect on well-behaved sources, reaction time, and implementation complexity.

We concluded that the delta-2 mechanism, even though not conforming to the ideal marking probability behavior, nor being the simplest one, has the nice property of penalizing excessive traffic sources, by marking more than the excessive cells for large offending factors. As a result, the cell loss probability of well-behaved sources may even be reduced below the desired levels.

The design of B-ISDNs differs from the design of conventional data networks mainly in the traffic statistics, nodal bottlenecks, and in the possibility of dynamically reconfiguring the logical network topology through the use of Digital Cross-connect Systems (DCS). Initially, we identified the correspondent design problems, and in particular, we introduced the *topology tuning* problem. In other words, the logical topology can be dynamically adjusted in order to reduce the number of intermediate hops along the path and thus reduce store and forward delay and nodal processing overhead, or simply to allocate more bandwidth to a congested path. It also simplifies the congestion control problem since, when a logical link becomes congested, the sources can be immediately stopped. We presented an algorithm for the topology tuning of B-ISDN using a P/S design approach, and applied it to a simple example.

Finally, the design of B-ISDNs can be also viewed as the design of a multiservice C/S network where the required bandwidth per call is not fixed but depends on the traffic mix. We suggested some improvements on one of the design strategies presented in the literature in order to account for this variability with the traffic mix.

## 6.2 Future work

In this Section, we identify topics that deserve further study, and some suggestions on how to approach them.

In this research we considered three traffic models, namely, periodic, bursty (with exponentially distributed burst and silence periods), and a video-phone type variable bit rate traffic. Further work is necessary in the characterization and modeling of new services traffic. Other analytical models (e.g., Markovian Modulated Poisson Process - MMPP) can be considered for the modeling of new services and the mix of different traffic sources. A particularly interesting case is the mix of well-behaved and offending cells, for the study of input rate control mechanisms. This is a particularly difficult problem because of the priority handling at the buffers; and the need to distinguish cell loss for each priority class (well-behaved and offending ones).

We have studied the IRC mechanisms behavior under steady offending traffic sources. Even though we derived some bounds for reaction times, it would be of interest to study the behavior of such mechanisms under transient (sporadic surges of) offending cells.

Most of our simulations were performed at the cell level, and consequently, required long simulation times even for moderate cell loss probabilities ( $10^{-4}$  to  $10^{-5}$ ). Simulation results are always useful, either because no analytical solution is known, or just to validate analytical methods and approximations. Therefore, we suggest the use of some (new) techniques to reduce simulation times, such as the Extreme Value Theory [Ber90]; and to simulate as much as possible at higher levels (e.g., burst or frame levels) rather than at the cell level.

Regarding the network design problem, there are opportunities to extend this

work in many directions. First, a more accurate model for buffer overflow must be developed, based on a more realistic model of user traffic, such as bursty or VBR traffic<sup>1</sup>. Regarding the "topology tuning" version of the algorithm, more work is required in the development of the interface between the network monitoring facility and the network optimization program (e.g., how to measure traffic requirements; how to measure node/link congestion, etc). Also, more work is needed on the transparent implementation of the dynamic reconfiguration (i.e., bandwidth allocation and routing) in an operational network. Furthermore, if several ATM nets (public and/or private) are embedded in the same fiber facility, the issue of how to fairly partition (and, perhaps, dynamically share) the common facility among subnets must be investigated.

In the characterization of the B-ISDN network problem as a (multiservice) C/S design problem, as future research we suggest the implementation of the proposed methods, possibly the development of new ones, and their comparison.

---

<sup>1</sup>We assumed Poisson arrivals, and exponential service times



## APPENDIX A

### Uniform Arrival and Service Model

The uniform arrival and service (UAS) model [AMS82] is a fluid-flow approximation of the state of an infinite buffer multiplexer where no packetization is considered. The multiplexer is fed by a finite number ( $N$ ) of independent and identical bursty sources which alternate between active and silence periods, both exponentially distributed. During its active period a source transmits at a uniform rate. Bits flow into the multiplexer at a rate which depends on the number of active sources, and bits flow out of the multiplexer at a fixed service rate.

The average silence period is denoted by  $1/\lambda$ , while the average active period is denoted by  $1/\mu$ . Without loss of generality, the unit of information is chosen to be the amount generated by a source in an average active period. The output multiplexer rate relative to a source's transmission rate is denoted by  $c$ .

Therefore, if  $i$  sources are simultaneously active, the buffer length will increase at rate  $(i - c)$  as long as  $i > c$ ; it will remain fixed as long as  $i = c$ ; if not empty, it will decrease at rate  $(c - i)$ , as long as  $i < c$ ; and if empty, it will remain so as long as  $i \leq c$ . We assume that the following stability condition is satisfied:

$$\rho = \frac{N\lambda}{c(\mu + \lambda)} < 1. \quad (\text{A.1})$$

The set of differential equations that governs the equilibrium buffer distribution is derived, and is given by

$$(i - c) \frac{dF_i}{dx} = (N - i + 1)\lambda F_{i-1} - \{(N - i)\lambda + i\mu\}F_i + (i + 1)\mu F_{i+1}, \quad (\text{A.2})$$





where,

$$\begin{aligned} A(k) &\stackrel{\text{def}}{=} (N/2 - k)^2 - (N/2 - c)^2 \\ B(k) &\stackrel{\text{def}}{=} 2(\mu - \lambda)(N/2 - k)^2 - N(\mu + \lambda)(N/2 - c) \\ C(k) &\stackrel{\text{def}}{=} -(\mu + \lambda)^2 \{(N/2)^2 - (N/2 - k)^2\} \end{aligned}$$

We are interested only in the stable (negative) eigenvalues. There are  $N - [c]$  of them.

Let  $\Phi(x)$  denote the generating function of  $\phi$ . For each stable eigenvalue,  $\Phi(x)$  is then obtained from

$$\Phi(x) = (x - r_1)^{c_1} (x - r_2)^{N - c_1}, \quad (\text{A.5})$$

where,

$$\begin{aligned} r_1 &= \{-(z + \mu - \lambda) + \sqrt{(z + \mu - \lambda)^2 + 4\lambda\mu}\}/2\lambda \\ r_2 &= \{-(z + \mu - \lambda) - \sqrt{(z + \mu - \lambda)^2 + 4\lambda\mu}\}/2\lambda \end{aligned}$$

and,

$$c_1 = \frac{zc - N\lambda + N\lambda r_1}{\lambda(r_1 - r_2)}.$$

Using the boundary conditions Equation A.4 becomes,

$$\mathbf{F}(x) = \mathbf{F}(\infty) + \sum_{i=0}^{N-[c]-1} e^{zix} a_i \phi_i, \quad (\text{A.6})$$

where,

$$a_j = - \left( \frac{\lambda}{\mu + \lambda} \right)^N \prod_{i=0; i \neq j}^{N-[c]-1} \frac{z_i}{z_i - z_j}, \quad 0 \leq j \leq N - [c] - 1.$$

For further details please refer to [AMS82].



## APPENDIX B

### Uniform Arrival and Service Model for Finite Buffer Size

The uniform arrival and service model for finite buffer size is the same fluid-flow approximation presented in Appendix A, except that the buffer is no longer infinite. The finite buffer changes the boundary conditions and the eigenvalues and eigenvectors are no longer obtained in closed form, but have to be computed by numerical methods.

In the talker activity model, the number of active lines (sources)  $i$  is modeled by a continuous-time birth-death process [Wei78], where the transition rates are given by

$$p(i, i + 1) = (N - i)\lambda \quad 0 \leq i < N \quad (\text{B.1})$$

$$p(i, i - 1) = i\mu \quad 0 < i \leq N \quad (\text{B.2})$$

where  $1/\mu$  is the mean talkspurt length,  $1/\lambda$  is the mean silence length, and  $N$  is the number of input lines. In our notation,  $\mu = \frac{B_p}{Ln_{call}}$ .

The equilibrium probability of  $i$  lines being active,  $b_i$ , is given by the binomial distribution

$$b_i = \frac{\left(\frac{\lambda}{\mu}\right)^i \binom{N}{i}}{\left(1 + \lambda/\mu\right)^N}. \quad (\text{B.3})$$

The equilibrium probability that  $i$  lines are active and the queue length does not exceed  $x$ , represented by  $F_i(x)$ , is obtained by solving the matricial equation [Tuc88, Equation 5]

$$\mathbf{D} \frac{d}{dx} \mathbf{F}(x) = \mathbf{M} \mathbf{F}(x) \quad 0 < x < m \quad (\text{B.4})$$

where  $\mathbf{D}$  is a diagonal matrix whose elements contains the increasing or decreasing queue length rates,  $\mathbf{M}$  is the talker activity transition rate matrix, and  $m$  is the maximum buffer size (i.e.,  $m = Kn_{cell}/B_p$ ).

The loss probability is given by [Tuc88, Equation 12]

$$\text{Loss} = (1/\alpha N) \sum_{i=C}^N (i - c) u_i \quad (\text{B.5})$$

where, in our notation,  $\alpha = 1/b$ ,  $c = W/B_p$ ,  $C = [c]$ , and  $u_i = b_i - F_i(m-)$ .

The solution to Equation B.4 is of the form [Tuc88, Equation 6]

$$F(x) = \sum_{k=0}^N e^{z_k x} a_k \Phi_k \quad 0 < x < m$$

where  $z_k$  is an eigenvalue of  $\mathbf{D}^{-1}\mathbf{M}$ ,  $\Phi_k$  is its correspondent right eigenvector, and  $a_k$  are coefficients obtained by solving suitable boundary equations.

Finally, the eigenvalues and eigenvectors are obtained by the use of numerical methods [PW71], while the coefficients  $a_k$ 's are obtained by solving the following set of boundary equations:

$$F_i(0) = \sum_{k=0}^N a_k \{\phi_k\}_i = 0 \quad c < i \leq N$$

$$F_i(m-) = b_i \quad 0 \leq i < c.$$

## APPENDIX C

### Invariance of R with same K/L ratio

The cell loss probability, and therefore also the expansion factor, of bursty sources on a multiplexer can be shown to be independent of the values of the buffer length ( $K$ ) and average burst length ( $L$ ), as long as their ratio is kept constant. This property has been previously observed by Li [Li89]. The proof, however, is novel.

This proposition made in section 3.3.2 can be formally proven as follows:

In the talker activity model, the number of active lines (sources)  $i$  is modeled by a continuous-time birth-death process [Wei78], where the transition rates are given by Equations B.1-B.2.

Since  $\lambda/\mu = 1/(b-1)$ , the equilibrium probability of  $i$  lines being active,  $b_i$ , given by Equation B.3, does not change with  $L$  as long as the burstiness is kept constant. What changes with  $L$  are the transition rates, which affects the amount of generated traffic per active source.

The loss probability is given by Equation B.5 where,  $u_i = b_i - F_i(m-)$ . We have already shown that  $b_i$  does not change with  $L$ , now we have to show that  $F_i(m-)$  does not change if the ratio  $L/K$  is kept fixed.

$F_i(m-)$  is given by

$$F_i(m-) = \sum_{k=0}^N e^{z_k m} a_k \{\phi_k\}_i.$$

If we multiply  $L$ , and therefore  $1/\lambda$  and  $1/\mu$ , by a factor  $\delta$ , this has the effect

of dividing all elements of the transition matrix  $\mathbf{M}$  by  $\delta$ , i.e.,

$$L' = \delta L \Rightarrow \mathbf{M}' = \frac{1}{\delta} \mathbf{M}.$$

Let  $z'_k$  be an eigenvalue of  $\mathbf{D}^{-1} \mathbf{M}'$  and  $\Phi'_k$  be its corresponding right eigenvector.

It is easy to show that  $z'_k = z_k/\delta$  and  $\Phi'_k = \Phi_k$ .

Now, if  $K$  and therefore  $m$  is also multiplied by the same factor  $\delta$ , we have that

$$\begin{aligned} e^{z'_k m'-} &= e^{\frac{z_k}{\delta} \delta m-} \\ &= e^{z_k m-}. \end{aligned}$$

Furthermore, since the  $a_k$ 's are obtained by solving the following set of boundary equations

$$\begin{aligned} F_i(0) &= \sum_{k=0}^N a_k \{\phi_k\}_i = 0 \quad c < i \leq N \\ F_i(m-) &= b_i \quad 0 \leq i < c \end{aligned}$$

which do not change if  $L$  and  $K$  are multiplied by the same factor  $\delta$ , we conclude that the loss probability is also kept unchanged.

## APPENDIX D

### Variable Bit Rate Traffic Fluid Flow Model

Maglaris et al. [MAS<sup>+</sup>88] used the UAS model of Appendix A to model the state of a statistical multiplexer to which is offered a finite number of independent variable bit rate (VBR) sources.

The bit rate is quantized into finite discrete levels at random Poisson time instances. Transitions between levels are assumed to occur with exponential transition rates. The quantization step, the number of states, and the transition rates are tuned to fit the average variance and autocovariance function of a videophone call measured data.

Let us denote by  $M$  the maximum number of quantization levels that the aggregate traffic may reach. Maglaris et al. found that the use of  $M = 20 \times N$ , where  $N$  is the number of identical sources, exhibits a good match with simulation results. Furthermore, we can think about  $M$  as the number of independent bursty “mini-sources” (i.e., each of them alternates between active and silence periods).

With this interpretation, Maglaris et al. [MAS<sup>+</sup>88] used Anick et al. approach [AMS82] (see Appendix A), to obtain the multiplexer buffer statistics.

We can use the results reported in Appendix A as long as we substitute the number of sources ( $N$ ) by the number of mini-sources ( $M$ ), the relative output multiplexer rate  $c$  by  $c/A$ , and the queue length  $x$  by  $x \cdot c/A$ , throughout.

For the measured data, the UAS parameters must be set as follows:

$$\mu = 3.9 / \left( 1 + \frac{5.04458N}{M} \right)$$

$$\lambda = 3.9 - \mu$$

$$A = 0.1 + 0.52 \frac{N}{M}$$



## APPENDIX E

### Bandwidth Allocation and Routing in the Embedded Network

Representing the path  $\pi_u$  that pipe  $u$  follows in the backbone network by the vector  $\mathbf{p}_u$  (where  $p_{um} = 1$  if arc  $m \in \pi_u$ , and  $p_{um} = 0$ , otherwise); and by  $\bar{\mathbf{p}}(k; n)$  the  $n$ -th path that commodity  $k$  follows in the embedded network, the bandwidth allocation and routing problem can be stated as follows:

**Given:** Topology and capacities of backbone network.

Topology of embedded network.

Offered traffic.

**Objective:** Minimize average packet delay; i.e.,

$$\min z = \frac{1}{RC} \sum_{u=1}^{\bar{M}} \frac{\bar{C}_u \bar{f}_u}{\bar{C}_u - \bar{f}_u} \quad (\text{E.1})$$

**Variables:** Pipe capacities and routing on the embedded network.

**Constraints:**

$$\sum_{u=1}^{\bar{M}} \bar{C}_u \mathbf{p}_u \leq \mathbf{C} \quad (\text{E.2})$$

$$\sum_{n=1}^{N_k} \bar{x}(k; n) = r_k \quad \forall k \quad (\text{E.3})$$

$$\sum_{k=1}^Q \sum_{n=1}^{N_k} \bar{x}(k; n) \bar{\mathbf{p}}(k; n) = \bar{\mathbf{f}} \quad (\text{E.4})$$

$$\bar{\mathbf{f}} \leq \bar{\mathbf{C}} \quad (\text{E.5})$$

$$\bar{\mathbf{C}} \geq \mathbf{0}, \bar{\mathbf{f}} \geq \mathbf{0}, \bar{\mathbf{x}} \geq \mathbf{0} \quad (\text{E.6})$$

where  $\mathbf{C} = (C_1, C_2, \dots, C_M)^T$  is the vector of backbone trunk capacities,  $\bar{\mathbf{C}} = (\bar{C}_1, \bar{C}_2, \dots, \bar{C}_M)^T$  is the vector of pipe capacities,  $\bar{C}_u = m_u C$ ,  $\bar{\mathbf{f}} = (\bar{f}_1, \bar{f}_2, \dots, \bar{f}_M)^T$  is the vector of pipe flows, and  $\bar{\mathbf{x}} = (\bar{x}(1; 1), \bar{x}(1; 2), \dots, \bar{x}(Q; N_Q))^T$  is the vector of flows in the paths defined over the pipes.

Constraint (E.2) expresses the condition that the capacity of each backbone trunk must not be exceeded by the aggregate capacity of all the pipes that use the backbone trunk. Equation (E.3) expresses the condition that the flow of each commodity  $k$  carried on all the paths  $\bar{\pi}(k; 1), \bar{\pi}(k; 2), \dots, \bar{\pi}(k; N_k)$  must be equal to the flow  $r_k$  of commodity  $k$  offered to its source.

Equation (E.4) states that the aggregate flow on each pipe must be equal to the sum of all the commodity flows on all the commodity paths that use the pipe. Constraint (E.5) states that the aggregate flow on each pipe must not exceed its capacity.

Constraint (E.5) is included in the objective function as a penalty function, and therefore, it can be removed from the formulation.

Since the problem is convex we use Frank-Wolfe's steepest descent method [Mar75] for finding the global minimum. In this method, given a feasible solution  $(\bar{\mathbf{C}}^{K-1}, \bar{\mathbf{f}}^{K-1})$ , we can find a feasible direction of descent by solving the following problem:

$$\begin{aligned} \min \quad & \nabla z(\bar{\mathbf{C}}^{K-1}, \bar{\mathbf{f}}^{K-1}) \cdot (\bar{\mathbf{C}}, \bar{\mathbf{f}}) \\ \text{subject to} \quad & \text{(E.2)- (E.4) and (E.6)} \end{aligned} \tag{E.7}$$

Observe that this problem can be separated into the following two problems:

$$\min \quad \nabla_{\bar{\mathbf{C}}} z(\bar{\mathbf{C}}^{K-1}, \bar{\mathbf{f}}^{K-1}) \cdot \bar{\mathbf{C}} \tag{E.8}$$

$$s. t. \sum_{u=1}^{\bar{M}} \bar{C}_u \mathbf{p}_u \leq \mathbf{C} \quad (\text{E.9})$$

$$\bar{\mathbf{C}} \geq \mathbf{0} \quad (\text{E.10})$$

and

$$\min y = \nabla_{\bar{\mathbf{f}}} z(\bar{\mathbf{C}}^{K-1}, \bar{\mathbf{f}}^{K-1}) \cdot \bar{\mathbf{f}} \quad (\text{E.11})$$

$$s. t. \sum_{n=1}^{N_k} \bar{x}(k; n) = r_k \quad \forall k \quad (\text{E.12})$$

$$\sum_{k=1}^Q \sum_{n=1}^{N_k} \bar{x}(k; n) \bar{\mathbf{p}}(k; n) = \bar{\mathbf{f}} \quad (\text{E.13})$$

$$\bar{\mathbf{f}} \geq \mathbf{0}, \bar{\mathbf{x}} \geq \mathbf{0} \quad (\text{E.14})$$

If  $\bar{\mathbf{C}}^{\#}$  minimizes problem (E.8)-(E.10) and  $\bar{\mathbf{f}}^{\#}$  minimizes (E.11)-(E.14), then  $(\bar{\mathbf{C}}^{\#}, \bar{\mathbf{f}}^{\#}) - (\bar{\mathbf{C}}^{K-1}, \bar{\mathbf{f}}^{K-1})$  is a feasible direction of descent.

**ALGORITHM 1** (*Algorithm for bandwidth allocation and routing in the embedded network.*)

**Input:** Topology and backbone trunk capacities, commodities arc-path vectors, traffic requirements, initial feasible solution  $(\bar{\mathbf{C}}^0, \bar{\mathbf{f}}^0)$ , and tolerance  $t$ .

**Objective function:** Equation (E.1).

**STEP 0:** Set  $K = 0$  and  $z^{\text{old}} = \infty$ .

**STEP 1:** Set  $K = K + 1$ . Find the vector  $\mathbf{C}^{\#}$  that minimizes (E.8)-(E.10); (This problem can be solved using the revised simplex method [Mur76].) And find the vector  $\bar{\mathbf{f}}^{\#}$  that minimizes (E.11)-(E.14). (This problem can be solved by constructing a minimum path solution  $(\bar{\mathbf{f}}^{\#}, \bar{\mathbf{x}}^{\#})$  for the embedded network, where the cost of each pipe  $\bar{a}_u$  is  $\bar{l}_u = \partial z(\bar{\mathbf{C}}^{K-1}, \bar{\mathbf{f}}^{K-1}) / \partial \bar{f}_u$ .)

**STEP 2:** Find the value  $\alpha^*$  that minimizes  $z[\alpha(\bar{C}^*, \bar{f}^*) + (1 - \alpha)(\bar{C}^{K-1}, \bar{f}^{K-1})]$ .

This optimum may be obtained by any convenient line search method (such as the golden section technique [BS79]).

**STEP 3:** Set  $(\bar{C}^K, \bar{f}^K) = \alpha^*(\bar{C}^*, \bar{f}^*) + (1 - \alpha^*)(\bar{C}^{K-1}, \bar{f}^{K-1})$ .

If  $z(\bar{C}^{K-1}, \bar{f}^{K-1}) - z(\bar{C}^K, \bar{f}^K) \leq t$ , go to 4, a potential minimum has been found; otherwise go to 1.

**STEP 4:** If  $z^{\text{old}} - z(\bar{C}^K, \bar{f}^K) \geq t$ , go to 5. Otherwise, set  $(\bar{C}^*, \bar{f}^*) = (\bar{C}^{\text{old}}, \bar{f}^{\text{old}})$  if  $z^{\text{old}} \leq z(\bar{C}^K, \bar{f}^K)$ , set  $(\bar{C}^*, \bar{f}^*) = (\bar{C}^K, \bar{f}^K)$  if  $z^{\text{old}} > z(\bar{C}^K, \bar{f}^K)$ , and stop, the global minimum has been obtained.

**STEP 5:** Set  $(\bar{C}^{\text{old}}, \bar{f}^{\text{old}}) = (\bar{C}^K, \bar{f}^K)$  and  $z^{\text{old}} = z(\bar{C}^{\text{old}}, \bar{f}^{\text{old}})$ . Find the vector  $\bar{C}^a$  that solves

$$\begin{aligned} \min z &= \frac{1}{R} \sum_{u=1}^M \frac{\bar{C}_u \bar{f}_u^{K-1}}{\bar{C}_u - \bar{f}_u^{K-1}} \\ \text{s. t. } &\sum_{u=1}^M \bar{C}_u \mathbf{p}_u \leq \mathbf{C} \\ &\bar{\mathbf{C}} \leq \bar{\mathbf{f}}^{K-1} \end{aligned}$$

$\bar{C}^a$  can be found using Algorithm 1, described in [PRG82], with  $\bar{C}^{K-1}$  as initial feasible solution). Set  $(\bar{C}^K, \bar{f}^K) = (\bar{C}^a, \bar{f}^{K-1})$  and go to 1.

**Output:** The global minimum is  $(\bar{C}^*, \bar{f}^*, \bar{x}^*)$ , where  $\bar{x}^*$  is such that  $(\bar{f}^*, \bar{x}^*)$  satisfies (E.3), (E.4) and (E.6).

## Bibliography

- [ABS88] Gerald R. Ash, Bruce M. Blake, and Steven D. Schwartz. Integrated network routing and design. In *Proc. 12th International Teletraffic Congress*, pages 1.4A.3.1-8, Torino, Italy, June 1988.
- [ACM81] G. R. Ash, R. H. Cardwell, and R. P. Murray. Design and optimization of networks with dynamic routing. *The Bell System Technical Journal*, 60(8):1787-1820, October 1981.
- [AD89] Hamid Ahmadi and Wolfgang E. Denzel. A survey of modern high-performance switching techniques. *IEEE Journal on Selected Areas in Communications*, 7(7):1091-1103, September 1989.
- [Akh87] Shahid Akhtar. Congestion control in a fast packet switching network. Master's thesis, Washington University, December 1987.
- [Ami88] Hamid R. Amirazizi. Controlling synchronous networks with digital cross-connect systems. In *Proc. GLOBECOM '88*, pages 1560-1563, Hollywood, FL, November 1988. IEEE.
- [AMS82] D. Anick, D. Mitra, and M. M. Sondhi. Stochastic theory of a data-handling system with multiple sources. *The Bell System Technical Journal*, 61(8):1871-1894, October 1982.
- [Ash87] Gerald R. Ash. Traffic network routing, control, and design for the ISDN era. In *Proc. 5th ITC Seminar*, Lake Como, Italy, May 1987.
- [AW88] R. G. Addie and R. E. Warfield. Bandwidth switching and new network architectures. In *Proc. 12th International Teletraffic Congress*, pages 2.3iiA.1-7, Torino, Italy, June 1988.
- [Bat68] K. E. Batcher. Sorting networks and their applications. In *Proc. Spring Joint Computer Conference*, pages 307-314, Atlantic City, NJ, April 1968. AFIPS.
- [BCS90] Krishna Bala, Israel Cidon, and Khosrow Sohraby. Congestion control for high speed packet switched networks. In *Proc. INFOCOM '90*, pages 520-526, San Francisco, CA, June 1990. IEEE.

- [Ber90] Ignacio Berberana. Application of extreme value theory to the analysis of a network simulation. In *Proc. 23rd Annual Simulation Symposium*, pages 105–121, Nashville, TN, April 1990. IEEE.
- [BFIL87] Francesco Bernabei, Alessandro Forcina, Eugenio Iannone, and Marco Listanti. Analisi delle prestazioni di reti Delta (in Italian). *Note Recensioni e Notizie*, pages 127–137, 1987.
- [BFL88] F. Bernabei, A. Forcina, and M. Listanti. On non-blocking properties of parallel delta networks. In *Proc. INFOCOM '88*, pages 326–333, New Orleans, Louisiana, March 1988. IEEE.
- [BILV85] F. Bernabei, E. Iannone, M. Listanti, and F. Villani. Generalization of Jenq model for performance analysis of banyan interconnection networks. Technical report, Fondazione Ugo Bordoni, Roma, Italy, 1985.
- [BS79] M. S. Bazaraa and C. M. Shetty. *Nonlinear Programming Theory and Algorithms*, pages 257–259. John Wiley and Sons, New York, 1979.
- [BST88] D. V. Batorsky, D. R. Spears, and A. R. Tedesco. The evolution of broadband network architectures. In *Proc. GLOBECOM '88*, pages 367–373, Hollywood, FL, November 1988. IEEE.
- [BT89] Richard G. Bubenik and Jonathan S. Turner. Performance of a broadcast packet switch. *IEEE Transactions on Communications*, 37(1):60–69, January 1989.
- [Car89] W. T. Carter. An evolutionary path to integrated communications using pure ATM techniques. In *Proc. ICC '89*, pages 695–698, Boston, June 1989. IEEE.
- [CCI88] CCITT. *Recommendation I.121: Broadband Aspects of ISDN*, 1988.
- [CKT89] Christos G. Cassandras, Michelle Hruby Kallmes, and Don Towsley. Optimal routing and flow control in networks with real-time traffic. In *Proc. INFOCOM '89*, pages 784–791, Ottawa, Ont., Canada, April 1989. IEEE.
- [DCLM89] Zbigniew Dziong, Jean Choquette, Ke-Qiang Liao, and Lorne Mason. Admission control and routing in ATM networks. In *Proc. ITC Specialist Seminar*, Adelaide, Australia, September 1989. Paper No. 15.3.

- [Dec87] Maurizio Decina. Evolution towards wideband communication networks. In *Proc. 1987 International Switching Symposium*, pages 417-422, Phoenix, AZ, March 1987. IEEE.
- [DJ81] Daniel M. Dias and Robert Jump. Analysis and simulation of buffered delta networks. *IEEE Transactions on Computers*, 30(4):273-282, April 1981.
- [DJ88] Lars Dittmann and Søren B. Jacobsen. Statistical multiplexing of identical bursty sources in an ATM network. In *Proc. GLOBECOM '88*, pages 1293-1297, Hollywood, FL, November 1988. IEEE.
- [DL85] J. N. Daigle and J. D. Langford. Queueing analysis of a packet voice communication system. In *INFOCOM '85*, pages 18-26, 1985.
- [DL86] John N. Daigle and Joseph D. Langford. Models for analysis of packet voice communications systems. *IEEE Journal on Selected Areas in Communications*, 4(6):847-855, September 1986.
- [DQ87] M. Dieudonne and M. Quinquis. Switching techniques for asynchronous time-division multiplexing (or fast packet switching). In *Proc. 1987 International Switching Symposium*, pages 367-372, Phoenix, AZ, March 1987. IEEE.
- [DT89] George E. Daddis, Jr. and H. C. Torng. A taxonomy of broadband integrated switching architectures. *IEEE Communications Magazine*, 27(5):32-42, May 1989.
- [DT90] M. Decina and T. Toniatti. On bandwidth allocation to bursty virtual connections in ATM networks. In *Proc. ICC '90*, Atlanta, GA, April 1990. IEEE.
- [DTVV90] M. Decina, T. Toniatti, P. Vaccari, and L. Verri. Bandwidth assignment and virtual call blocking in ATM networks. In *Proc. INFOCOM '90*, San Francisco, CA, June 1990. IEEE.
- [Eck79] Adrian E. Eckberg, Jr. The single server queue with periodic arrival process and deterministic service times. *IEEE Transactions on Communications*, 27(3):556-562, March 1979.
- [EH88] A. E. Eckberg and T.-C. Hou. Effects of output buffer sharing on buffer requirements in an ATDM packet switch. In *Proc. INFOCOM '88*, pages 459-466, New Orleans, Louisiana, March 1988. IEEE.

- [ELL89] A. E. Eckberg, D. T. Luan, and D. M. Lucantoni. Bandwidth management: A congestion control strategy for broadband packet networks - characterizing the throughput-burstiness filter. In *Proc. ITC Specialist Seminar*, Adelaide, Australia, September 1989. Paper No. 4.4.
- [ELLW89] M. I. Eiger, H. L. Lemberg, K. W. Lu, and S. S. Wagner. Cost analyses of emerging broadband fiber loop architectures. In *Proc. ICC '89*, pages 156-161, Boston, June 1989. IEEE.
- [ES89] Kai Y. Eng and Mario Santoro. Multi-Gb/s optical cross-connect switch architectures: TDM versus FDM techniques. In *Proc. GLOBECOM '89*, pages 999-1003, Dallas, TX, November 1989. IEEE.
- [ESS88] Berth Eklundh, Krister Sällberg, and Bengt Stavenow. Asynchronous transfer modes - options and characteristics. In *Proc. 12th International Teletraffic Congress*, pages 1.3A.3.1-7, Torino, Italy, June 1988.
- [Fen81] Tse-Yun Feng. A survey of interconnection networks. *IEEE Computer Magazine*, pages 12-27, December 1981.
- [FP88] Jerome S. Fleischman and P. E. Proctor. Operations planning for the information age. *IEEE Journal on Selected Areas in Communications*, 6(4):633-640, May 1988.
- [Ger88] M. Gerla. Treenet, a multi-level fiber optics MAN. In *Proc. INFOCOM '88*, pages 363-372. IEEE, 1988.
- [GF88] M. Gerla and L. Fratta. Design and control in processor limited packet networks. In *Proc. 12th International Teletraffic Congress*, pages 5.2B.5.1-7, Torino, Italy, June 1988.
- [GH87] P. Gerke and J. F. Huber. Fast packet switching - a principle for future system generations? In *Proc. 1987 International Switching Symposium*, pages 373-379, Phoenix, AZ, March 1987. IEEE.
- [GK77] Mario Gerla and Leonard Kleinrock. On topological design of distributed computer networks. *IEEE Transactions on Communications*, 25(1):48-60, January 1977.



- [GkKW90] Gita Gopal, Chong kwon Kim, and Abel Weinrib. Dynamic network configuration management. In *Proc. ICC 90*, pages 295–301, Atlanta, GA, April 1990.
- [GL89] Alexander Gersht and Kyoo Jeong Lee. A congestion control framework for ATM networks. In *Proc. INFOCOM '89*, pages 701–710, Ottawa, Ont., Canada, April 1989. IEEE.
- [GL73] L. Rodney Goke and G. J. Lipovski. Banyan networks for partitioning multiprocessor systems. In G.J. Lipovski and S.A. Szygenda, editors, *Proc. 1st Annual Symposium on Computer Architecture*, pages 21–28, Gainesville, FL, 73. ACM.
- [GO89] Jerry Gechter and Peter O'Reilly. Conceptual issues for ATM. *IEEE Network*, 3(1):14–16, January 1989.
- [GRF89] G. Gallassi, G. Rigolio, and L. Fratta. ATM: Bandwidth assignment and bandwidth enforcement policies. In *Proc. GLOBECOM '89*, pages 1788–1793, Dallas, TX, November 1989. IEEE.
- [Gro87] W. D. Grover. The SELFHEALING network – a fast distributed restoration technique for networks using digital crossconnect machines. In *Proc. GLOBECOM '87*, pages 1090–1095, Tokio, Japan, November 1987. IEEE.
- [HA87] Joseph Y. Hui and Edward Arthurs. A broadband packet switch for integrated transport. *IEEE Journal on Selected Areas in Communications*, 5(8):1264–1273, October 1987.
- [HA88] Kazuo Hagimoto and Kazuo Aida. Multigigabit-per-second optical baseband transmission system. *Journal of Lightwave Technology*, 6(11):1678–1685, November 1988.
- [Hab88] Kohei Habara. ISDN: A look at the future through the past. *IEEE Communications Magazine*, 26(11):25–32, November 1988.
- [HK84] Alan Huang and Scott Knauer. Starlite: a wideband digital switch. In *Proc. GLOBECOM '84*, pages 121–125, December 1984.
- [HK88] Michael G. Hluchyj and Mark J. Karol. Queueing in high-performance packet switching. *IEEE Journal on Selected Areas in Communications*, 6(9):1587–1597, December 1988.

- [HL86] Harry Hefes and David M. Lucantoni. A Markov modulated characterization of packetized voice and data traffic and related statistical multiplexer performance. *IEEE Journal on Selected Areas in Communications*, 6(4):856-868, September 1986.
- [HMI+88] K. Hajikano, K. Murakami, E. Iwabuchi, O. Isono, and T. Kobayashi. Asynchronous transfer mode switching architecture for broadband ISDN - multistage self-routing switching (MSSR). In *Proc. International Conference on Communications*, pages 911-915, Philadelphia, PA, June 1988. IEEE.
- [Hui87] Joseph Hui. A broadband packet switch for multi-rate services. In *Proc. International Conference on Communications*, pages 782-788, Seattle, Washington, June 1987.
- [Hui88] Joseph Y. Hui. Resource allocation for broadband networks. *IEEE Journal on Selected Areas in Communications*, 6(9):1598-1608, December 1988.
- [Hui90] Joseph Y. Hui. *Switching and Traffic Theory for Integrated Broadband Networks*. Kluwer Academic Publishers, Boston, MA, 1990.
- [IK88] Hajime Imai and Takao Kaneda. High-speed distributed feedback lasers and InGaAs avalanche photodiodes. *Journal of Lightwave Technology*, 6(11):1634-1642, November 1988.
- [Jac90] Andrew R. Jacob. A survey of fast packet switches. *Computer Communication Review*, 20(1):54-64, January 1990.
- [Jen83] Yih-Chyun Jenq. Performance analysis of a packet switch based on single-buffered banyan network. *IEEE Journal on Selected Areas in Communications*, 1(6):1014-1021, December 1983.
- [JLL89] A. E. Eckberg Jr., D. T. Luan, and D. M. Lucantoni. Meeting the challenge: Congestion and flow control strategies for broadband information transport. In *Proc. GLOBECOM '89*, pages 1769-1773, Dallas, TX, November 1989. IEEE.
- [JV89] P. Joos and W. Verbiest. A statistical bandwidth allocation and usage monitoring algorithm for ATM networks. In *Proc. ICC 89*, pages 415-422, Boston, MA, June 1989. IEEE.

- [KHM87] Mark J. Karol, Michael G. Hluchy, and Samuel P. Morgan. Input versus output queueing on a space-division packet switch. *IEEE Trans. on Communications*, 35(12):1347-1356, December 1987.
- [Kle72] Leonard Kleinrock. *Communication Nets-Stochastic Message Flow and Delay*. Dover Publications, New York, 1972.
- [Kle75] Leonard Kleinrock. *Queueing Systems, Vol I: Theory*. Wiley-Interscience, New York, 1975.
- [Kle76] Leonard Kleinrock. *Queueing Systems, Vol II: Computer Applications*. Wiley-Interscience, New York, 1976.
- [KLG88] Hyong Sok Kim and Alberto Leon-Garcia. Performance of buffered banyan networks under nonuniform traffic patterns. In *Proc. INFOCOM '88*, pages 344-353, New Orleans, Louisiana, March 1988. IEEE.
- [KS83] Clyde P. Kruskal and Marc Snir. The performance of multistage interconnection networks for multiprocessors. *IEEE Transactions on Computers*, 32(12):1091-1098, December 1983.
- [KS89] K. Kawashima and H. Saito. Teletraffic issues in ATM networks. In *Proc. ITC Specialist Seminar*, Adelaide, Australia, September 1989. Paper No. 17.5.
- [KSHM88] Yuji Kato, Toshio Shimoe, Kazuo Hajikano, and Koso Murakami. Experimental broadband ATM switching system. In *Proc. GLOBECOM '88*, pages 1288-1292, Hollywood, FL, November 1988. IEEE.
- [KSW86] Clyde P. Kruskal, Marc Snir, and Alan Weiss. The distribution of waiting times in clocked multistage interconnection networks. In *Proc. 1986 International Conference on Parallel Processing*, pages 12-19, University Park, PA, August 1986. IEEE.
- [Kue89] Paul J. Kuehn. From ISDN to IBCN (integrated broadband communication network). In *Proc. IFIP*, pages 479-486. North-Holland, 1989.
- [LBG88] J.-R. Louvion, P. Boyer, and A. Gravey. A discrete-time single server queue with Bernoulli arrivals and constant service time. In *Proc. 12th International Teletraffic Congress*, pages 2.4B.2.1-2.4B.2.8, Torino, Italy, June 1988.

- [LC89] S. C. Liew and K. W. Cheung. A broadband optical local network based on multiple wavelengths and multiple RF subcarriers. In *Proc. ICC '89*, pages 162–170, Boston, June 1989. IEEE.
- [Lea86] Chin-Tau A. Lea. The load-sharing banyan network. *IEEE Transactions on Computers*, C-35(12):1025–1034, December 1986.
- [Li89] San-Qi Li. Study of information loss in packet voice systems. *IEEE Transactions on Communications*, 37(11):1192–1202, November 1989.
- [Lio90] Ming L. Liou. Visual telephony as an ISDN application. *IEEE Communications Magazine*, 28(2):30–38, February 1990.
- [LY89] Ming-Jeng Lee and James R. Yee. An efficient near-optimal algorithm for the joint traffic and trunk routing problem in self-planning networks. In *Proc. IEEE INFOCOM '89*, pages 127–135, Ottawa, Ont., Canada, April 1989. IEEE.
- [Mar75] B. Martos. *Nonlinear Programming Theory and Methods*, pages 238–248. North-Holland Publishing Co., Amsterdam-Oxford, 1975.
- [MAS+87] Basil Maglaris, Dimitris Anastassiou, Prodipp Sen, Gunnar Karlsson, and John Robbins. Performance analysis of statistical multiplexing for packet video sources. In *Proc. GLOBECOM '87*, pages 1890–1899, Tokio, Japan, November 1987. IEEE.
- [MAS+88] Basil Maglaris, Dimitris Anastassiou, Prodipp Sen, Gunnar Karlsson, and John D. Robbins. Performance models of statistical multiplexing in packet video communications. *IEEE Transactions on Communications*, 36(7):834–844, July 1988.
- [Min89] Steven E. Minzer. Broadband ISDN and asynchronous transfer mode (ATM). *IEEE Communications Magazine*, 27(9):17–24, September 1989.
- [MS89] Steven E. Minzer and Dan R. Spears. New directions in signaling for Broadband ISDN. *IEEE Communications Magazine*, 27(2):6–14, February 1989.
- [Mur76] K. Murty. *Linear and Combinatorial Programming*, pages 187–189. John Wiley and Sons, New York, 1976.

- [MVB89] Bogdan Materna, Brian J. N. Vaughan, and Charles W. Britney. Evolution from LAN and MAN access networks towards the integrated ATM network. In *Proc. GLOBECOM '89*, pages 1455–1461, Dallas, TX, November 1989. IEEE.
- [NH87] Tomy M. J. Ng and Doan B. Hoang. Joint optimization of capacity and flow assignment in a packet-switched communications network. *IEEE Transactions on Communications*, 35(2):202–209, February 1987.
- [NTFH87] Satoshi Nojima, Eiichi Tsutui, Haruki Fukuda, and Masamichi Hashimoto. Integrated services packet network using bus matrix switch. *IEEE Journal on Selected Areas in Communications*, 5(8):1284–1292, October 1987.
- [OMKM89] Yuji Oie, Masayuki Murata, Koji Kubota, and Hideo Miyahara. Effect of speedup in nonblocking packet switch. In *Proc. ICC '89*, pages 410–414, Boston, June 1989. IEEE.
- [OSMM90a] Yuji Oie, Tatsuya Suda, Masayuki Murata, and Hideo Miyahara. Survey of switching techniques in high-speed networks and their performance. In *Proc. INFOCOM '90*, pages 1242–1251, San Francisco, CA, June 1990. IEEE.
- [OSMM90b] Yuji Oie, Tatsuya Suda, Masayuki Murata, and Hideo Miyahara. Survey of the performance of nonblocking switches with FIFO input buffers. In *Proc. ICC '90*, pages 737–741, Atlanta, GA, April 1990. IEEE.
- [Pan87] Sushil N. Pandhi. The universal data connection. *SPECTRUM*, 24(7):31–37, July 1987.
- [Pat88] Achille Pattavina. Multichannel bandwidth allocation in a broadband packet switch. *IEEE Journal on Selected Areas in Communications*, 6(9):1489–1499, December 1988.
- [PLK89] Michal Pióro, Józef Lubacz, and Ulf Körner. Traffic engineering problems in multiservice circuit switched networks. In *Proc. ITC Specialist Seminar*, Adelaide, Australia, September 1989. Paper No. 11.3.

- [PRG82] R. A. Pazos-Rangel and M. Gerla. Express pipe networks. In *Global Telecommunications Conference*, pages B2.3.1-5, Miami, FL, December 1982.
- [PW71] Peters and Wilkinson. *Handbook for Automatic Computation*, volume II, pages 372-395. 1971.
- [Rat90] Erwin P. Rathgeb. Comparison of policing mechanisms for ATM networks. Submitted to INFOCOM '90, 1990.
- [RD89] G. Ramamurthy and R. S. Dighe. Integration of high speed continuous stream data traffic in a broadband packet network. In *Proc. INFOCOM '89*, pages 1063-1071, Ottawa, Ont., Canada, April 1989. IEEE.
- [Roc87] Edouard Y. Rocher. Information outlet, ULAN versus ISDN. *IEEE Communications Magazine*, 25(4):18-32, April 1987.
- [Ros86] F. Ross. FDDI — a tutorial. *IEEE Communications Magazine*, 24(5):10-18, 1986.
- [Rub75] Izhak Rubin. Message path delays in packet-switching communication networks. *IEEE Transactions on Communications*, 23(2):186-192, February 1975.
- [SA88] Hiromichi Shinohara and Koichi Asatani. Evolutionary approach to broadband services and network. In *Proc. GLOBECOM '88*, pages 387-393, Hollywood, FL, November 1988. IEEE.
- [Sie90] Howard Jay Siegel. *Interconnection Networks for Large-Scale Parallel Processing - Theory and Case Studies*. McGraw-Hill, New York, NY, 2nd edition, 1990.
- [SKY89] P. W. Shumate, O. Krumpholz, and K. Yamaguchi. Special Issue on Subscriber Loop Technology. *Journal of Lightwave Technology*, 7(11), November 1989.
- [Sta89] William Stallings. *ISDN: An Introduction*. Macmillan Publishing Co., New York, NY, 1989.
- [TCS84] A. Thomas, J. P. Coudreuse, and M. Serval. Asynchronous time-division techniques: An experimental packet network integrating

- videocommunication. In AEI, editor, *Proc. XI International Switching Symposium*, pages 32 C - 2, Florence, Italy, May 1984. North-Holland.
- [TGA88] Phuoc Tran-Gia and Hamid Ahmadi. Analysis of a discrete-time  $G^{(X)}/D/1-S$  queueing system with applications in packet-switching systems. In *Proc. INFOCOM '88*, pages 861-870, New Orleans, Louisiana, March 1988. IEEE.
- [Tru54] C. J. Truitt. Traffic engineering techniques for determining trunk requirements in alternate routing trunk networks. *The Bell System Technical Journal*, 33(2):277-302, March 1954.
- [Tuc88] Roger C. F. Tucker. Accurate method for analysis of a packet-speech multiplexer with limited delay. *IEEE Transactions on Communications*, 36(4):479-483, April 1988.
- [Tur86a] Jonathan S. Turner. Design of an Integrated Services Packet Network. *IEEE Journal on Selected Areas in Communications*, 4(8):1373-1380, November 1986.
- [Tur86b] Jonathan S. Turner. New directions in communications (or which way to the information age?). *IEEE Communications Magazine*, 24(10):8-15, October 1986.
- [Tur87] Jonathan S. Turner. The challenge of multipoint communication. In *Proc. 5th ITC Seminar*, Lake Como, Italy, May 1987.
- [TW83] Jonathan S. Turner and Leonard F. Wyatt. A packet network architecture for integrated services. In *Proc. GLOBECOM '83*, pages 45-50, San Diego, CA, Nov./Dec. 1983. IEEE.
- [Ver90] Pramode K. Verma. *ISDN Systems*. Prentice Hall, Englewood Cliffs, NJ, 1990.
- [VPV88] W. Verbiest, L. Pinnoo, and B. Voeten. Statistical multiplexing of variable bit rate video sources in asynchronous transfer mode networks. In *Proc. GLOBECOM '88*, pages 208-213, Hollywood, FL, November 1988. IEEE.
- [VR89] J. T. Virtamo and J. W. Roberts. Evaluating buffer requirements in an ATM multiplexer. In *Proc. GLOBECOM '89*, pages 1473-1477, Dallas, TX, November 1989. IEEE.

- [VV87] Richard Vickers and Toomas Vilmansen. The evolution of telecommunications technology. *IEEE Communications Magazine*, 25(7):6-18, July 1987.
- [Wei78] C. J. Weinstein. Fractional speech loss and talker activity model for TASI and for packet-switched speech. *IEEE Transactions on Communications*, 26(8):1253-1257, August 1978.
- [Wei87] Stephen B. Weinstein. Telecommunications in the coming decades. *IEEE Spectrum*, 24(11):62-67, November 1987.
- [Wei90] Stephen B. Weinstein (Pramode K. Verma Ed.). *ISDN Systems*, chapter 8, pages 262-304. Prentice Hall, Englewood Cliffs, NJ, 1990.
- [WKFR89] Gillian Woodruff, Rungroj Kositpaiboon, Gordon Fitzpatrick, and Philip Richards. Control of ATM statistical multiplexing performance. In *Proc. ITC Specialist Seminar*, Adelaide, Australia, September 1989. Paper No. 17.2.
- [Wu85] L. T. Wu. Mixing traffic in a buffered banyan network. In *Proc. Proc. ACM 9th Data Communications Symposium*, pages 134-139. ACM, September 1985.
- [YH88] C. Han Yang and Satoshi Hasegawa. FITNESS - failure immunization technology for network service survivability. In *Proc. GLOBECOM '88*, pages 1549-1554, Hollywood, FL, November 1988. IEEE.
- [YHA87] Yu-Shuan Yeh, Michael G. Hluchyj, and Anthony S. Acampora. The Knockout switch: A simple, modular architecture for high-performance packet switching. *IEEE Journal on Selected Areas in Communications*, 5(8):1274-1283, October 1987.
- [Zan88] Pamela J. Zanella. Customer network reconfiguration applications utilizing digital cross-connect systems. In *Proc. GLOBECOM '88*, pages 1538-1543, Hollywood, FL, November 1988. IEEE.