

UNIVERSIDADE FEDERAL DE SANTA CATARINA  
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA  
CURSO DE PÓS-GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO

**AVALIAÇÃO DE DESEMPENHO  
DE COMUTADORES ATM**

DISSERTAÇÃO SUBMETIDA À UNIVERSIDADE FEDERAL DE SANTA  
CATARINA PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIA  
DA COMPUTAÇÃO

Jane Ferreira Cunha

Florianópolis, agosto de 1998

# Avaliação de Desempenho de Comutadores ATM

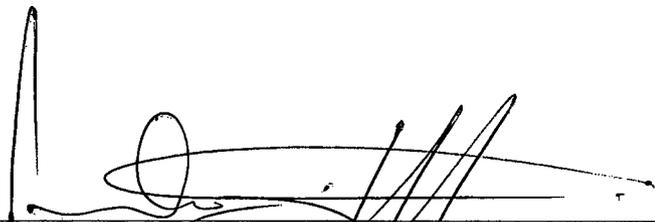
JANE FERREIRA CUNHA

Esta dissertação foi julgada adequada para a obtenção do título de Mestre em Ciência da Computação, e aprovada em sua forma final pelo Programa de Pós-Graduação

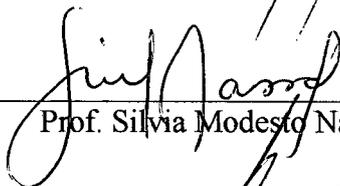


Prof. Jorge Muniz Barreto, Dr.  
Coordenador do Curso

**Banca Examinadora :**



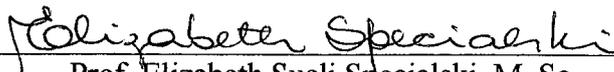
Prof. Paulo José de Freitas Filho, Dr.  
Orientador



Prof. Silvia Modesto Nassar, Dra.



Prof. Vittorio Bruno Mazzola, Dr.



Prof. Elizabeth Sueli Specialski, M. Sc.



Prof. Maria Marta Leite, M. Sc.

## **AGRADECIMENTOS**

Ao meu orientador Paulo Freitas, que sempre acreditou em mim

À Beth e à Marta, que me levaram para a área de Redes de Computadores, e têm me guiado nela desde então

À professora Silvia, por toda a ajuda no tratamento estatístico do trabalho

Ao professor Vitório, por ter aceitado participar da banca, mesmo sem ter muito tempo para ler o trabalho

Aos meu chefe Daniel, pelo apoio

Aos meus colegas Ana Lúcia, Iberê e Valesca, pelo apoio e cobertura, nas muitas horas em que precisei faltar ao trabalho

Às minhas amigas Nara e Janete, sempre prontas a me dar uma palavra animadora

## RESUMO

A tecnologia ATM vem despontando como a solução para a transmissão de vários tipos de mídia pelas redes locais e de longa distância, uma vez que alia a alta velocidade à qualidade de serviço exigida por cada uma destas mídias.

Entretanto, como em toda tecnologia emergente, ainda há uma grande confusão no mercado, gerada pela avalanche de produtos sendo lançados. Vários fabricantes brigam pela liderança, alegando que as características dos seus produtos (principalmente os comutadores ATM), fazem deles os melhores, mais eficientes e mais rápidos. No caso específico do desempenho, a decisão por um determinado equipamento torna-se, então, bastante penosa, já que além da multiplicidade de opções, ainda não foram definidos pelos organismos de padronização (em especial o ATM Forum e o ITU-T) um conjunto de padrões de desempenho que possam servir de base para uma avaliação adequada deste aspecto, e nem se sabe se tal definição é viável.

Assim, o objetivo deste trabalho é iniciar uma investigação que possa indicar quais são os parâmetros dos comutadores ATM que mais influenciam no seu desempenho. Considerando que este equipamento é de longe o mais caro e o mais importante numa rede ATM, a escolha de um comutador com bom desempenho tem um grande significado no projeto e implantação de uma rede ATM.

Para esta investigação foi escolhida a técnica da simulação, uma vez que não estava disponível um ambiente real de testes. Baseando-se em descrições de comutadores disponíveis na literatura técnica, foram construídos modelos e executados uma série de experimentos, de forma a obter-se valores referentes à latência e taxa de descarte. Apesar das limitações impostas ao trabalho pela grande abrangência do tema, foi possível tirar-se algumas conclusões importantes, como a demonstração da necessidade de se projetar com cuidado a capacidade dos *buffers*.

# ABSTRACT

ATM technology is proving to be the solution for transmitting many kinds of media through LANs and WANs, because it puts together the high speed and quality of service they demand.

However, as every emerging technology, there's still a great deal of confusion in the market, brought by the great number of products being made available. Many manufacturers fight for the biggest market share, claiming that the features of their products (especially ATM switches) make them the best, fastest, and most efficient. If one talks about performance, the decision to buy a given device becomes very difficult. There is a broad array of options, and performance standards haven't been defined so far by organizations such as ATM Forum and ITU-T, that might be used as references in a suitable evaluation, and one doesn't even know if such definition is possible.

Therefore, this work's goal is to begin an investigation that may point out which parameters of an ATM switch have more influence on the performance. Considering that this device is the most important and the most expensive in an ATM network, the choice of a switch that shows good performance has a significant meaning in the design and implementation of such a network.

The simulation technique was chosen for this investigation, for there wasn't an actual test environment available. Based on switches descriptions found in the literature, models were built and a series of experiments were made, to get values for latency and discard rate. In spite of the limitations imposed to the work by the theme's breadth, some important conclusions were drawn, like showing the need of a careful design of buffers capacity.

# GLOSSÁRIO

AAL	- ATM Adaptation Layer
ABR	- Available Bit Rate
ANOVA	- Analysis of Variance
ARP	- Address Resolution Protocol
ATM	- Asynchronous Transfer Mode
BUS	- Broadcast and Unknown Server
CAC	- Connection Admission Control
CBR	- Constant Bit Rate
CLP	- Cell Loss Priority
CPCS	- Common Part Convergence Sublayer
CS	- Convergence Sublayer
EPD	- Early Packet Discard
GFC	- Generic Flow Control
HEC	- Header Error Control
IETF	- Internet Engineering Task Force
IP	- Internet Protocol
IPX	- Internetwork Packet Exchange
ITU-T	- International Telecommunications Union - Telecommunications
LAN	- Local Area Network
LEC	- LAN Emulation Client
LECS	- LAN Emulation Configuration Server
LES	- LAN Emulation Server
MAC	- Media Access Control

MPOA - Multiprotocol over ATM

NHRP - Next Hop Routing Protocol

NIST - National Institute of Standards and Technology

NNI - Network-to-Network Interface

OAM - Operations Administration and Maintenance

OSI - Open Systems Interconnection

PABX - Private Automatic Branch Exchange

PICS - Protocol Implementation Conformance Statements

PIXIT - Protocol Implementation eXtra Information for Testing

PM - Physical Medium

PPD - Partial Packet Discard

PT - Payload Type

PVC - Permanent Virtual Circuit

QoS - Quality of Service

RDSI - Rede Digital de Serviços Integrados

RDSI-FE - Rede Digital de Serviços Integrados – Faixa Estreita

RDSI-FL - Rede Digital de Serviços Integrados – Faixa Larga

RFC - Request for Comments

SAR - Segmentation and Reassembly

SM - Switch Management

SNA - System Network Architecture

SSCS - Service Specific Convergence Sublayer

SVC - Switched Virtual Circuit

TC - Transmission Convergence

TCP - Transmission Control Protocol

UNI - User-to-Network Interface

VBR-nrt - non real time Variable Bit Rate  
VBR-rt - real time Variable Bit Rate  
VC - Virtual Circuit  
VCC - Virtual Channel Connection  
VCI - Virtual Channel Identifier  
VP - Virtual Path  
VPI - Virtual Path Identifier

# SUMÁRIO

<b>GLOSSÁRIO .....</b>	<b>III</b>
<b>CAPÍTULO I - INTRODUÇÃO .....</b>	<b>1</b>
1.1 - HISTÓRICO .....	1
1.2 - DIFICULDADES NO PROJETO DE REDES ATM.....	2
1.3 – OBJETIVOS E ESTRUTURA DO TRABALHO .....	3
<b>CAPÍTULO II - A TECNOLOGIA ATM.....</b>	<b>5</b>
2.1 – INTRODUÇÃO.....	5
2.2 - A RDSI.....	5
2.3 - O PROTOCOLO ATM .....	6
2.4 - ARQUITETURA DOS COMUTADORES ATM.....	13
<b>CAPÍTULO III - ESTADO DA ARTE.....</b>	<b>17</b>
3.1 - INTRODUÇÃO.....	17
3.2 - ARQUITETURA DE COMUTADORES.....	17
3.3 - MODELAGEM E TESTES DE AVALIAÇÃO DE DESEMPENHO.....	18
3.3.1 - <i>Instituições de padronização</i> .....	19
3.3.2 - <i>Fabricantes</i> .....	21
3.3.3 - <i>Periódicos</i> .....	22
3.3.4 - <i>Artigos acadêmicos</i> .....	23
<b>CAPÍTULO IV - METODOLOGIA DE PESQUISA .....</b>	<b>27</b>
4.1 – INTRODUÇÃO .....	27
4.2 - DEFINIÇÃO DA METODOLOGIA.....	27
4.3 – APLICAÇÃO DA METODOLOGIA.....	30
4.3.1 - <i>Projeto e desenvolvimento dos modelos</i> .....	31
4.3.2 – <i>Especificação das métricas</i> .....	41
4.3.3 – <i>Definição dos parâmetros, fatores e carga</i> .....	42
4.3.4 – <i>Planejamento e realização dos experimentos</i> .....	46
<b>CAPÍTULO V - ANÁLISE ESTATÍSTICA DOS RESULTADOS OBTIDOS.....</b>	<b>48</b>
5.1 - INTRODUÇÃO.....	48
5.2 - ESTATÍSTICAS BÁSICAS .....	48
5.3 - ANÁLISE DE VARIÂNCIA .....	50
5.3.1 – <i>Análise de variância para o grupo de métricas de latência</i> .....	50

5.3.2 – <i>Análise de variância para o grupo de métricas de taxa de descarte</i> .....	51
5.4 – OUTRAS INFORMAÇÕES OBTIDAS DA ANÁLISE DE VARIÂNCIA .....	53
5.4.1 – <i>O coeficiente de determinação <math>R^2</math></i> .....	53
5.4.2 – <i>O coeficiente de correlação <math>r</math></i> .....	54
5.5 – INTERAÇÃO ENTRE A LATÊNCIA E A TAXA DE DESCARTE .....	55
<b>CAPÍTULO VI – COMENTÁRIOS FINAIS .....</b>	<b>60</b>
6.1 – INTRODUÇÃO .....	60
6.2 – RESULTADOS OBTIDOS .....	61
6.2.1 – <i>Latência</i> .....	61
6.2.2 – <i>Taxa de descarte</i> .....	62
6.2.3 – <i>Resumo dos resultados</i> .....	63
6.3 – IMPORTÂNCIA DO TRABALHO .....	64
6.4 – LIMITAÇÕES .....	64
6.5 – SUGESTÕES PARA TRABALHOS FUTUROS .....	65
<b>CAPÍTULO VII - BIBLIOGRAFIA.....</b>	<b>66</b>
<b>CAPÍTULO VIII – ANEXOS.....</b>	<b>71</b>
ANEXO A – TABELAS DE RESULTADOS OBTIDOS NO STATISTICA .....	71
TABELAS COM TESTE F PARA AS VARIÁVEIS DE LATÊNCIA .....	71
TABELAS COM TESTE F PARA AS VARIÁVEIS DE TAXA DE DESCARTE.....	72
TABELAS DE MÉDIA E DESVIO PADRÃO PARA A MÉTRICA LATÊNCIA GERAL POR FATOR .....	73
TABELAS DE MÉDIA E DESVIO PADRÃO PARA A MÉTRICA LATÊNCIA CBR POR FATOR .....	75
TABELAS DE MÉDIA E DESVIO PADRÃO PARA A MÉTRICA LATÊNCIA VBR POR FATOR .....	77
TABELAS DE MÉDIA E DESVIO PADRÃO PARA A MÉTRICA LATÊNCIA ABR POR FATOR .....	78
TABELAS DE MÉDIA E DESVIO PADRÃO PARA A MÉTRICA LATÊNCIA UBR POR FATOR .....	80
TABELAS DE MÉDIA E DESVIO PADRÃO PARA A MÉTRICA TAXA DE DESCARTE GERAL POR FATOR.....	82
TABELAS DE MÉDIA E DESVIO PADRÃO PARA A MÉTRICA TAXA DE DESCARTE CBR POR FATOR.....	83
TABELAS DE MÉDIA E DESVIO PADRÃO PARA A MÉTRICA TAXA DE DESCARTE VBR POR FATOR .....	85
TABELAS DE MÉDIA E DESVIO PADRÃO PARA A MÉTRICA TAXA DE DESCARTE ABR POR FATOR .....	87
TABELAS DE MÉDIA E DESVIO PADRÃO PARA A MÉTRICA TAXA DE DESCARTE UBR POR FATOR .....	88
ANEXO B – DESCRIÇÃO DOS MODELOS .....	91
<i>PRÉ-PROCESSAMENTO</i> .....	92
<i>ENTRADA NA MATRIZ DE COMUTAÇÃO</i> .....	93
<i>MATRIZ DE COMUTAÇÃO</i> .....	93
<i>PÓS-PROCESSAMENTO</i> .....	95

## LISTA DE FIGURAS

Figura 2.1 - Os planos no modelo de referência ATM.....	8
Figura 2.2 - Formato das células ATM.....	9
Figura 2.3 - Arquitetura genérica de um comutador ATM.....	14
Figura 4.1 – Fluxo geral da dentro do comutador .....	32
Figura 4.2 – Matriz com arquitetura de memória compartilhada (a) com prioridade e (b) sem prioridade .....	34
Figura 4.3 – Matriz com arquitetura de interconexão total (a) com prioridade e (b) sem prioridade.....	35
Figura 4.4 – Matriz com arquitetura de múltiplos estágios, (a) com prioridade e (b) sem prioridade.....	36
Figura 5.1 – Diagrama de dispersão para as métricas gerais de latência e taxa de descarte .....	55
Figura B.1 – Pré-processamento nas matrizes de memória e meio compartilhados .....	96
Figura B.2 – Pré-processamento da matriz com múltiplos estágios.....	97
Figura B.3 – Entrada da matriz de memória compartilhada com divisão por prioridade de tráfego .....	98
Figura B.4 – Entrada da matriz de memória compartilhada sem divisão por prioridade de tráfego .....	99
Figura B.5 – Entrada da matriz de meio compartilhado com divisão por prioridade de tráfego.....	100
Figura B.6 – Entrada da matriz de meio compartilhado sem divisão por prioridade de tráfego.....	101
Figura B.7 – Entrada da matriz de múltiplos estágios com divisão por prioridade de tráfego.....	102
Figura B.8 – Entrada da matriz de múltiplos estágios sem divisão por prioridade de tráfego.....	103

Figura B.9 – Matriz de memória compartilhada.....	104
Figura B.10 – Matriz de meio compartilhado.....	105
Figura B.11 – Matriz de múltiplos estágios (parte 1).....	106
Figura B.12 – Matriz de múltiplos estágios (parte 2).....	107
Figura B.13 – Segmento de pós-processamento.....	108

## LISTA DE TABELAS

Tabela 5.1 – Estatísticas descritivas relativas à latência (em $\mu\text{s}$ ) .....	49
Tabela 5.2 – Estatísticas descritivas relativas à taxa de descarte (em %) .....	49
Tabela 5.3 – Teste F para a variável LATÊNCIA.....	51
Tabela 5.4 – Teste F para a variável TAXA-DESCARTE.....	52
Tabela 5.5 – Teste F para a variável TAXA-DESCARTE-CBR .....	53
Tabela 5.6 – Valores de $R^2$ para as variáveis em estudo.....	54
Tabela 5.7 – Valores de $r$ para as métricas do grupo de latência, para cada fator .....	54
Tabela 5.8 – Valores de $r$ para as métricas do grupo de descarte, para cada fator.....	55

# CAPÍTULO I - INTRODUÇÃO

## 1.1 - Histórico

Desde a década de 50, quando os cientistas apresentaram ao mundo o computador eletrônico, a sociedade vem registrando avanços tecnológicos inacreditáveis. No começo, aqueles gigantes de luzes e fios tinham por função facilitar cálculos e realizar algum processamento básico, desde que o indivíduo que o operava tivesse um profundo conhecimento sobre o modo pelo qual o computador tinha sido construído e funcionava.

Duas décadas depois, no entanto, estas máquinas já tinham evoluído para um ponto em que já eram capazes de prestar bons serviços a grandes corporações. Nesta época, surge o conceito de processamento remoto, ou seja, terminais geograficamente dispersos acessando um computador central. Era um embrião das redes, que nasceram de verdade quando estes grandes equipamentos centrais foram interligados. Nesta fase inicial, temos o surgimento de duas figuras de destaque: a arquitetura SNA (*System Network Architecture*), proprietária da IBM, ainda hoje largamente utilizada, e a rede científica que daria lugar mais tarde ao meio de comunicação mais falado nos últimos anos: a Internet. Só o que trafegava então eram dados textuais, e as velocidades de 1200 bps (bits por segundo) e 2400 bps eram bastante comuns e adequadas.

Nos anos oitentas houve uma grande mudança, com o aparecimento dos computadores pessoais. Finalmente, o poder da informação era tirado das mãos do tirânico CPD (Centro de Processamento de Dados), e posto ao alcance de qualquer pessoa. O sucesso foi instantâneo. Podia-se agora ter em casa uma pequena máquina com mais recursos do que os “gigantes” de 20 anos atrás.

Foram nestes anos mais recentes, no entanto, que assistimos a uma transformação sem precedentes na forma pela qual nos comunicamos e informamos. Tão ou mais importante que a interligação de sistemas é a interligação direta de pessoas, graças à Internet.

Assim, com as novas utilidades, cresce a necessidade de trafegar vários tipos de mídias além da palavra escrita, notadamente voz e vídeo. A voz, porque embora a rede telefônica tenha se desenvolvido de forma independente em relação à rede de comunicação de dados, a tendência é que a fronteira entre elas fique cada menos clara, até que elas se tornem uma só. O vídeo, porque agora as imagens em movimento são cada vez menos monopólio da televisão, e entram nas casas e nas empresas via computador. Esta necessidade esbarra, entretanto, nas limitações da infraestrutura das redes atuais, ainda projetadas para atender a uma demanda de dados baseados em texto, como no início. Foi esta necessidade que levou à criação da arquitetura ATM (*Asynchronous Transfer Mode*), que promete altas velocidades e suporte aos variados usos das redes que hoje se esboçam.

Empresas de maior porte já começam a implantar redes com tecnologia ATM. Algumas em proveito próprio, outras como infraestrutura para venda de serviços. Um exemplo flagrante são as empresas de telecomunicações, como a Telesc (Telecomunicações de Santa Catarina), que a partir de setembro de 1998, começa a instalar os equipamentos que formarão o *backbone* ATM estadual, que será usado para a extensão da rede corporativa, e também como o meio para o oferecimento de serviços ligados à comunicação de dados. Além disso, instituições educacionais como a Universidade Federal de Santa Catarina vêem no ATM o melhor meio para novos métodos de ensino e pesquisa, como a videoconferência e o ensino a distância.

## **1.2 - Dificuldades no projeto de redes ATM**

Por se tratar de uma tecnologia nova, o ATM ainda carece de várias definições, padronizações e experiências sólidas no mercado. Ainda não há, por parte dos organismos internacionais de padronização, a definição de uma metodologia para avaliar vários aspectos dos produtos que estão no mercado, aí incluindo o desempenho (o ATM Forum, no entanto, já publicou alguma diretrizes para testes de conformidade, interoperabilidade e desempenho). Desta forma, quando se faz necessária uma compra de qualquer dos elementos de uma rede ATM, faltam parâmetros de comparação para que se possa decidir pelos itens mais adequados. Dentre estes elementos, sem dúvida alguma, os comutadores são os principais, tanto pelo papel central que ocupam numa

rede ATM, quanto pelo preço. Portanto, é desta lacuna hoje existente, e da relevância destas informações, que vem a motivação deste trabalho.

Há que se ressaltar que a obtenção destes dados não se constitui numa tarefa fácil. Os comutadores são equipamentos bastante complexos, uma vez que eles se propõem a integrar os vários serviços de telecomunicações existentes atualmente (e quiçá os que venham a ser criados). Múltiplas configurações de *hardware* e *software* são possíveis, e à medida em que os padrões vão sendo publicados, o funcionamento deles muda.

### 1.3 – Objetivos e estrutura do trabalho

Diante do exposto na seção anterior, este trabalho se propõe a atingir os seguintes objetivos:

- ◆ Identificar os principais componentes e processos existentes num comutador ATM;
- ◆ Definir como é a interação entre estes componentes e processos, com base em algumas alternativas de projeto de comutadores, em especial relacionadas à arquitetura da matriz de comutação e aos *buffers*;
- ◆ Testar o comportamento do sistema sob diversas situações de tráfego e entender, de forma geral e específica para cada situação, quais são os parâmetros do comutador que tiveram maior importância no que se refere ao desempenho;
- ◆ Finalmente, especificar os valores destes parâmetros que fizeram o comutador alcançar o melhor desempenho.

Para que esta tarefa fosse factível, o comutador foi visto como um sistema isolado, ou seja, não se levou em conta a sua interação com a rede, outros comutadores ou computadores. Além disso, fixou-se nas características básicas comuns a todos os comutadores, com a intenção de se chegar a resultados que possam ser aplicados a qualquer equipamento disponível comercialmente.

O texto aqui apresentado divide-se da seguinte forma :

O capítulo 2 faz uma breve introdução à tecnologia ATM, bem como à arquitetura dos comutadores.

O capítulo 3 discorre sobre o estado da arte, tanto na área de funcionamento de comutadores como na área de modelagem e avaliação de desempenho.

O capítulo 4 descreve a metodologia de pesquisa e os modelos computacionais que serviram como fonte dos dados analisados.

O capítulo 5 mostra os resultados obtidos a partir de uma análise estatística dos dados, através de análise de variância, coeficiente de correlação e diagrama de dispersão.

O capítulo 6 contém os comentários finais, enfatizando os resultados obtidos e a importância do trabalho.

Por fim, tem-se a bibliografia e os anexos, que contêm os dados estatísticos obtidos na etapa de análise dos resultados e uma descrição mais detalhada dos modelos em ARENA.

# **CAPÍTULO II - A TECNOLOGIA ATM**

## **2.1 – Introdução**

A tecnologia ATM foi planejada de modo a refletir a evolução de várias tecnologias de uso corrente, tanto na área de telefonia e telecomunicações quanto na área de redes de computadores, e desta forma se tornar o ponto de união entre estes dois universos.

Este capítulo faz uma pequena introdução sobre o tema, destacando os pontos mais relevantes ao trabalho. Inicialmente apresenta-se a origem do ATM. Em seguida, descreve-se o protocolo ATM, como seus planos e camadas, terminando com a apresentação da arquitetura genérica de um comutador.

## **2.2 - A RDSI**

A evolução dos meios de comunicação era, até recentemente, a evolução individual de cada meio: o sistema telefônico, as redes de comunicação de dados, redes de televisão e assim por diante. Cada um procura melhorar as suas formas de transmissão para melhor atender ao serviço ao qual se destina, e a utilização deste meio por outro serviço não obtém resultados realmente satisfatórios. Seria possível o emprego de uma única rede que permitisse o acesso aos vários serviços, de uma forma econômica e ao mesmo tempo eficiente?

A União Internacional de Telecomunicações (ITU-T) assumiu a tarefa de padronizar este desejo, com a cooperação de outros organismos. O resultado do trabalho foi o surgimento da Rede Digital de Serviços Integrados (RDSI) que, além de ser um caminho para a disponibilização dos vários serviços, faz uso de uma rede digital (que é mais econômica e eficiente que a sua correspondente analógica) de um extremo a outro.

A RDSI definida inicialmente pelo ITU-T é o que hoje chamamos de RDSI-FE (RDSI Faixa Estreita), pois atinge velocidades de até 2,048 Mbps (padrão E1 europeu). Usa como transporte a rede telefônica, de comutação de circuitos. Mais recentemente,

porém, verificou-se que estas velocidades não seriam suficientes para o tráfego mais pesado que surgia, como a videoconferência. Modificações na RDSI foram feitas, e chegou-se à RDSI-FL (RDSI Faixa Larga), que pode chegar a velocidades medidas em gigabits por segundo. Como rede de transporte, foi definido um novo padrão, que ultrapassaria a esfera da RDSI e se tornaria também uma solução para a interconexão de redes de dados em altas velocidades : o ATM. O sucesso do ATM foi tão grande, que a pressão do mercado fez com que os principais fornecedores de infraestrutura de rede não esperassem pelas definições do ITU-T, e criassem uma associação própria, para definir padrões provisórios (que depois necessitam ser referendados pelo ITU-T) que lhes permitissem fabricar produtos que interoperassem com o de outros fabricantes, chamada ATM Forum.

## 2.3 - O PROTOCOLO ATM

O ATM é um protocolo orientado à conexão, cuja transmissão se baseia em pequenas unidades de tamanho fixo, denominadas células. Depois de muita discussão, foi definido um valor de 53 bytes para o tamanho da unidade, sendo 5 bytes para o cabeçalho e 48 para a informação. Embora o cabeçalho ocupe uma parte considerável da célula, chegou-se à conclusão de que células pequenas levam vantagens sobre outras maiores.

Assim como no X.25, o ATM possui dois tipos de conexão : a conexão por circuito virtual permanente (PVC - *Permanent Virtual Circuit*), que é estabelecida quando o equipamento se conecta à rede e permanece aberta indefinidamente, e a conexão por circuito virtual comutado (SVC - *Switched Virtual Circuit*), que, a cada vez que se faz necessária, passa pelas etapas de estabelecimento e depois de liberação da conexão.

Além das velocidades, o grande diferencial do ATM é o comprometimento com parâmetros de qualidade de serviço (QoS - *Quality of Service*), de maneira a acomodar adequadamente as exigências das várias categorias de tráfego na rede. De uma forma geral, as categorias são, de acordo com o ATM Forum :

- ◆ **Constant Bit Rate (CBR)** - Nesta categoria estão as conexões que necessitam de uma largura de banda fixa e contínua, baixíssimo atraso e

variação no tempo de propagação. São as aplicações isócronas, como videoconferência, vídeo-sob-demanda, voz e outras aplicações de áudio não comprimido;

- ◆ **Real-Time Variable Bit Rate (rt-VBR)** - Diz respeito a aplicações que também precisam de baixíssimo atraso e variação do tempo de propagação, mas em fontes que transmitem numa taxa variável. Aplicações típicas são as que utilizam áudio comprimido e com supressão de silêncio. Alguns tipos de comunicação multimídia também estão nesta categoria, incluindo também a videoconferência;
- ◆ **Non-Real-Time Variable Bit Rate (nrt-VBR)** - Nesta categoria enquadram-se as aplicações que têm um tráfego em rajadas (*bursty*) e não têm requerimentos tão restritos quanto ao atraso ou variação deste (um tempo máximo, no entanto, é definido). A aplicação mais comum neste caso é a transferência de dados num processamento *on-line* (por exemplo, aplicações bancárias);
- ◆ **Available Bit Rate (ABR)** - A categoria ABR foi criada para as fontes que têm a habilidade de aumentar ou reduzir a taxa de transmissão, se for necessário. Assim, pode-se aproveitar mais uma disponibilidade temporária da rede. Uma taxa máxima e uma taxa mínima são especificadas. Um exemplo de aplicação é a interconexão de LANs, que tipicamente usam pilhas de protocolos baseadas em roteamento como TCP/IP, e podem facilmente variar a taxa de emissão de pacotes;
- ◆ **Unspecified Bit Rate (UBR)** - Esta categoria baseia-se na idéia de “melhor esforço”, e não assume compromissos de qualidade de serviço. Desta forma, pode-se aproveitar qualquer capacidade eventualmente ociosa da rede. As aplicações nesta categoria geralmente não são *on-line*, como por exemplo a transferência de arquivos em *background*.

O modelo da arquitetura ATM pode ser visto na figura 2.1. Este modelo divide-se em planos e camadas. São 3 os planos existentes:

- ◆ *Plano do usuário* - Faz a transferência de informação dos usuários;

- ◆ *Plano de controle* - Realiza funções como a sinalização de ativação, manutenção e desativação de conexões;
- ◆ *Plano de gerenciamento* - Gerencia os planos (incluindo ele próprio) e as camadas (trata dos fluxos de informação de operação e manutenção relativos a cada camada).

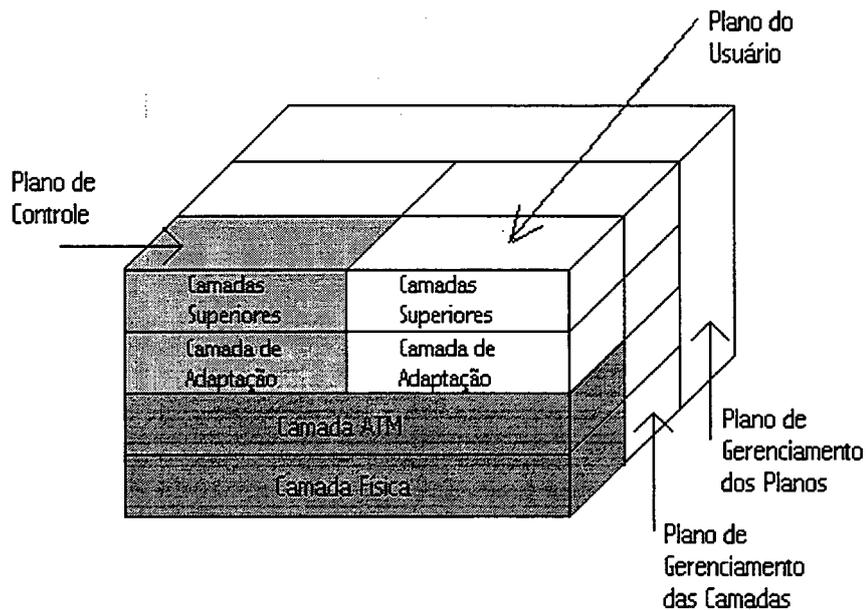


Figura 2.1 - Os planos no modelo de referência ATM

Já as camadas são as seguintes:

### ***Camada Física***

Assim como a camada física do modelo OSI, ela é responsável pela interação com o meio físico, incluindo alinhamento de bits, sinalização na linha e conversão eletro-ótica. Há também um controle de erros, baseado no campo HEC (*Header Error*

Control) do cabeçalho. Divide-se em duas subcamadas : *Physical Medium* (PM) e *Transmission Convergence* (TC).

### Camada ATM

Esta camada tem por função a multiplexação e demultiplexação de células e o encaminhamento/chaveamento, entre outros.

O formato da célula ATM não é único; depende da interface onde foi gerado. O ITU-T padronizou uma interface UNI (*User-to-Network Interface*) e uma NNI (*Network-to-Network Interface*) quando da criação da RDSI, para marcar os vários níveis de interconexão de equipamentos. A figura 2.2 mostra os dois formatos.

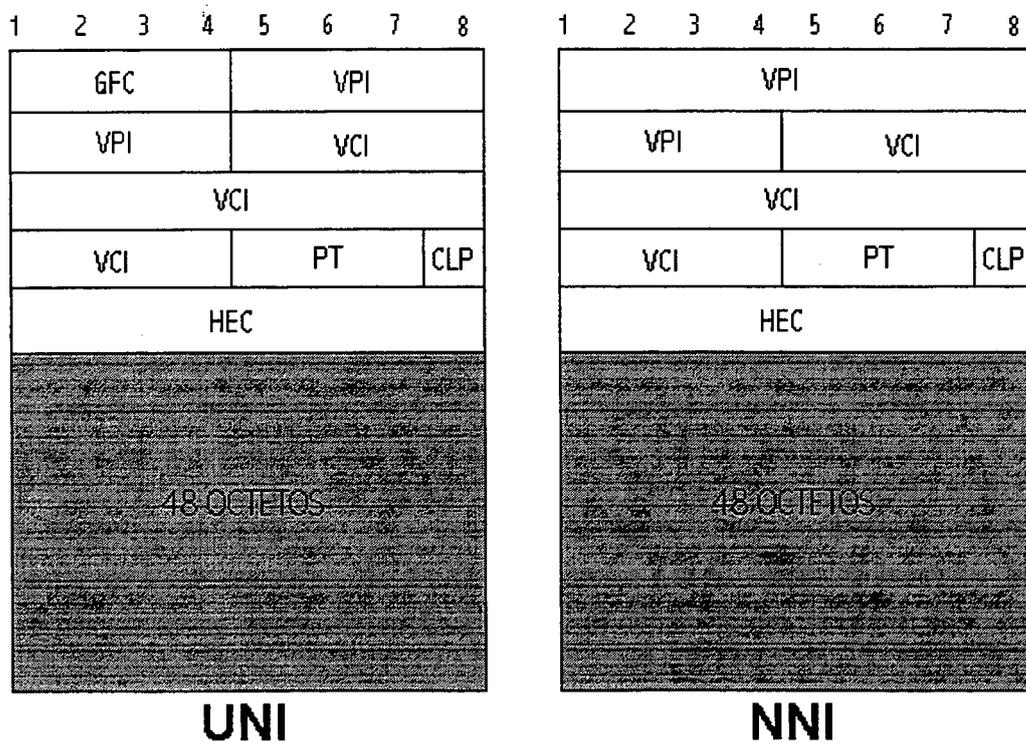


Figura 2.2 - Formato das células ATM

O campo GFC (*Generic Flow Control*) é definido para o mecanismo de controle de fluxo genérico.

Os campos VPI (*Virtual Path Identifier*) e VCI (*Virtual Channel Identifier*) associam a célula a uma conexão virtual, e são usados pelos comutadores para fazer o encaminhamento das células.

O campo PT (*Payload Type*) indica o tipo de informação contida na célula (de usuário ou manutenção), bem como se a célula passou por elementos congestionados durante o percurso.

O campo CLP (*Cell Loss Priority*) indica quais células serão descartadas primeiro, caso necessário.

Por último, o campo HEC, como já foi visto, é usado para detecção de erros.

### ***Camada de Adaptação***

A camada de adaptação (AAL – *ATM Adaptation Layer*) atua como um intermediário entre a camada ATM e os diversos tipos de serviço (CBR, rt-VBR, nrt-VBR, ABR e UBR) mostrados anteriormente, através de suas duas subcamadas : SAR (*Segmentation and Reassembly*) e CS (*Convergence Sublayer*), que fazem a montagem e segmentação da informação para as células e a multiplexação dos serviços, entre outras coisas.

Dependendo do tipo de serviço, usa-se uma diferente AAL. As definidas pelo ITU-T são:

- ◆ **AAL0** - Camada de adaptação nula; os serviços são repassados diretamente à camada ATM;
- ◆ **AAL1** - Adequada para serviços CBR orientados à conexão (ditos de classe A). Deve se preocupar com itens como a compensação da variação estatística do retardo e a recuperação do relógio da origem;

- ◆ **AAL2** - Adequada para serviços rt-VBR orientados à conexão (ditos de classe B). Deve se preocupar com itens como a compensação da variação estatística do retardo. Os procedimentos da AAL2 ainda não foram definidos;
- ◆ **AAL3/4** - Esta camada foi definida para os outros tipos de tráfego (ditos de classe C e D). Inicialmente, havia a AAL3 e a AAL4, mas quando se constatou que a diferença entre elas era pouco significativa, uniu-se as duas em uma só. Para preservar as peculiaridades de cada uma, o ITU-T propôs subdividir a subcamada CS em duas: a parte comum da subcamada de convergência (CPCS - *Common Part Convergence Sublayer*) e a subcamada de convergência específica de serviço (SSCS - *Service Specific Convergence Sublayer*);
- ◆ **AAL5** - A AAL5 foi criada para realizar as funções da AAL3/4 (embora não realize todas) de uma forma mais eficiente, já que a AAL3/4, com seu cabeçalho pesado, acaba por gerar muito *overhead*.

Na área de interconexão de redes locais, na grande maioria dos casos é utilizada a AAL5 como camada de adaptação, devido às características do tráfego.

O IETF (*Internet Engineering Task Force*) e o ATM Forum vêm definindo vários padrões para uma migração da infraestrutura de redes atuais (como IP e IPX) para ATM. Já surgiram definições como a RFC 1577 (*Classical IP Over ATM*) e o *LAN Emulation*.

A RFC 1577 define como usar o IP diretamente sobre ATM. O protocolo ARP foi modificado. Neste padrão, é criada a figura do Servidor ARP, de endereço conhecido por todas as estações da rede, cuja função é manter um mapeamento de endereços IP para endereços ATM de todas as máquinas na sua sub-rede lógica (i.e., máquinas com o mesmo endereço de rede). Assim, quando um computador deseja se comunicar com outro, obtém junto ao Servidor ARP o endereço do destino (todas as máquinas se registram ao se ligarem à rede) e estabelece uma conexão ATM diretamente com a máquina desejada. Esta abordagem do IETF, porém, possui 3 grandes problemas:

a) não permite a utilização de *broadcasts* e *multicasts*, muito comuns em redes locais;

b) vale só para o protocolo IP;

c) exige a presença de um roteador para a comunicação entre máquinas de diferentes sub-redes, mesmo que estejam ligadas na mesma rede ATM, e este roteador pode vir a ser um gargalo na rede.

Já o LAN Emulation definido pelo ATM Forum torna transparente a rede ATM ao protocolo da camada de rede, que pensa ter por baixo Ethernet ou Token-Ring. Neste padrão existem 4 personagens : o LEC (*LAN Emulation Client*), que são as máquinas puramente usuárias da rede, que se associam a uma rede lógica (*Elan*), o LES (*LAN Emulation Server*) que faz o registro e a resolução de endereços (MAC/ATM) da *Elan* onde ele se encontra, o BUS (*Broadcast and Unknown Server*), que controla todo o *broadcast* e *multicast* na rede, e o LECS (*LAN Emulation Configuration Server*), que entre outras coisas guarda os endereços dos LES de todas as *Elans* que ele conhece. Aqui, o processo de comunicação é o seguinte: um LEC, primeiramente, solicita ao LECS o endereço do LEC correspondente a sua *Elan* (embora possa-se optar por fornecer ao LEC, na sua configuração, diretamente o endereço do LES). Obtido o endereço, o LEC acessa o LES, registra-se e solicita o endereço do outro LEC, com o qual deseja-se comunicar, fornecendo para isso o seu endereço MAC. Quando há necessidade de utilização de *broadcast* ou *multicast*, contata-se o BUS (o endereço do BUS também é obtido junto ao LECS).

Mas se de um lado o *LAN Emulation* supre deficiências da RFC 1577, permitindo a existência de outros protocolos (como o IPX) e o uso de *broadcast*, mantém a necessidade do emprego de roteadores e adiciona um passo extra na transmissão, que é a tradução do endereço MAC (Ethernet ou Token-Ring) para endereço ATM.

Para resolver estes problemas, o ATM Forum está definindo o MPOA (*Multiprotocol Over ATM*) que, além de permitir diversos protocolos nas camadas superiores, utiliza como protocolo de roteamento o NHRP (*Next Hop Routing Protocol*), criado pelo IETF para dividir o trabalho de mapeamento de endereços de rede em endereços ATM por diversas máquinas conectadas à “nuvem” ATM (denominadas “roteadores virtuais”), evitando gargalos.

## 2.4 - ARQUITETURA DOS COMUTADORES ATM

Apesar do sucesso de uma rede ATM depender fundamentalmente dos equipamentos de comutação que provêem o serviço de transporte de informações, nenhum padrão foi definido para a arquitetura interna destes equipamentos. Assim, cada fabricante procurou resolver a sua maneira a questão de como construir o comutador que fosse mais rápido, mas também de menor custo e taxa de erros possível, gerando uma enorme variedade de escolha para o comprador. Esta seção tece algumas considerações sobre problemas inerentes ao projeto de um comutador, para dar uma pequena idéia da complexidade envolvida no assunto.

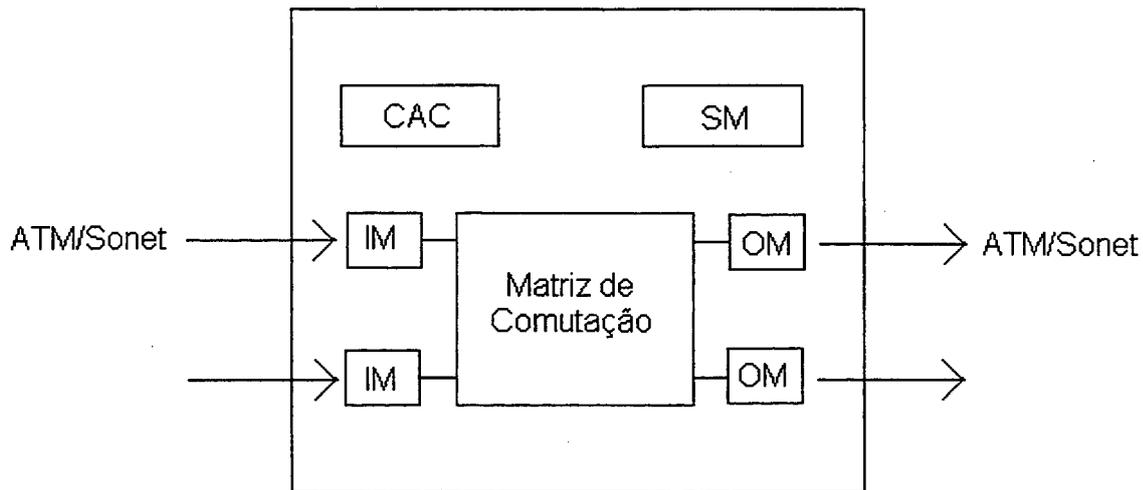
Um comutador ATM é muito mais do que uma peça de *hardware* que recebe uma célula numa de suas portas (dita de entrada, no caso) e a encaminha para outra(s) (dita(s), no caso, de saída). Na verdade, ele também realiza várias funções de controle e gerenciamento, além de atender a diferentes requerimentos de controle de tráfego. O conjunto de suas funções divide-se entre os três planos apresentados na seção 2.2 :

*Plano do usuário* – Neste plano encontra-se a função principal de um comutador, que é a de comutar células de usuário recebidas nas portas de entrada para a(s) porta(s) de saída adequada(s). Esta função pode ser dividida em três blocos : o módulo de entrada (junto às portas de entrada), a matriz de comutação (que faz o roteamento das células) e o módulo de saída (junto às portas de saída);

*Plano de controle* – Compreende as funções relacionadas ao estabelecimento e controle de conexões VP/VC (*Virtual Path/Virtual Circuit*). O comutador identifica células de sinalização, e até gera outras. O Controle de Admissão de Conexões (CAC) é responsável pela maioria de funções de sinalização;

*Plano de gerenciamento* – Neste plano encontram-se as funções de controle e monitoração da rede (gerência de falhas, configuração, desempenho, segurança, contabilização e tráfego). Estas funções podem ser representadas como sendo realizadas pelo bloco Gerenciamento do Comutador (SM – *Switch Management*), responsável pelos procedimentos de operação e manutenção (OAM).

Uma arquitetura genérica de um comutador com seus blocos funcionais pode ser vista na figura 2.3.



SM - Gerenciamento do Comutador  
CAC - Controle de Admissão de Conexão  
IM - Módulo de Entrada  
OM - Módulo de Saída

Figura 2.3 - Arquitetura genérica de um comutador ATM

Ainda que uma boa interação entre os blocos funcionais sejam de extrema importância, é na função principal do comutador que tem se concentrado o maior esforço da indústria no sentido de melhorar o desempenho e a eficiência do equipamento. As questões que mais têm desafiado os projetistas são:

1) *Como otimizar o roteamento das células dentro do comutador?*

Na entrada, uma etiqueta (*tag*) pode ser adicionada à célula, indicando por onde ela deve sair. Entre as abordagens usadas para resolver esta questão, tem-se :

**Memória compartilhada** – todas as células que chegam são colocadas numa memória comum, de onde são retiradas para a saída correspondente por um controlador;

**Meio compartilhado** – todas as células que chegam são colocadas sequencialmente num barramento comum. A este barramento estão ligadas também as saídas, através de filtros que liberam ou recusam a passagem de uma célula;

**Interconexão total** – todas as entradas têm conexão direta com cada uma das saídas. Assim como na abordagem de meio compartilhado, aqui também existe a figura dos filtros, deixando passar apenas as células direcionadas a aquela saída em particular;

**Divisão de espaço** – nesta abordagem, entre a entrada e a saída, a célula passa por múltiplos estágios, como no caso de uma rede Banyan.

## 2) Como otimizar o armazenamento das células no comutador?

Como células de mais de uma entrada podem ser destinadas à mesma saída, *buffers* são sempre necessários, caso contrário muitas células seriam perdidas. Estes *buffers* podem estar:

**Na entrada** – quando células de mesmo destino chegam ao mesmo tempo, apenas uma segue, enquanto as outras ficam esperando na entrada. O grande problema desta abordagem é que uma célula que aguarda a liberação da sua porta de saída pode trancar outras células direcionadas para outras saídas que estão livres. Isto pode degradar bastante o desempenho, por isso em geral não se usa este tipo de armazenamento;

**Na saída** - células com o mesmo destino entram em uma fila na própria porta de saída. Esta é atualmente a abordagem mais comum entre os projetistas;

**Internamente** – num comutador de múltiplos estágios, cada elemento intermediário contém um *buffer*. É uma abordagem complexa em termos de implementação.

Existem ainda os “*buffers* de recirculação”, onde na chegada de células conflitantes (i.e., com o mesmo destino), uma segue e outra é redirecionada para a entrada.

Escolher a abordagem de roteamento tem uma série de implicações, principalmente quanto à escalabilidade (ou seja, quanto o comutador pode crescer em termos de número de interfaces), envio de mensagens de *multicast* (uma mensagem de entrada direcionada simultaneamente a várias saídas) e tolerância a falhas. Em cada uma delas há pontos fortes e pontos fracos.

Além disso, no armazenamento também deve ser levado em conta o tipo de tráfego: alguns são sensíveis ao tempo, outros não. Como dar prioridade ao primeiro? Alguns comutadores implementam, em cada saída, um *buffer* para cada tipo de tráfego.

Portanto, na escolha de um equipamento há um sem-número de fatores a serem considerados. “Como é a matriz de comutação?” “Onde estão os *buffers* e qual a sua capacidade?” “Como é feito o descarte de células em caso de congestionamento?” “Qual é o tempo de latência (tempo que a célula leva para atravessar o comutador)?” São apenas algumas das perguntas que devem ser respondidas.

# CAPÍTULO III - ESTADO DA ARTE

## 3.1 - Introdução

As potencialidades da tecnologia do ATM têm atraído muita atenção de pesquisadores, tanto ligados ao mundo comercial quanto ao acadêmico. Tal fato faz com que exista um profusão de publicações relacionadas ao tema. Com intenção de se prender ao objetivo do trabalho, buscou-se na literatura apenas artigos que diziam respeito à arquitetura de comutadores ou à modelagem e testes de avaliação de desempenho.

## 3.2 - Arquitetura de comutadores

Como não há nenhum padrão definido para a implementação do funcionamento interno de um comutador, e naturalmente não há interesse por parte dos fabricantes em divulgar detalhes da tecnologia empregada na fabricação de seus equipamentos. Sobre este tópico existe pouco material bibliográfico disponível, a não ser aqueles em que o assunto é tratado de forma bastante superficial. Mesmo assim, algumas publicações encontradas contribuíram em muito para a criação dos modelos, descritos no capítulo IV.

Em primeiro lugar, um *white paper* da Xylan levanta alguns pontos importantes a serem considerados na escolha de um comutador ATM [XYL97]. Entre eles, pode-se citar:

- ♦ **Integração entre tecnologia LAN e ATM** – a empresa argumenta que, para tirar-se vantagem da comutação em alta velocidade característica do ATM, o ideal é que o comutador tenha o seu *backplane* implementado usando esta tecnologia, mesmo que a função principal do comutador seja a interligação de redes Ethernet e/ou Token-Ring, e a ligação a uma rede ATM seja através de um *uplink* (interface com velocidade maior do que as outras no comutador);

- ◆ **Matriz não bloqueante** – uma matriz de comutação é dita não bloqueante quando tem seus *buffers* na saída. *Buffers* na entrada podem causar o que se chama de bloqueio de início de fila (*head-of-line blocking*), isto é, a primeira célula da fila fica presa por não ter espaço na saída para comportá-la, e portanto atrasa também as outras células da mesma fila;
- ◆ **Descarte inteligente de células** – o descarte aleatório de células causa o descarte de vários pacotes de camadas superiores no destino; mais inteligente é o descarte de células do mesmo pacote. Para isso, foram desenvolvidos métodos como o EPD e PPD (*Early Packet Discard* e *Partial Packet Discard*).

Outras duas fontes foram de extrema importância para a modelagem. Um artigo escrito na Universidade de Ohio [FAH97] descreve com riqueza de detalhes as arquiteturas das matrizes de comutação mais conhecidas atualmente, levantando para cada uma suas vantagens e desvantagens. Além disso, fala das funções de controle e gerenciamento, discorre sobre possibilidades de alocação de *buffers* e técnicas de implementação de *broadcast* e *multicast* e controle de tráfego.

No Brasil, há que se destacar o trabalho desenvolvido no laboratório Telemídia da PUC-RJ [SIL97], que fez um levantamento bastante amplo dos comutadores encontrados no mercado nacional, caracterizando-os em termos de interfaces suportadas, arquitetura da matriz, sinalização, serviços e tráfegos suportados. Além disso, também descreve os processos realizados por um comutador genérico.

### **3.3 - Modelagem e testes de avaliação de desempenho**

Em relação à modelagem e a testes de avaliação de desempenho já se dispõe de uma bibliografia mais ampla. As fontes usadas como informação podem ser divididas em quatro tipos :

- ◆ Instituições de padronização;
- ◆ Publicações técnicas dos fabricantes;
- ◆ Artigos de periódicos;
- ◆ Artigos acadêmicos.

Este tópicos pode ter várias abordagens de acordo com o tipo de fonte, tornando-se, portanto, conveniente classificar as publicações para um melhor entendimento.

### 3.3.1 - Instituições de padronização

Entre as instituições que se preocupam em definir padrões para a tecnologia ATM, destacam-se o ATM Forum e o IETF.

O ATM Forum é uma associação de fabricantes de equipamentos, mas que hoje tem também como membros outros tipos de empresas e até usuários. O objetivo desta associação é definir padrões provisórios para interoperabilidade, para que os fabricantes tenham uma diretriz até que um padrão definitivo seja ratificado pelo ITU-T. Entretanto, na prática, muitas das suas especificações acabam por se tornar definitivas.

O ATM Forum não interfere na maneira pela qual os fabricantes constroem seus comutadores, mas produziu uma pequena introdução [ATM94a] às áreas de teste de implementações ATM. São elas:

- ◆ **Conformidade** – verifica se a implementação está de acordo com a implementação do protocolo;
- ◆ **Desempenho** – verifica como se comporta a implementação sob diversas condições de tráfego em termos de desempenho;
- ◆ **Interoperabilidade** – como se comporta a implementação em relação a outras implementações.

O documento não entra em detalhes de cada área, e remete ao uso do PICS (*Protocol Implementation Conformance Statements*), baseado na especificação do protocolo, e do PIXIT (*Protocol Implementation eXtra Information for Testing*) que contém informações extras que podem não estar contidas na especificação do protocolo. Ambos são questionários contendo as questões consideradas mais relevantes, mas não oferecem parâmetros de referência para se dizer se o resultado de um teste é bom ou ruim. No caso de testes de desempenho, o documento os divide em duas categorias :

- ◆ Teste em situação normal – quando a rede se encontra sob condições normais de uso;



- ◆ Teste em situação de sobrecarga – quando a rede se encontra sob uma carga excessiva de tráfego.

Dentre as especificações mais importantes do ATM Forum, destacamos a versão 4.0 do protocolo de Gerenciamento de Tráfego [ATM96a], que objetiva o controle de tráfego para o uso eficiente da rede. Mesmo assim, ele não tem em vista parâmetros de desempenho de um comutador em si, mas da rede como um todo. Esta especificação, no entanto, serviu como referência na determinação da proporção de dados de controle usada nos modelos, pois além da definição das classes de tráfego (CBR, VBR-rt, VBR-nrt, ABR e UBR), lista funções que devem ser feitas pelos comutadores, como por exemplo :

- ◆ **Controle de Admissão de Conexão (CAC)** – responsável pela aceitação ou não do estabelecimento de uma nova conexão. A decisão irá se fundamentar na disponibilidade do comutador e da rede em atender à qualidade de serviço solicitada, que prevê a taxa de transmissão, retardo máximo, entre outros itens;
- ◆ **Controle dos Parâmetros de Uso (UPC)** – controla o tráfego gerado pelo usuário final, para que ele esteja adequado ao contrato de tráfego firmado no estabelecimento da conexão. Um envio de células a uma taxa acima da combinada, por exemplo, pode fazer com que elas sejam imediatamente descartadas;
- ◆ **Gerenciamento dos Recursos de Rede** – responsável pelo controle dos circuitos, congestionamentos, etc.

Já o IETF é responsável, na verdade, por padrões relacionados à Internet, publicados através de RFCs (*Requests for Comments*). Considerando a crescente interação entre ATM e a Internet, seja como *backbone* central nos EUA, seja nas redes dos usuários, vários padrões de uso dos protocolos TCP/IP sobre ATM já foram criados. Referente ao tema desempenho, pode-se citar três RFCs como relevantes:

**RFC1242** – Este RFC [BRA91] define uma terminologia para *benchmarking* (comparação entre equipamentos usando uma medida de referência) de equipamentos de interconexão. O documento trata mais de roteadores, já que à época em que ele foi escrito pouco se falava de comutadores. Ele define termos como latência (intervalo de

tempo entre o fim do primeiro bit na porta de entrada e o início do primeiro bit na porta de saída) e carga constante (quadros constantes a intervalos fixos de tempo).

**RFC não numerado (*Internet Draft*)** – Este RFC [CRA97] estende o RFC 1242 para abarcar a área de tecnologia de comutação de células. Entre os termos definidos, tem-se:

- ◆ Tempo de estabelecimento de conexão – tempo necessário para o circuito virtual ser estabelecido;
- ◆ Tempo de segmentação e remontagem de quadros – tempo necessário para se quebrar um quadro em células, ou se juntar células em um quadro;
- ◆ Throughput – taxa máxima pela qual nenhum dos quadros repassados ao comutador para envio são descartados;
- ◆ Gerenciamento de congestionamento – eficiência de medidas tomadas pelo comutador para lidar com o congestionamento.

**RFC1944** – Este RFC [BRA96] traz uma metodologia para avaliação de desempenho de equipamentos de interconexão, recomendando os seguintes testes (os termos apresentados seguem a definição dos RFCs citados anteriormente):

- ◆ Throughput – tem por objetivo determinar o *throughput* do equipamento;
- ◆ Latência – tem por objetivo determinar o valor da latência no equipamento;
- ◆ Taxa de perda – procura determinar a taxa de perda de quadros como a razão de quadros perdidos sobre quadros recebidos pelo equipamento para encaminhamento;
- ◆ Recuperação do sistema – objetiva caracterizar a velocidade pela qual o equipamento se recupera de uma condição de sobrecarga.

### 3.3.2 - Fabricantes

Os fabricantes, obviamente, dedicam-se mais a provar que os seus produtos são os melhores do mercado. Apesar disso, eles também fornecem alguns dados que, se não são inteiramente confiáveis, ao menos dão uma noção dos valores esperados em termos de latência, capacidade de *buffers* e características que um comutador deve ter.

Por exemplo, a Fore Systems fez um comparativo entre seu comutador ASX-1000 e os de alguns concorrentes, como CISCO, 3COM, Bay e Xylan [FOR97a]. O *white paper* não explica em que circunstâncias foram feitas as medições, mas alguns resultados apresentados foram úteis na criação dos modelos, como :

- ◆ Latência- os valores observados ficaram entre 11  $\mu$ s (comutador da 3Com) e 62  $\mu$ s (comutador da Bay Networks);
- ◆ Capacidade e localização dos buffers – as capacidades variaram de 32000 a 1024000 células, e em geral a localização dos *buffers* é na saída da matriz de comutação;
- ◆ Velocidade do barramento – os valores encontrados ( de 2,48 a 9,95 Gbps) não levaram em consideração o tipo de arquitetura da matriz, apenas o *backplane*.

### 3.3.3 - Periódicos

Algumas revistas também têm feito testes de desempenho com comutadores. No entanto, são testes para avaliar toda a interação entre o(s) comutador(es), as máquinas de usuários finais e o meio físico em uma situação bem específica.

A revista **Network Computing**, por exemplo, fez uma avaliação de comutadores de grande porte, usados primordialmente para tráfego de voz e/ou Frame Relay [NETC97]. Os realizadores do teste solicitaram que cada fabricante participante (IBM, GDC e Nortel) criasse uma rede com dois comutadores, para que fossem feitas medições, primeiro com uma configuração padrão, e depois com a configuração que mais otimizasse o desempenho do equipamento. Duas baterias de testes foram feitas : uma com transmissão usando Frame Relay sobre ATM, com dados gerados e contabilizados por um analisador de tráfego, e a outra com transmissão de voz sobre ATM, a partir da interligação de dois equipamentos de PBX. De forma geral, o comutador da GDC (General DataComm) saiu-se melhor.

Uma das conclusões importantes deste artigo é que o desempenho das implementações de características proprietárias superam em muito o desempenho de implementações padronizadas. Tal fato é fácil de se compreender, visto que os padrões muitas vezes sacrificam o desempenho em nome da interoperabilidade.

### 3.3.4 - Artigos acadêmicos

Uma grande parte dos artigos publicados têm discutido a questão do tráfego IP e TCP sobre ATM. As perguntas mais levantadas dizem respeito à relação entre TCP/IP e a classe de tráfego usada para o transporte (ABR ou UBR), tentando descobrir até que ponto deve-se preocupar com a segurança da entrega em detrimento do desempenho e do uso mais eficiente da rede. A resposta determina o uso de ABR (que tem garantido uma taxa mínima de transmissão e um controle de fluxo a nível da camada ATM) ou de UBR (não tem nenhuma garantia, mas é mais simples e tem sua eficiência aumentada quando se usa políticas de descarte).

Estudos feitos através de simulação [KAL96] [KAL97a] mostram que o desempenho de TCP sobre ABR é dependente da capacidade dos *buffers* do comutador, exibindo um *throughput* significativamente maior quando o comutador consegue comportar o tráfego de todas as fontes. Como há controle de fluxo na camada ATM, no entanto, à medida em que o tráfego aumenta, as fontes diminuem sua taxa de envio, e não há tanta perda de pacotes. Com UBR também há esta dependência, mas como nesta classe não se faz um controle de fluxo em nível ATM, as fontes mantêm sua taxa de transmissão, e muitos pacotes são perdidos. Segundo cálculos feitos, o uso de UBR exigiria dos comutadores capacidade dos *buffers* igual à soma das taxas de todas as fontes [KAL97] e [GOY97], o que torna o UBR não escalável em termos de fontes. Por tudo isso, o artigo conclui que o ABR joga o congestionamento para os pontos de entrada da rede, onde ele tem menos conseqüências, enquanto que o UBR mantém o congestionamento nos comutadores, onde ele é mais nocivo.

Discute-se, ainda, sobre como fazer um controle de congestionamento, para tentar evitar-se um descarte no comutador, o que tem um grande impacto na latência. Duas abordagens monopolizam o tema [JAI96] :

- ◆ Controle baseado em taxa – estabelece que a fonte começa a mandar na velocidade que julga adequada, e muda depois de acordo com o *feedback* recebido da rede. Logo, pode-se dizer que este é um controle feito fim-a-fim.
- ◆ Controle baseado em crédito – ao contrário do primeiro, este controle é feito por todos os equipamentos no caminho da conexão. Cada nó mantém uma fila

para cada circuito virtual (VC), e determina o número de células que pode receber naquele VC. Este número é conhecido como crédito.

Depois de muitas discussões, a primeira abordagem foi escolhida pelo ATM Forum.

Voltando-se à linha de avaliação de desempenho, tem-se [KAR97], onde o autor tem como objetivo a criação de uma estrutura para testes do comportamento do protocolo TCP, aproveitando para propor uma taxionomia potencialmente útil para outras avaliações em ambiente ATM. O artigo apresenta alguns cenários de teste (número de ligações consideradas como gargalos, conexões em um ou nos dois sentidos, uso de LAN ou WAN, e o ATM como *backbone*) e propõe definições para algumas métricas, como :

- ◆ Throughput – número de bits enviados por unidade de tempo;
- ◆ Eficiência – soma dos *throughputs* de todas as fontes sobre o *throughput* máximo possível;
- ◆ Justiça – mede se todas as conexões estão tendo a mesma chance de transmitir;
- ◆ CLR – taxa de perda de células – número de células perdidas sobre o número de células enviadas.

O NIST (*National Institute of Standards and Technology*) também tem feito testes de desempenho com TCP/IP sobre ATM [FOU95]. Para comprovar algumas crenças comuns atualmente, pesquisadores do NIST ligaram duas estações Sun com placas ATM da Fore Systems (100 Mbps), e usaram o programa de domínio público *Netperf*, disponível na Internet, para fazer uma medição do *throughput* entre as estações. Os resultados obtidos mostraram que as aplicações foram o fator limitante do *throughput* (isto é, não conseguiu-se gerar tráfego suficiente para ocupar toda a banda da rede), e que apesar do grande *overhead* gerado pelo TCP/IP, seu uso se justifica em razão da segurança e confiabilidade na comunicação.

Além destes testes, o NIST desenvolveu também um simulador próprio, com interface baseada no padrão *X-Windows*. É um simulador simples, que permite a montagem de um modelo que tem como componentes os comutadores, as máquinas de

usuários finais, ligadas aos comutadores, os programas geradores e receptores de tráfego e o meio físico de interligação. Para cada componente fornece-se algumas informações, como velocidade do meio físico, classe e taxa de geração de tráfego, etc. É um programa bastante interessante, flexível e extensível, mas não foi utilizado porque é bastante voltado à simulação da rede como um todo, não de um comutador único.

Por fim, não pode-se deixar de citar o já mencionado levantamento do laboratório Telemidia da PUC-RJ [SIL97], que faz uma comparação entre o desempenho dos comutadores pesquisados. Como o trabalho apresentado nesta dissertação, ele se concentra no comutador, e não na rede. A diferença é que a comparação foi feita por medição, e não por simulação. Para isso, equipamentos de geração e análise de tráfego foram ligados a portas dos comutadores e, para reproduzir situações de cargas reais, cabos de *loopback* ligaram portas entre si, para que o tráfego que saísse por uma entrasse por outra. Todas as portas tinham interface OC-3 a 155 Mbps, que são as mais comuns no mercado.

Inicialmente, era gerada uma carga de 90% da capacidade máxima da porta. As cargas eram, então, gradativamente diminuídas até o momento em que o comutador não mais descartasse células. Neste ponto, foram medidos :

- ◆ Throughput – o *throughput* foi medido sempre com uso de tráfego CBR. Constatou-se que os comutadores conseguiam chegar perto do máximo *throughput* possível na interface;
- ◆ Latência – a latência apresentou pouca variação entre as medições feitas com pouca (30% do máximo) e muita (90% do máximo) carga, mantendo-se em torno dos 18  $\mu$ s;
- ◆ Jitter – o *jitter* corresponde à variação do retardo de transmissão. É uma medida muito importante quando o tráfego é sensível ao retardo, como no caso de CBR. Assim como a latência, o *jitter* também mostrou pouca variação, em torno de 2  $\mu$ s;
- ◆ Armazenamento de células – foi medido também qual o número de células que os comutadores tinham capacidade de armazenar quando recebesse uma rajada de células. Os resultados mostram valores entre 1100 e 24100 células.

Como será visto no capítulo 5, estes resultados são consistentes com os obtidos neste trabalho, tomando-se por base as circunstâncias em que não há descarte.

# CAPÍTULO IV - METODOLOGIA DE PESQUISA

## 4.1 – Introdução

No momento em que se define uma metodologia de avaliação de desempenho, deve-se tomar muito cuidado para não incorrer-se em erros comuns, como:

- ◆ falta de objetivos;
- ◆ objetivos tendenciosos;
- ◆ técnica errada de avaliação;
- ◆ detalhamento demasiado ou inexistente, entre outros.

Para evitar tais erros, o ideal é adotar-se uma abordagem sistemática como a proposta em [JAI91], utilizada neste trabalho.

Esta metodologia é apresentada na seção 4.2. A seção 4.3 mostra como a metodologia foi aplicada. Note-se que, com o intuito de tornar a exposição do trabalho realizado mais clara, há uma diferença entre a ordem dos passos na metodologia original e a mostrada nesta seção.

## 4.2 - Definição da metodologia

A metodologia proposta por Raj Jain é composta dos seguintes passos :

### 1. *Definição dos objetivos e do sistema*

Os objetivos podem ser a princípio difíceis de se precisar, mas é o primeiro passo para a resolução do problema. Já o sistema deve ter suas “fronteiras” bem claras, pois isto afeta as métricas de desempenho, bem como as cargas usadas para a comparação.

## 2. *Elaboração da lista de serviços e resultados esperados*

Cada sistema provê um conjunto de serviços (a transferência de dados, no caso das redes, por exemplo), e para cada um deles existe um conjunto de possíveis resultados, desejados ou não. Isso será importante para a escolha das métricas e cargas.

## 3. *Seleção das métricas*

Métricas são critérios para a comparação do desempenho. De uma forma geral, as associamos com os três tipos de resultados possíveis de uma solicitação de serviço:

- ◆ Solicitação atendida corretamente – Neste grupo inclui-se as métricas relacionadas à velocidade;
- ◆ Solicitação atendida incorretamente – Neste grupo inclui-se as métricas referentes à confiabilidade do sistema;
- ◆ Solicitação não atendida – Neste grupo inclui-se as métricas ligadas à disponibilidade do sistema.

Para a escolha de boas métricas, recomenda-se verificar se elas possuem:

- ◆ **Baixa variação**, para diminuir o número de experimentos necessários;
- ◆ **Não redundância**, para evitar números desnecessários;
- ◆ **Compleitude**, para refletir todos os resultados possíveis.

As métricas mais comuns são: tempo de resposta, *throughput*, eficiência (capacidade utilizável sobre capacidade nominal), utilização (tempo ocupado sobre tempo total decorrido), confiabilidade (tempo sem erros), disponibilidade (MTBF, *Mean Time Between Failures*, tempo médio entre falhas) e a relação custo/desempenho.

## 4. *Elaboração da lista de parâmetros*

No caso, os parâmetros que afetam o desempenho. A lista pode ser dividida em dois tipos: parâmetros de sistema (que geralmente não variam de uma instância de sistema para outra) e de carga (que são características das solicitações dos usuários e, portanto, bastante variáveis). A lista deve ser o mais abrangente possível.

### ***5. Seleção dos fatores para estudo***

Fatores são parâmetros que sofrerão variação durante o estudo. Os valores que eles podem assumir são chamados de níveis. Para facilitar, é melhor começar com poucos fatores e poucos níveis em cada um, e ir aumentando a lista conforme a necessidade. Para boa escolha de fatores, deve-se usar os parâmetros que mais influenciam no desempenho.

### ***6. Seleção da técnica de avaliação***

Em termos de mais alto nível, as três técnicas existentes são a simulação, a modelagem analítica e a medição. Qual delas usar depende do tempo e recursos disponíveis para a resolução do problema e nível de acuidade desejado. O ideal é que, independentemente da técnica escolhida, use-se as outras para corroborar o resultado.

### ***7. Seleção da carga***

A carga consiste de uma lista de solicitações de serviço ao sistema, e pode variar de acordo com a técnica escolhida. O importante é que ela reflita o uso real do sistema. A seleção da carga deve levar em consideração que :

- ◆ Deve “exercitar” todos os serviços que importam ao estudo, ou seja, todos os serviços que afetam o desempenho devem ser solicitados;
- ◆ O nível de detalhe deve refletir a realidade, ou seja, se o sistema recebe uma grande variedade de solicitações, o uso de apenas uma como carga não é representativa;
- ◆ Parâmetros como taxa de chegada de solicitações, uso de recursos, seqüência e quantidade de uso devem também estar o mais próximos possível do uso real;
- ◆ A carga deve representar a utilização atual do sistema.

### ***8. Planejamento dos experimentos***

De posse da lista de fatores e seus níveis, deve-se decidir uma seqüência de experimentos de modo a obter o máximo de informações com o mínimo de esforço. Na prática, recomenda-se dividir os experimentos em duas fases : na primeira, o número de

fatores pode ser alto, mas o número de níveis é baixo. Na segunda, o número de fatores é reduzido e o número de níveis daqueles que são significativos é aumentado.

### ***9. Análise e interpretação dos dados***

Nesta etapa deve-se usar técnicas estatísticas adequadas para fazer uma consolidação dos resultados obtidos, de modo que se possa tirar conclusões sobre o desempenho do sistema.

### ***10. Apresentação dos resultados***

Muita atenção deve ser dada à apresentação final da avaliação, principalmente se o público-alvo é um público não técnico.

## **4.3 – Aplicação da metodologia**

A aplicação desta metodologia foi bastante importante para que o trabalho fosse conduzido de forma organizada. Primeiramente, foram estabelecidos os objetivos, o escopo do sistema, os serviços oferecidos e a técnica de avaliação, mostrados a seguir.

### ***Objetivos :***

- ◆ Comparação entre desempenhos de comutadores genéricos sob uma determinada configuração;
- ◆ Determinação dos fatores mais relevantes no que se refere ao desempenho destes equipamentos.

### ***Sistema :***

O sistema corresponde a um comutador ATM, que interage com o meio através do recebimento de células ou quadros por uma das interfaces, que são comutados e devolvidos por outra.

### ***Serviço :***

Transferência de dados, sejam eles textuais, gráficos, voz ou vídeo.

### ***Técnica de avaliação :***

Inicialmente, optou-se pela técnica da medição, com a intenção de se investigar o comportamento de comutadores oferecidos no mercado, e assim obter resultados que fossem calcados em valores reais. Entretanto, para que isso fosse possível, fazia-se necessário um ambiente de teste que representasse com fidelidade situações normais de uso dos comutadores. Como tal ambiente não pôde ser disponibilizado, tanto por restrições técnicas como financeiras, concluiu-se que a simulação ofereceria melhores resultados.

A tarefa foi, então, dividida em 5 etapas :

- ◆ *Etapa 1* – Projeto e desenvolvimento dos modelos;
- ◆ *Etapa 2* – Especificação das métricas (correspondente ao passo 3);
- ◆ *Etapa 3* – Definição dos parâmetros, fatores e carga (correspondente aos passos 4, 5 e 7);
- ◆ *Etapa 4* – Planejamento e realização dos experimentos (correspondente ao passo 8);
- ◆ *Etapa 5* – Análise estatística dos resultados obtidos (correspondente ao passo 9).

As subseções 4.3.1 a 4.3.4 descrevem as quatro primeiras etapas, enquanto que a quinta será abordada no próximo capítulo.

#### **4.3.1 - Projeto e desenvolvimento dos modelos**

A simulação de um sistema pressupõe a existência de um modelo computacional que imite o seu funcionamento da forma mais fiel possível. Na literatura, boas descrições de um comutador ATM foram encontradas em [SIL97] e em [FAH97]. O primeiro trabalho faz um levantamento das características de comutadores de vários fabricantes, e faz uma breve apresentação dos processos associados ao seu funcionamento. Já o segundo faz um detalhamento das arquiteturas das matrizes de comutação.

Baseado nestes trabalhos e em especificações técnicas de equipamentos disponíveis comercialmente, os modelos desenvolvidos foram subdivididos em 3 submodelos, cada um deles correspondendo a fases ou conjunto de processos que células ou quadros devem realizar, e que, conforme a arquitetura do comutador, os diferencia uns dos outros. O fluxo geral da célula ou quadro dentro de um comutador, com destaque para os submodelos, pode ser visto na figura 4.1.

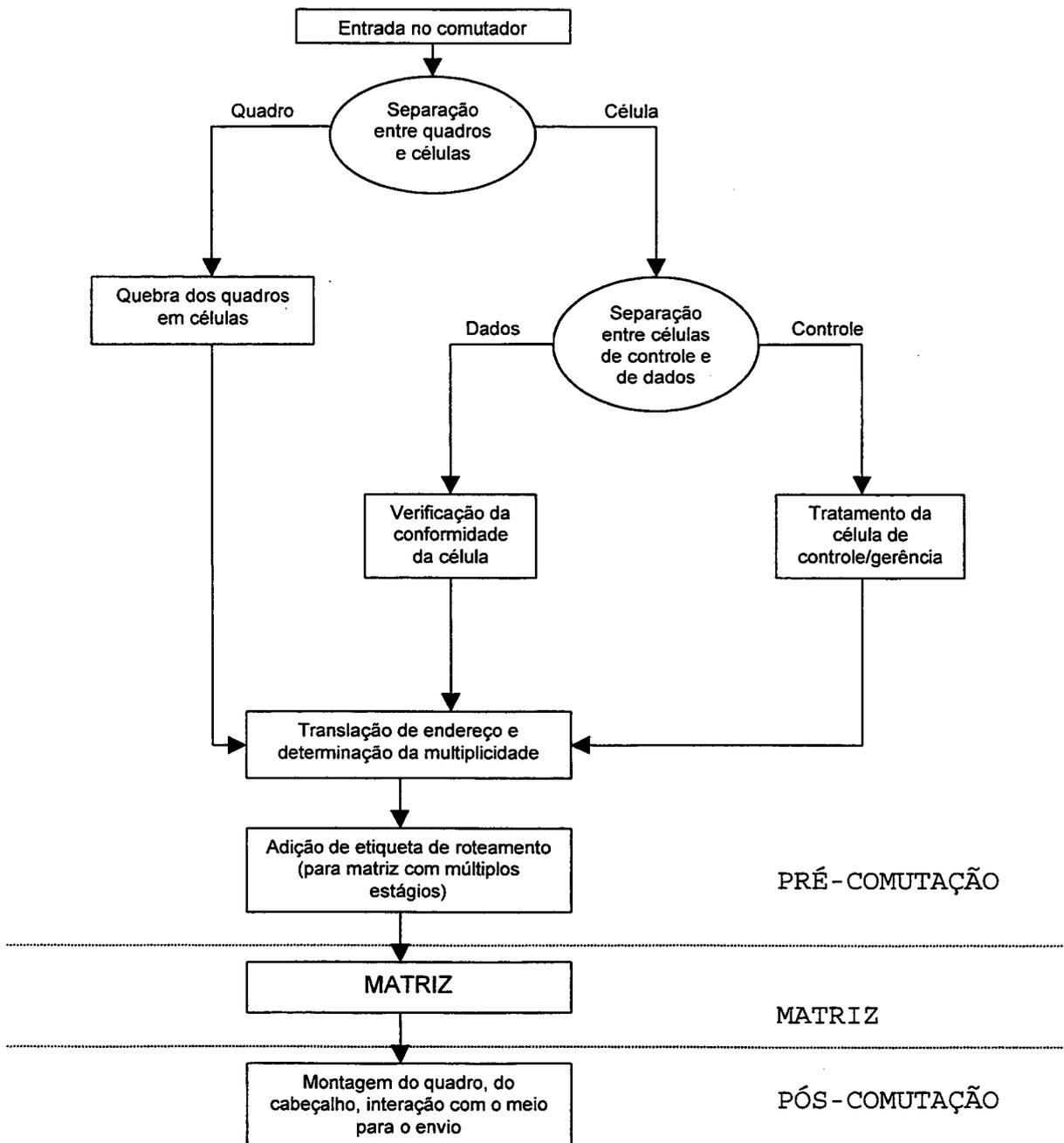


Figura 4.1 – Fluxo geral da dentro do comutador

Pré-comutação

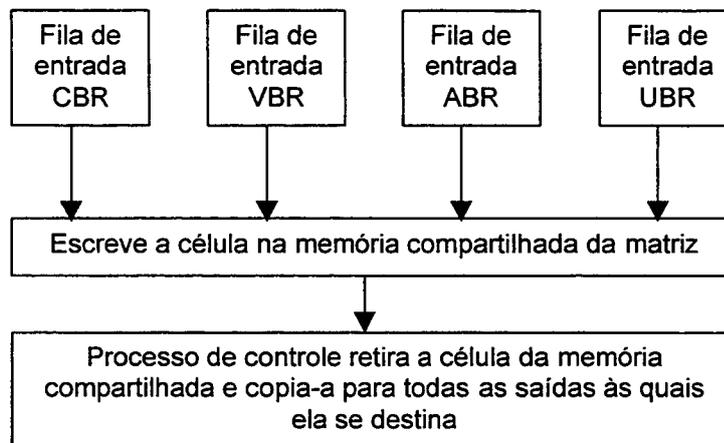
Neste segmento é feita uma representação dos processos realizados antes de ser feita a comutação propriamente dita. Nos modelos estão presentes os mais relevantes, que são :

- ◆ *A quebra de um quadro em células*, no caso de comutadores que aceitam placas com interfaces não ATM, como Ethernet, Token Ring e Frame Relay. Esta quebra é necessária, já que toda a comutação interna é feita através de células;
  
- ◆ *O tratamento de células de controle*, ou seja, células que não transportam dados dos usuários finais, mas sim informações relativas às funções de controle (como as de estabelecimento de conexões) ou de gerenciamento, de acordo com padrões definidos;
  
- ◆ *A verificação da conformidade do fluxo de entrada com o contrato de tráfego pré-estabelecido* (assumindo-se que a célula já faz parte de uma conexão ATM). O caso mais comum de não conformidade é o envio a uma taxa maior do que a combinada. Neste tipo de situação várias medidas podem ser tomadas, como o descarte imediato da célula, ou a sua marcação (mudando-se o bit de *Cell Loss Priority* (CLP) para um), tornando-a mais suscetível a um descarte, caso isso se faça necessário. Este trabalho adota a segunda estratégia;
  
- ◆ *A translação de endereços*, o que em ATM corresponde ao par VPI/VCI. Cada par VPI/VCI identifica uma conexão, e é alterado entre a entrada e a saída de cada comutador. Portanto, ao receber uma célula com um certo VPI/VCI, o comutador deve determinar para qual saída ela deve ser encaminhada, indicando este fato à matriz de comutação através da substituição do par VPI/VCI. Este processo é também responsável pela

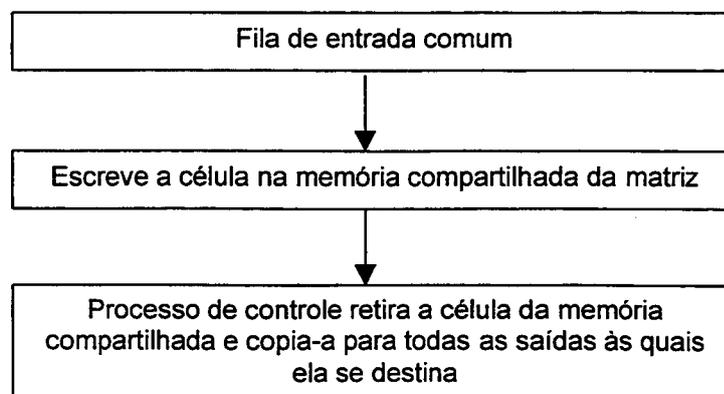
verificação da multiplicidade das células, isto é, se as células serão repassadas para uma, várias ou todas as conexões de saída.

### Comutação na matriz

Neste segmento é feita uma representação da matriz de comutação, recebendo-se a célula em uma de suas entradas e copiando-a para a(s) saída(s) correspondente(s). Note-se que a matriz pode tratar todas as células igualmente, ou pode fazer uso de prioridades, beneficiando as classes de tráfego mais sensíveis à latência. As duas abordagens são contempladas nos modelos. As figuras 4.2, 4.3 e 4.4 apresentam os três tipos de matrizes usados na modelagem.

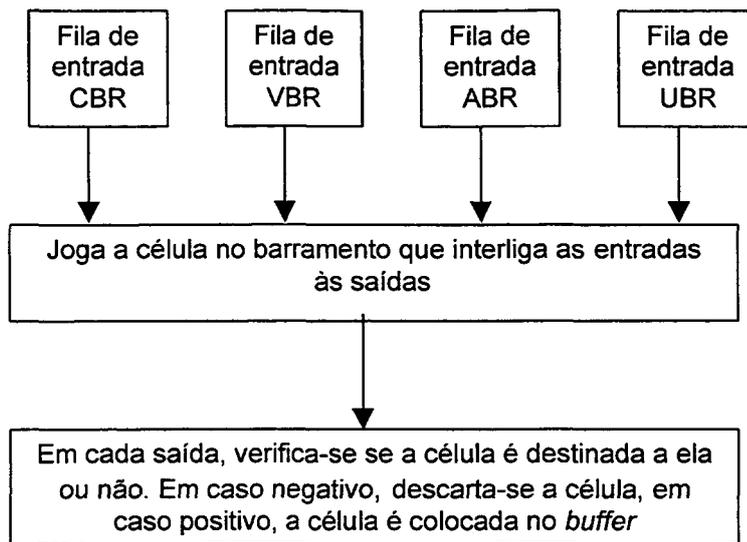


(a)

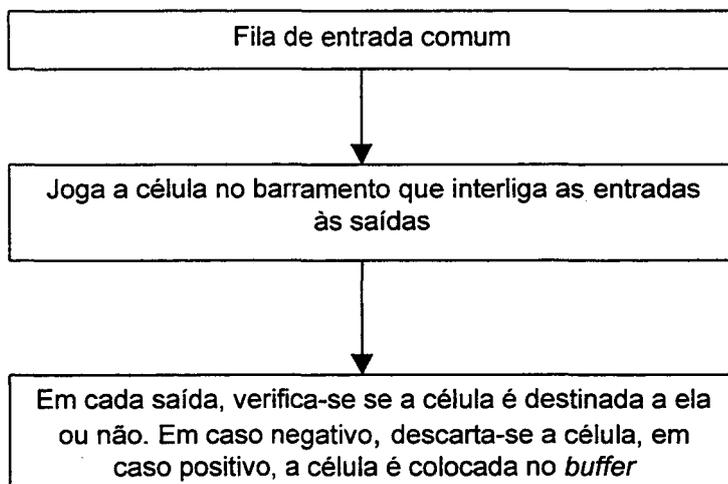


(b)

Figura 4.2 – Matriz com arquitetura de memória compartilhada (a) com prioridade e (b) sem prioridade

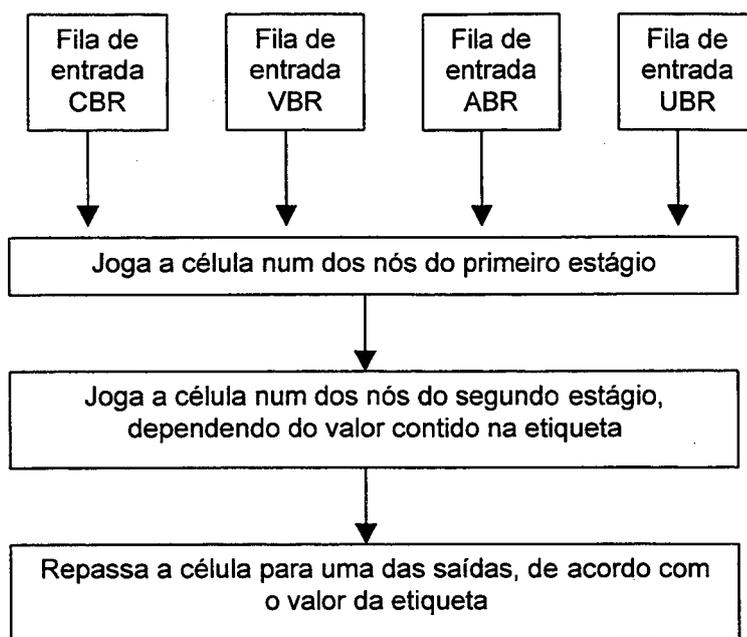


(a)

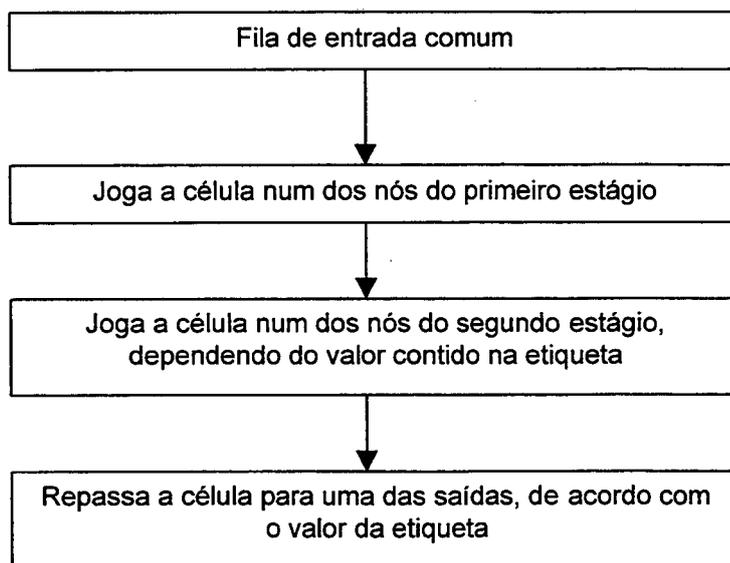


(b)

Figura 4.3 – Matriz com arquitetura de interconexão total (a) com prioridade e (b) sem prioridade



(a)



(b)

Figura 4.4 – Matriz com arquitetura de múltiplos estágios, (a) com prioridade e (b) sem prioridade

### Pós-comutação

Neste segmento assume-se as tarefas inerentes ao processamento da célula no momento em que ela já chegou à saída correta, e encontra-se em vias de ser repassada ao próximo comutador ou ao computador destino. Ela compreende a montagem do cabeçalho ou do quadro e a interação com o meio. Como o funcionamento interno do comutador é mais rápido do que a velocidade do meio físico, são utilizados *buffers*, para que as células ou os quadros não sejam perdidos por causa desta discrepância. Mesmo assim, pode haver descarte.

Neste trabalho, considerou-se que as fases de pré- e pós-processamento são comuns a todos os comutadores. Já em relação à matriz, decidiu-se por criar modelos que refletissem as diferentes arquiteturas existentes, apresentadas em [FAH97]. Assim, foram feitos três modelos : um com matriz de memória compartilhada, um com matriz de interconexão total, e um com matriz em estágios, seguindo redes de Banyan.

Além disso, para que se pudesse representar a prioridade por classe de tráfego na entrada da matriz, foi preciso criar um modelo para cada situação (com e sem prioridade), gerando-se, portanto, um total de seis modelos.

De modo a não se produzir uma complexidade maior do que a necessária, manteve-se o modelo com poucas entradas (quatro) e saídas (também quatro).

### Definição dos tempos dos processos

Para garantir a fidelidade do modelo ao sistema em avaliação, o primeiro caminho tomado na definição da duração de cada processo foi buscar esta informação junto aos fabricantes, na tentativa de se usar valores reais. Entretanto, esta informação não estava disponível, seja pela ausência de uma fonte, seja pelo impedimento de se tornar tal dado público, uma vez que a concorrência acirrada faz com que não seja interessante a divulgação de certas informações.

Desta forma, procurou-se usar valores que, embora não sejam exatamente aqueles encontrados no mundo real, fossem coerentes o suficiente para que se obtivessem resultados significativos. Para tanto, adotou-se as seguintes premissas :

- ◆ Entre os três segmentos (pré-processamento, matriz e pós-processamento), a matriz é o mais rápido, uma vez que se baseia fortemente em *hardware* e pouco em *software*;
- ◆ Considerando que para se evitar a situação de bloqueio de início de fila (*head-of-line blocking*) os fabricantes em geral implementam os *buffers* na saída, e que muito se vem estudando qual a capacidade ideal dos mesmos, é razoável se admitir que é na saída que se concentra a maior parte do tempo que a célula passa no comutador;
- ◆ Ainda que a célula gaste mais tempo nas saídas do que nas entradas, na parte do pré-processamento são feitas tarefas relativamente pesadas, como a verificação da conformidade das células.

Assim, foram atribuídas aos processos as durações a seguir. Observe que foi escolhido como unidade de tempo o milissegundo em detrimento do microssegundo, que é a unidade de tempo usada normalmente para se medir latência dentro de um comutador. Tal escolha deveu-se a restrições do *software* utilizado na simulação, mas não interfere de nenhuma forma nas conclusões obtidas.

#### Processo 1 – Quebra de quadros em células

Um quadro Ethernet tem um tamanho mínimo de 64 *bytes* e um tamanho máximo de 1516 *bytes*, o que gera entre 2 e 30 células na hora em que este quadro é dividido em células de 48 *bytes* (desconsiderando o cabeçalho). Como ligações Ethernet são predominantes em comutadores com interfaces não ATM, usou-se estes números como parâmetros na definição de quantas células seriam criadas a partir do recebimento de um quadro. Então, foi atribuído um tempo de 0,001 ms para cada célula gerada de uma parte do quadro, multiplicado pelo número de células geradas, número este que é o resultado de uma distribuição uniforme tendo 2 como mínimo e 30 como máximo.

#### Processo 2 – Tratamento de células de controle

Uma célula de controle contém, na maioria das vezes, um pedido de estabelecimento ou de término de conexão, informações de gerência ou de roteamento. Cada uma exige um tratamento diferente, mas por uma questão de simplicidade, foi

atribuído um tempo único de 0,006 ms. Dada a natureza da informação, estas células são consideradas de alta prioridade.

### Processo 3 – Verificação da conformidade da célula

Ao se receber uma célula de dados dentro de uma conexão já existente, deve-se verificar se ela está conforme com o contrato de tráfego firmado quando do estabelecimento da conexão, principalmente no que se refere à taxa de transmissão. Células não conformes são marcadas com CLP=1, ou seja, elegíveis para descarte se necessário. Para este processo, foi atribuído um tempo de 0,006 ms.

### Processo 4 – Translação de endereço e determinação da multiplicidade da célula

O endereço de uma célula é representado pelos identificadores VPI e VCI, constantes no cabeçalho. Entretanto, as células não mantêm os mesmos valores para os identificadores por todos os comutadores entre a fonte e o destino. Em vez disso, em cada comutador, novos valores são associados a estes identificadores. Ressalte-se que estes novos valores indicam também se a célula se destina a uma, várias ou todas as saídas. A este processo foi atribuído um tempo de 0,003 ms.

### Processo 5 – Multiplicação das células e adição da etiqueta de roteamento a cada uma (matriz de múltiplos estágios)

A matriz de múltiplos estágios faz a comutação baseando-se em uma etiqueta “afixada” previamente na célula. Tal tarefa é feita junto com a translação de endereço. Além disso, ela exige que, na situação de uma célula para várias saídas, a multiplicação seja feita ANTES da entrada da célula original na matriz de comutação. O tempo atribuído a estas tarefas foi de 0,002 ms por célula.

### Processos da matriz 1 – memória compartilhada

#### Processo 1.1 – Escrita na memória compartilhada

Processo relativamente rápido, pois não é feito nenhum teste na célula. Tempo atribuído : 0,002 ms.

*Processo 1.2 – Leitura da memória compartilhada e cópia para a(s) saída(s) de destino*

Este processo leva um pouco mais de tempo do que o anterior, pois embora a leitura seja rápida, é preciso verificar, a partir do endereço, quais são as saídas de destino. A este processo, foi atribuído um tempo de 0,004 ms.

*Processos da matriz 2 – interconexão total*

*Processo 2.1 – Transferência da célula pelo barramento*

Este processo é bastante rápido, já que a velocidade do barramento é grande e todas as saídas recebem a célula automaticamente. Tempo atribuído : 0,001 ms.

*Processo 2.2 – Teste de seleção na saída*

Como as células são enviadas a todas as saídas, independentemente do fato de a elas estarem destinadas ou não, há necessidade de uma seleção em cada saída para que os seus *buffers* sejam ocupados apenas por células que tenham aquela saída como destino. Tempo estimado para o teste : 0,001 ms.

*Processos da matriz 3 – múltiplos estágios*

*Processo 3.1 - Passagem da célula pelo primeiro estágio*

Em cada estágio pelo qual a célula passa, usa-se um dos bits da sua etiqueta para direcionar a célula para uma de suas duas saídas. O tempo atribuído a este processo é de 0,001 ms.

*Processo 3.2 – Passagem da célula pelo segundo estágio*

Da mesma forma que o processo 3.1. O tempo atribuído é o mesmo : 0,001 ms.

*Etapa de pós-processamento*

Este processo engloba as tarefas relacionadas à função de encaminhar a célula ou quadro via meio físico para o próximo passo no caminho definido pela conexão, seja ele um outro comutador ou o computador do destinatário final. Dentre estas tarefas encontram-se, por exemplo, a montagem do quadro ou cabeçalho da célula e interação

com o meio. O tempo atribuído é um pouco alto justamente por causa desta última tarefa : 0,012 ms, se se fizer necessária a montagem de um quadro, e 0,01 ms, em caso contrário.

#### **4.3.2 – Especificação das métricas**

A avaliação de desempenho é feita através de métricas, que são critérios para a comparação do desempenho entre sistemas, ou entre diferentes situações a que um sistema pode ser submetido. As métricas (também chamadas de variáveis de interesse, ou de desempenho) utilizadas aqui são a latência e a taxa de descarte por motivo de congestionamento. Além disso, pela importância atual das classes de tráfego, foram criadas métricas de latência e taxa de descarte para cada classe. Portanto, as variáveis de interesse são:

**Latência** - retardo que as células sofrem entre a entrada e a saída do comutador. No caso de células que foram criadas dentro do sistema, como as geradas pela quebra de quadros ou duplicadas, usa-se como momento de entrada a chegada do quadro ou da célula original ao comutador. Células descartadas não têm a sua latência computada.

Além da latência geral, quando foram levadas em conta todas as células não descartadas, foram usadas variáveis específicas para as classes de tráfego CBR, VBR, ABR e UBR, denominadas respectivamente : *Latência CBR*, *Latência VBR*, *Latência ABR* e *Latência UBR*.

**Taxa de Descarte** – esta métrica calcula o número de células descartadas em relação ao número total de células que passaram pela matriz de comutação.

De forma análoga à latência, também foram usadas métricas próprias para cada classe de tráfego, denominadas *Taxa de Descarte CBR*, *Taxa de Descarte VBR*, *Taxa de Descarte ABR* e *Taxa de Descarte UBR*.

### 4.3.3 – Definição dos parâmetros, fatores e carga

Os comutadores ATM têm um grande conjunto de parâmetros. Entre eles, pode-se citar:

- ◆ Protocolos de sinalização suportados (versão da UNI, NNI, etc);
- ◆ Número e velocidade de interfaces suportadas;
- ◆ Arquitetura da matriz de comutação;
- ◆ Processadores utilizados;
- ◆ Política de descarte em caso de congestionamento;
- ◆ Localização e capacidade dos *buffers*;
- ◆ Classes de tráfego suportadas.

Como o estudo envolvendo todos os parâmetros não seria possível, dada a complexidade introduzida, dentre eles foi retirado um subconjunto representativo (a decisão sobre a representatividade do parâmetro fundamenta-se na literatura, como em [XYL97] [SIL97] [KAL97] ) no que se refere ao desempenho, que foram usados como fatores. Fatores são os parâmetros que sofrerão uma variação durante o estudo, e os valores que cada fator assume são chamados de níveis. Deve-se tomar cuidado para que os fatores sejam um reflexo dos parâmetros que variam também no mundo real. Para esta simulação, os fatores considerados significativos são :

#### Prioridade por classe

Este fator leva em conta a classe de tráfego da célula que chega na matriz. A importância de se ter um mecanismo de prioridade cresce à medida em que se usa ATM cada vez mais para transportar tráfego extremamente sensível à latência junto com tráfego com menos exigências em relação ao tempo de permanência no sistema.

#### Capacidade dos buffers de saída

Há atualmente muita discussão a respeito do tamanho ideal dos *buffers* de saída, responsáveis por acomodar as células comutadas pela matriz e que ainda não puderam ser colocadas no meio físico. Quanto mais capacidade, melhor? Ou uma capacidade muito grande alivia o descarte, mas aumenta a latência? São questões que se

espera responder ao final do processo de experimentação. Estas questões são analisadas nos itens 6.2.1 e 6.2.2 deste trabalho.

### Intensidade de tráfego

O que acontece com a latência e o descarte quando o comutador começa a receber um tráfego pesado, talvez mais do que ele esteja preparado para receber? Ele lida com esta situação de uma forma elegante? Estas perguntas se tornam cada vez mais comuns, com o aumento de tráfego às vezes de forma drástica nas redes hoje existentes.

### Arquitetura da matriz de comutação

Muita pesquisa tem sido feita também em relação à arquitetura da matriz de comutação, tentando se chegar a uma arquitetura tão barata (em termos de número de componentes) quanto eficiente e escalável. Até que ponto as diferentes arquiteturas hoje disponíveis fazem diferença no tempo de latência?

### Distribuição de tráfego

Os comutadores ATM estão presentes em redes com vários tipos de tráfego, desde interligação de PABX até *backbone* de provedores Internet. É importante que os testes levem em conta estas diferentes situações.

Já os níveis associados a cada um deles são :

#### 1. Prioridade por classe

- ◆ **Com prioridade por classe** – neste nível, as matrizes de comutação têm como entrada uma fila para cada classe de tráfego, priorizando as células CBR sobre VBR, VBR sobre ABR, e ABR sobre UBR.
- ◆ **Sem prioridade de classe** – neste nível, as matrizes de comutação têm uma fila única de entrada, desprezando a classe de tráfego à qual a célula pertence.

## 2. Capacidade dos buffers de saída

- ◆ **Média capacidade** – neste nível, cada *buffer* pode armazenar até 400 células.
- ◆ **Alta capacidade** – aqui cada *buffer* armazena até 1000 células. 3.

## 3. Intensidade de tráfego

Para o tempo entre chegadas foi escolhida a distribuição exponencial, distribuição geralmente usada para este tipo de função. Como níveis, foram selecionados valores que, embora não tenham relação direta com a realidade, geram respectivamente uma carga média e pesada no modelo.

- ◆ **Normal** – distribuição exponencial com média de 0,3 por interface de entrada, ou uma média de 0,075 no total
- ◆ **Intenso** – distribuição exponencial com média de 0,08 por interface de entrada, ou uma média de 0,02 no total

## 4. Arquitetura da matriz de comutação

Conforme já mostrado na seção 4.3.1, são três as arquiteturas utilizadas :

- ◆ **Memória compartilhada** (*shared memory*)
- ◆ **Interconexão total** (*shared medium*)
- ◆ **Múltiplos estágios** (*space division*)

## 5. Distribuição de tráfego

Na eleição dos níveis deste fator, procurou-se representar as três situações mais encontradas em redes que usam comutadores ATM :

- ◆ **A interligação de PABX** – o ATM na rede telefônica é usado para emulação de circuitos dedicados. Neste caso, temos uma grande predominância

de tráfego CBR, pouca presença de quadros e células destinadas a mais de uma saída. No modelo, isso se traduz nos seguintes valores :

- classes : 60% CBR, 10% VBR, 15% ABR e 15% UBR
- multiplicidade : 80% *unicast* (destino de uma saída apenas), 10% *multicast* (várias saídas) e 10% *broadcast* (todas as saídas)
- células x quadros – 90% células e 10% quadros

- ◆ **Redes velozes para videoconferência** – o ATM se presta bem ao uso de videoconferência por causa do seu suporte a qualidade de serviço. Este tipo de uso se apóia principalmente sobre a classe VBR e *multicast*. No modelo, isso se traduz nos seguintes valores :

- classes : 10% CBR, 40% VBR, 25% ABR e 25% UBR
- multiplicidade : 40% *unicast*, 50% *multicast* (várias saídas) e 10% *broadcast* (todas as saídas)
- células x quadros – 60% células e 40% quadros

- ◆ **Interligação de redes locais tradicionais** – neste tipo de situação o ATM é usado como um *backbone* de alta velocidade para a interconexão das redes locais, que usam protocolos como TCP/IP. Nestas redes predomina um tráfego não muito sensível à latência sobre Ethernet ou Token Ring, e pouco *multicast* ou *broadcast*. No modelo, isso se traduz nos seguintes valores :

- classes : 10% CBR, 15% VBR, 40% ABR e 35% UBR
- multiplicidade : 80% *unicast*, 10% *multicast* (várias saídas) e 10% *broadcast* (todas as saídas)
- células x quadros – 20% células e 80% quadros

Os dados usados para análise foram coletados através da execução de vários experimentos. Cada experimento consiste na simulação dos modelos para diferentes combinações entre os níveis de cada fator.

#### 4.3.4 – Planejamento e realização dos experimentos

De uma forma geral, a decisão sobre as combinações a serem usadas deve ser tomada de acordo com o número de fatores, quantidade de níveis de cada um, tempo e recursos disponíveis para os testes. Existem, basicamente, três abordagens:

- ◆ **Simples** – Esta abordagem prevê a variação entre os níveis de um único fator a cada vez. Desta forma, interações entre fatores não são levadas em consideração;
- ◆ **Fatorial Completo** – Utiliza todas as possíveis combinações entre todos os níveis de cada fator. É a que gera mais informações, mas só é factível se o número de fatores e de níveis em cada um não for alto, e quando não for necessária a realização de várias replicações de um mesmo experimento;
- ◆ **Fatorial Fracionado** – Esta abordagem usa apenas uma parte das combinações que seriam usadas num fatorial completo, segundo certos critérios. Exige menos testes, mas em contrapartida também gera menos informações.

Levando-se em conta que neste trabalho o número de fatores e níveis é relativamente pequeno (3 fatores com 2 níveis, e 2 fatores com 3 níveis), e apenas uma replicação de cada experimento é necessária, chegou-se à conclusão que a abordagem mais adequada é a do fatorial completo.

Neste etapa o módulo de planejamento de experimentos do *software Statistica* foi utilizado para gerar a planilha com a lista de todas as simulações que deveriam ser feitas. Esta planilha é composta por colunas representando os fatores e as variáveis de interesse cujos valores deseja-se obter, e por linhas representando os experimentos, numa proporção de uma linha por experimento. Cada uma, portanto, contém os níveis que devem ser usados naquele experimento específico.

Feito o planejamento, procedeu-se à execução das simulações, em número de 72 (este número advém do total de possíveis combinações entre todos os níveis de cada fator :  $3 \times 3 \times 2 \times 2 \times 2$ ). Cada simulação gerou um valor para cada variável de interesse, que

foi passado para a planilha. Uma vez preenchida por completo, a planilha foi usada novamente pelo *software* **Statistica** para a análise estatística, apresentada no capítulo 5.

# CAPÍTULO V - ANÁLISE ESTATÍSTICA DOS RESULTADOS OBTIDOS

## 5.1 - Introdução

A realização dos experimentos com os modelos computacionais, conforme descrito no capítulo 4, geraram um conjunto de dados referentes aos dois grupos de métricas utilizados (grupo de latência e grupo de taxa de descarte). Sobre este conjunto foi feita uma análise estatística, que serviram de base para as conclusões sobre o comportamento do computador.

A análise inicia na seção 5.2, com algumas estatísticas básicas, entre elas a média e o desvio padrão para as métricas gerais (isto é, sem se prender a uma determinada classe de tráfego). Em seguida, na seção 5.3 é feita a análise de variância, para a determinação dos fatores que mais influenciam cada grupo de métricas. A seção 5.4 investiga esta influência um pouco mais a fundo através dos valores de  $R^2$  e  $r$ , e por último, a seção 5.5 mostra a interação entre as métricas gerais (Latência e Taxa de Descarte) através de um diagrama de dispersão.

## 5.2 - Estatísticas básicas

Para a análise estatística das variáveis, mais uma vez foi usado o *software* Statistica. Inicialmente, os valores foram consolidados em estatísticas descritivas, mais especificamente média, valores mínimos e máximos e desvio padrão, como mostram as tabelas 5.1 e 5.2.

<i>LATÊNCIA</i>	<i>MÉDIA</i>	<i>VALOR MÍNIMO</i>	<i>VALOR MÁXIMO</i>	<i>DESVIO PADRÃO</i>
Geral	2865,675	79,900	12522,000	4349,173
CBR	2861,119	78,500	12538,000	4350,824
VBR	2866,154	77,200	12545,000	4349,671
ABR	2868,464	79,100	12521,000	4350,224
UBR	2854,538	79,600	12021,000	4290,485

Tabela 5.1 – Estatísticas descritivas relativas à latência (em  $\mu$ s)

<i>DESCARTE</i>	<i>MÉDIA</i>	<i>VALOR MÍNIMO</i>	<i>VALOR MÁXIMO</i>	<i>DESVIO PADRÃO</i>
Geral	19,11	0,00	66,72	27,50
CBR	00,24	0,00	1,82	0,45
VBR	05,46	0,00	27,01	9,20
ABR	11,76	0,00	40,67	16,99
UBR	15,74	0,00	52,00	22,53

Tabela 5.2 – Estatísticas descritivas relativas à taxa de descarte (em %)

Já com estas estatísticas básicas, duas observações importantes podem ser feitas :

- ◆ Embora a classe UBR seja a de menor prioridade, foi a que teve na média a menor latência, embora esta diferença não seja estatisticamente representativa;
- ◆ Como pode-se ver pela diferença entre os valores mínimos e máximos, bem como pelo desvio padrão, a variação, tanto da latência como da taxa de descarte, foi bastante grande. Entretanto, as variáveis de latência apresentam uma grande homogeneidade de distribuição entre si, com valores de desvio

padrão bastante próximos. As variáveis de taxa de descarte, ao contrário, apresentam uma grande disparidade de valores, com valores máximos entre 1,82% e 52%, e desvios padrões variando entre 0,45% e 22,53%.

Com a intenção de descobrir como exatamente esta variação influencia o desempenho, passou-se em seguida a uma análise de variância, apresentada na seção 5.3.

### 5.3 - Análise de Variância

A análise de variância é feita através do teste F. Este teste é usado para verificar se existe uma diferença estatisticamente significativa entre médias. São duas as hipóteses:

- ◆  $H_0$  – as médias são iguais;
- ◆  $H_1$  – existe uma diferença estatisticamente significativa entre as médias.

A decisão entre se aceitar a hipótese  $H_0$  ou se recusá-la (aceitando  $H_1$ , portanto) dependerá do valor de  $p$ , obtido no teste. Se o valor de  $p$  for maior do que o nível de significância escolhido ( $\alpha$ ), então aceita-se  $H_0$ , ou seja, pode-se afirmar que em termos estatísticos as médias são iguais. Caso contrário, aceita-se  $H_1$ .

Este trabalho adota um nível de significância ( $\alpha$ ) de 0,05.

#### 5.3.1 – Análise de variância para o grupo de métricas de latência

A tabela 5.3 mostra o resultado do teste F para a variável geral LATÊNCIA, em relação a todos os fatores (prioridade de tráfego por classe, capacidade dos *buffers* de saída, intensidade de tráfego, arquitetura da matriz de comutação e distribuição do tráfego).

<i>FATOR</i>	<i>F<sub>obs</sub></i>	<i>p</i>
Prioridade	0,00000	0,998827
Capacidade	14,23079	0,000356
Intensidade	71,77633	0,000001
Matriz	0,06388	0,938181
Distribuição	19,29331	0,000001

Tabela 5.3 – Teste F para a variável LATÊNCIA

Analisando-se os valores de **p** nesta tabela, vê-se que para a variável LATÊNCIA, **p** assume um valor menor do 0,05 em relação aos fatores capacidade do *buffer*, intensidade de tráfego e distribuição de tráfego. Assim, pode-se afirmar que há diferença nas médias de latência para estes fatores, no nível de significância adotado. Este mesmo comportamento é seguido pelas variáveis LATÊNCIA-CBR, LATÊNCIA-VBR, LATÊNCIA-ABR e LATÊNCIA-UBR, como se pode ver nas tabelas A.1 a A.4, constantes no anexo A. Portanto, pode-se dizer que a variação de cada um destes três fatores tem influência significativa sobre variação da latência, independentemente da classe de tráfego da célula. Com efeito, atentando-se para as médias e desvios padrões de cada variável em relação a cada fator, verificamos que há uma grande diferença entre estes valores para cada nível. Já a influência da variação dos outros dois fatores (prioridade por classe e arquitetura da matriz de comutação) é pouca ou nenhuma. Os valores de média e desvio padrão para as variáveis de latência, agrupados por fator, podem ser vistos nas tabelas A.8 a A.32 do anexo A.

### 5.3.2 – Análise de variância para o grupo de métricas de taxa de descarte

A outra variável de desempenho geral é a TAXA-DESCARTE. A tabela 5.4 mostra os valores de F e **p** para esta variável, também relacionando-os para todos os fatores.

<i>FATOR</i>	<i>F<sub>obs</sub></i>	<i>p</i>
Prioridade	0,0001	0,992966
Capacidade	0,0056	0,940845
Intensidade	122,5581	0,000001
Matriz	0,1291	0,879097
Distribuição	31,7254	0,000001

Tabela 5.4 – Teste F para a variável TAXA-DESCARTE

Fazendo uma análise a partir dos valores contidos nesta tabela, vemos que para a variável TAXA-DESCARTE  $p$  assume um valor menor do que 0,05 para os fatores intensidade e distribuição de tráfego. Desta forma, pode-se afirmar que há diferença nas médias destes fatores para a taxa de descarte, no nível de significância adotado. Esta é a mesma situação das variáveis TAXA-DESCARTE-VBR, TAXA-DESCARTE-ABR e TAXA-DESCARTE-UBR, como pode ser visto nas tabelas A.5 a A.7 no anexo A. Logo, pode-se afirmar que os valores das taxas mencionadas seguramente sofrerão uma variação quando estes fatores forem alterados. Da mesma forma, pode-se ver nas tabelas A.33 a A.37, e A.43 a A.57, também presentes no anexo A, que a média e o desvio padrão mudam bastante de um nível para outro para os fatores especificados acima, enquanto que a mudança é bem pequena quando se trata dos outros fatores.

A variável TAXA-DESCARTE-CBR, porém, se comporta de uma maneira um pouco diferente das outras variáveis de descarte (como mostra a tabela 5.5), pois agora a arquitetura da matriz de comutação também tem influência no resultado do descarte para células CBR. As tabelas referentes a esta variável vão de A.38 a A.42. A prioridade por classe e a capacidade dos *buffers* continuam não influenciando o número de células CBR descartadas, mas no caso deste último fator, nota-se uma acentuada tendência no sentido de se mudar esta conclusão, pois o valor de  $p$  se aproxima do nível de significância escolhido ( $\alpha$ ).

<i>FATOR</i>	<i>F<sub>obs</sub></i>	<i>p</i>
Prioridade	0,33325	0,565775
Capacidade	3,55618	0,063864
Intensidade	47,7754	0,000001
Matriz	7,03388	0,001732
Distribuição	13,01137	0,000018

Tabela 5.5 – Teste F para a variável TAXA-DESCARTE-CBR

## 5.4 – Outras informações obtidas da análise de variância

### 5.4.1 – O coeficiente de determinação $R^2$

O coeficiente de determinação  $R^2$  indica quanto da variabilidade das métricas em estudo podem ser explicadas pela variação dos fatores, e quanto resulta da influência de elementos externos ao teste. Quanto maior o valor de  $R^2$ , mais se pode creditar a variação observada nas diferentes taxas pode ser explicada à variação dos fatores que foram significativos no teste F (ANOVA – ANalysis Of VAriance). Assim, pode-se dizer com mais segurança que o resultado da variável pela alteração dos níveis dos fatores não foi apenas uma coincidência, mas sim houve uma relação direta entre um e outro.

A partir dos resultados do teste F para o grupo de métricas de latência e para o grupo de métricas de taxa de descarte, são mostrados os valores de  $R^2$  na tabela 5.6.

Esta tabela mostra que uma grande parte da variação das métricas pode ser entendida como causada pela alteração dos níveis dos fatores. Expressando os valores da tabela em percentuais, temos um mínimo de 58% (para a taxa de descarte de células CBR) até 74,74% (para a taxa de descarte de células UBR) de variação explicada pelos fatores em estudo, sendo de 42% a cerca de 25% a variabilidade não explicada, isto é, causada por outras variações não controladas na experimentação.

<i>VARIÁVEIS</i>	$R^2$	<i>FATORES SIGNIFICATIVOS</i>
Latência geral	0,66088	capacidade <i>buffer</i> , intens. tráfego, distrib. tráfego
Latência CBR	0,66022	capacidade <i>buffer</i> , intens. tráfego, distrib. tráfego
Latência VBR	0,66045	capacidade <i>buffer</i> , intens. tráfego, distrib. tráfego
Latência ABR	0,66105	capacidade <i>buffer</i> , intens. tráfego, distrib. tráfego
Latência UBR	0,66694	capacidade <i>buffer</i> , intens. tráfego, distrib. tráfego
Taxa Descarte geral	0,74428	arquitetura matriz, intens. tráfego, distrib. tráfego
Taxa Descarte CBR	0,58910	intensidade tráfego, distribuição tráfego
Taxa Descarte VBR	0,67789	intensidade tráfego, distribuição tráfego
Taxa Descarte ABR	0,74268	intensidade tráfego, distribuição tráfego
Taxa Descarte UBR	0,74749	intensidade tráfego, distribuição tráfego

Tabela 5.6 – Valores de  $R^2$  para as variáveis em estudo

#### 5.4.2 – O coeficiente de correlação $r$

Outra informação conseguida através da estatística é fornecida pelos valores de  $r$ , que medem a correlação entre os fatores e as variáveis, isto é, qual a influência relativa de cada fator para uma variável. Os valores de  $r$  situam-se numa escala de 0 a 1, e quanto mais próximo de 1 o valor de  $r$  para um determinado fator, maior a influência deste nos valores da variável. Assim, pode-se dizer que o fator com o maior valor de  $r$  é o que causará a maior alteração da variável quando passar de um nível para outro.

As tabelas 5.7 e 5.8 mostram os valores de  $R$  de cada fator por variável. Pela grande diferença que existe entre eles, nota-se que existem fatores que não têm praticamente nenhuma importância relativa à métrica em questão, enquanto que outros têm uma importância bastante grande.

<i>FATOR</i>	<i>LATÊNCIA</i>	<i>LATÊNCIA</i>	<i>LATÊNCIA</i>	<i>LATÊNCIA</i>	<i>LATÊNCIA</i>
		<i>CBR</i>	<i>VBR</i>	<i>ABR</i>	<i>UBR</i>
Prior. classe	0,00	0,00	0,00	0,00	0,00
Capac. <i>buffers</i>	0,27	0,28	0,28	0,27	0,27
Intens. tráfego	0,62	0,62	0,62	0,62	0,62
Arquit. matriz	0,00	0,00	0,00	0,00	0,00
Distrib. tráfego	0,40	0,40	0,40	0,40	0,40

Tabela 5.7 – Valores de  $r$  para as métricas do grupo de latência, para cada fator

Assim, tem-se como fatores relevantes, em ordem decrescente, para as variáveis relativas à latência (LATÊNCIA, LATÊNCIA-CBR, LATÊNCIA-VBR, LATÊNCIA-ABR e LATÊNCIA-UBR): intensidade de tráfego e distribuição de tráfego.

FATOR	DESCARTE	DESCARTE	DESCARTE	DESCARTE	DESCARTE
		CBR	VBR	ABR	UBR
Prior. classe	0,00	0,05	0,01	0,00	0,00
Capac. buffers	0,00	0,15	0,01	0,00	0,01
Intens. tráfego	0,70	0,55	0,60	0,70	0,70
Arquit. matriz	0,02	0,12	0,00	0,00	0,00
Distrib. tráfego	0,48	0,28	0,18	0,49	0,46

Tabela 5.8 – Valores de r para as métricas do grupo de descarte, para cada fator

Já para as variáveis relativas à taxa de descarte (TAXA-DESCARTE, TAXA-DESCARTE-CBR, TAXA-DESCARTE-VBR, TAXA-DESCARTE-ABR E TAXA-DESCARTE-UBR) tem-se como fatores relevantes : intensidade de tráfego e distribuição de tráfego. É interessante notar-se, entretanto, que no caso da TAXA-DESCARTE-CBR, a influência não é tão concentrada como nas outras variáveis.

### 5.5 – Interação entre a latência e a taxa de descarte

A figura 5.1 mostra a interação entre as métricas gerais de latência e taxa de descarte, através de um diagrama de dispersão onde os pontos representam valores obtidos nos experimentos.

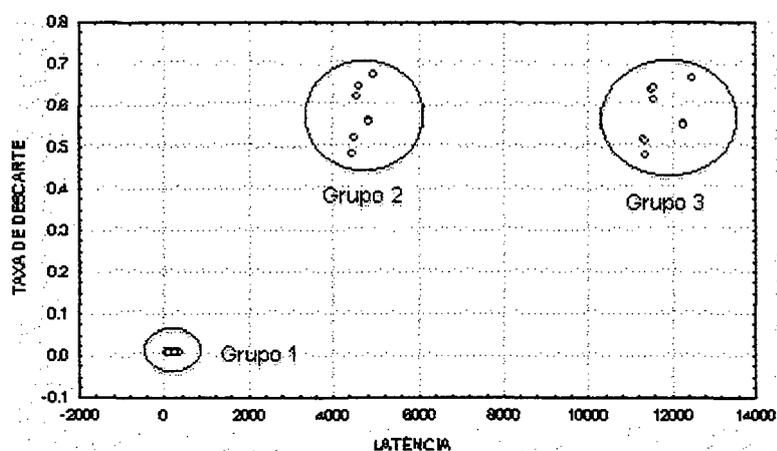


Figura 5.1 – Diagrama de dispersão para as métricas gerais de latência e taxa de descarte

Este gráfico apresenta a interação entre latência e taxa de descarte. Na verdade, ele mostra que, em relação à taxa de descarte, existem dois grupos principais onde, no interior destes, os valores desta variável pouco variam. Poderia-se classificá-los como, por exemplo, o grupo de baixa e o grupo de alta taxa de descarte.

Já em relação à latência, nota-se a existência de três grupos, que serão classificados como o grupo de baixa, média e alta latência.

Juntando-se a latência com a taxa de descarte, temos de novo três grandes grupos, já demarcados na figura, caracterizados e descritos a seguir. Como se fará referência novamente aos níveis de cada fator, convém antes rerepresentá-los :

#### 1. Prioridade por classe

- ◆ **Com prioridade por classe**
- ◆ **Sem prioridade de classe**

#### 2. Capacidade dos buffers de saída

- ◆ **Média capacidade** – cada *buffer* pode armazenar até 400 células
- ◆ **Alta capacidade** – cada *buffer* armazena até 1000 células

#### 3. Intensidade de tráfego

- ◆ **Normal** – distribuição exponencial com média de 0,3 por interface de entrada, ou uma média de 0,075 no total
- ◆ **Intenso** – distribuição exponencial com média de 0,08 por interface de entrada, ou uma média de 0,02 no total

#### 4. Arquitetura da matriz de comutação

- ◆ **Memória compartilhada** (*shared memory*)
- ◆ **Interconexão total** (*shared medium*)
- ◆ **Múltiplos estágios** (*space division*)

## 5. Distribuição de tráfego

### ◆ **A interligação de PABX**

- classes : 60% CBR, 10% VBR, 15% ABR e 15% UBR
- multiplicidade : 80% *unicast* (destino de uma saída apenas), 10% *multicast* (várias saídas) e 10% *broadcast* (todas as saídas)
- células x quadros – 90% células e 10% quadros

### ◆ **Redes velozes para videoconferência**

- classes : 10% CBR, 40% VBR, 25% ABR e 25% UBR
- multiplicidade : 40% *unicast*, 50% *multicast* (várias saídas) e 10% *broadcast* (todas as saídas)
- células x quadros – 60% células e 40% quadros

### ◆ **Interligação de redes locais tradicionais**

- classes : 10% CBR, 15% VBR, 40% ABR e 35% UBR
- multiplicidade : 80% *unicast*, 10% *multicast* (várias saídas) e 10% *broadcast* (todas as saídas)
- células x quadros – 20% células e 80% quadros

Os grupos, então, são :

#### Grupo 1 – Baixa latência e baixa taxa de descarte

Os elementos deste grupo apresentam as seguintes combinações de fatores e níveis :

- ◆ Prioridade por classe – em qualquer um dos níveis
- ◆ Capacidade dos buffers – em qualquer um dos níveis

- ◆ Intensidade de tráfego – em qualquer um dos níveis no caso de o fator distribuição do tráfego estar no primeiro nível , e no primeiro nível no caso de o fator distribuição de tráfego estar num dos outros dois níveis
- ◆ Arquitetura da matriz – em qualquer um dos níveis
- ◆ Distribuição do tráfego – nos dois primeiros níveis, caso a intensidade de tráfego esteja no primeiro, e somente no primeiro nível se a intensidade de tráfego estiver no segundo

Em suma, este grupo se caracteriza por uma baixa carga (isto é, baixa taxa de chegada de células ou quadros) e por uma baixa proporção de tráfego baseado em quadros.

#### Grupo 2 – Média latência e alta taxa de descarte

Os elementos deste grupo apresentam as seguintes combinações de fatores e níveis :

- ◆ Prioridade por classe – em qualquer um dos níveis
- ◆ Capacidade dos buffers – no primeiro nível
- ◆ Intensidade de tráfego – no segundo nível
- ◆ Arquitetura da matriz – em qualquer um dos níveis
- ◆ Distribuição do tráfego – no segundo ou terceiro nível

Resumindo, as características marcantes deste grupo são uma alta carga no sistema (com uma proporção média ou alta de quadros sendo recebidos), e *buffers* com média capacidade.

#### Grupo 3 – Alta latência e alta taxa de descarte

Os elementos deste grupo apresentam as seguintes combinações de fatores e níveis :

- ◆ Prioridade por classe – em qualquer um dos níveis
- ◆ Capacidade dos buffers – no segundo nível
- ◆ Intensidade de tráfego – no segundo nível

- ◆ Arquitetura da matriz – em qualquer um dos níveis
- ◆ Distribuição do tráfego – no segundo ou terceiro nível

Como o segundo grupo, neste também existe uma grande demanda ao sistema. A diferença reside no fato de que o terceiro grupo representa a situação em que os *buffers* estão na sua capacidade máxima.

Esta divisão em grupos, mostrando a grande interação entre latência e taxa de descarte, já pode ser identificada na seção 5.3, que mostra o resultado da aplicação do teste F para estas duas métricas. Este teste mostra não apenas que três fatores tiveram influência sobre a variação da latência, e dois fatores influenciaram a variação da taxa de descarte, mas também mostra que ambas as métricas têm dois fatores significativos em comum : intensidade e distribuição de tráfego. Isto posto, já podia-se esperar que a variação destes fatores provocasse uma variação conjunta destas métricas. Além disso, o fato de se ter poucos grupos evidencia que a mudança dos níveis dos fatores tendia a empurrar os valores das métricas para extremos, em vez de provocar variações gradativas.

Concluindo, a análise estatística realizada teve como objetivo delinear o comportamento dos modelos computacionais com um embasamento científico que permitisse a extrapolação para o sistema real. É com base nestes resultados, então, que pode-se chegar às conclusões sobre o comportamento dos comutadores existentes, que é assunto do próximo capítulo.

# CAPÍTULO VI – COMENTÁRIOS FINAIS

## 6.1 – Introdução

A avaliação de desempenho dentro do universo ATM é um tema bastante abrangente, pois envolve diversos aspectos relativos às tecnologias de transmissão de voz, dados e imagens. O que norteou este trabalho foi a idéia de definir-se o funcionamento de um comutador típico, salientando as diversas formas de interação entre seus componentes, o comportamento deste equipamento sob determinadas circunstâncias de tráfego e, principalmente, identificar fatores que influenciam o desempenho, e como selecioná-los de modo que o desempenho seja otimizado.

Pela ausência de um campo real, adotou-se a técnica da simulação, através de modelos computacionais. Procurou-se desenvolver os modelos da forma mais fiel ao sistema real, mas, infelizmente, algumas das informações utilizadas tiveram que ser estimadas, pois dados reais não estavam disponíveis.

As duas variáveis de interesse, ou métricas, escolhidas para medir o desempenho do comutador foram latência e taxa de descarte. À guisa de completude, foram analisadas também a latência e a taxa de descarte para cada classe de tráfego (CBR, VBR, ABR e UBR), para investigar se havia alguma diferença significativa entre as métricas mais gerais e as específicas para cada classe.

Depois de conduzidos os experimentos, os resultados foram analisados usando análise de variância. Este método revelou algumas informações interessantes, em geral corroborando observações empíricas e outros trabalhos publicados [SIL97].

Considerando todo o processo de desenvolvimento dos modelos, que descreveu todo o funcionamento do comutador, e os resultados analisados na próxima seção, pode-se dizer que os objetivos do trabalho foram plenamente alcançados.

A seção 6.2 discute os resultados obtidos na análise estatística dos valores obtidos na experimentação. A seção 6.3 reafirma a importância deste trabalho, e a seção

6.4 estabelece suas limitações. Por fim, a seção 6.5 faz algumas sugestões para futuros trabalhos.

## 6.2 – Resultados obtidos

As subseções 6.2.1 e 6.2.2 apresentam os resultados referentes à latência e à taxa de descarte respectivamente, e as conclusões que podem ser tiradas em cada caso.

### 6.2.1 – Latência

A latência mede o tempo que o comutador precisou para levar a célula (ou o quadro, quebrado em células) da interface de entrada até a interface de saída. Portanto, quanto menor o tempo, melhor. O que pode-se depreender dos testes é que, de fato, este tempo é pequeno, desde que a intensidade do tráfego não seja maior que a capacidade de comutação do equipamento – apesar de nesta situação já haver descarte, as células que ficam no sistema tendem a encher os *buffers*, desta forma fazendo com que a demora no atendimento a uma dada célula tenda a crescer consideravelmente.

Dáí vem outra constatação importante: a capacidade dos *buffers* deve ser cuidadosamente planejada. *Buffers* muito pequenos antecipam o descarte. *Buffers* muito grandes, por outro lado, causam menos descarte, mas aumentam tanto a latência, que a célula acaba tornando-se sem valor no destino, por *timeout*. Isto é ainda pior do que o descarte, pois consome recursos por um longo tempo sem representar atividade útil. O protocolo TCP, por exemplo, é bastante sensível ao retardo e, caso não receba nenhuma resposta dentro de um período de tempo preestabelecido, entra numa fase de recuperação conhecida como *slow-start*, o que tem sérias implicações no desempenho [KAL96].

A grande influência da distribuição do tráfego na latência se deve menos à classe predominante na taxa de envio de células ou quadros (CBR, VBR, ABR ou UBR) do que ao percentual de quadros recebidos. Na comutação, os quadros são quebrados em células, o que em se tratando de Ethernet significa que até 30 células podem ser geradas para cada um. Além da quantidade, o próprio processo de divisão consome um tempo precioso.

Já a arquitetura da matriz e a divisão por prioridade mostraram ser de pouca relevância para a latência. É compreensível, uma vez que a comutação propriamente dita acontece no *hardware*, muito mais rápido que processos realizados em *software* (ressalte-se, porém, que há várias outras maneiras de se implementar a prioridade, além da usada neste trabalho, e portanto pode haver um outro tipo de comportamento). Todas as classes tiveram sua latência aumentada de forma homogênea. Para que classes mais sensíveis ao retardo, como CBR e VBR, tenham um tratamento mais eficiente, outras implementações devem ser buscadas.

Em relação aos outros fatores, vê-se que eles não tiveram sua importância alterada da latência geral para a latência por classe.

### 6.2.2 – Taxa de descarte

A outra métrica utilizada para comparação foi a taxa de descarte, calculada como o número de células descartadas sobre o número de células comutadas pela matriz. Esta métrica, como era de se esperar, é influenciada basicamente pela intensidade de tráfego. Como na latência, há também uma participação da distribuição de tráfego. A matriz de comutação e a divisão por prioridade pouco importaram na taxa global; isto mostra que entre as arquiteturas de matrizes, não há nenhuma que gere algum tipo de gargalo. A questão da capacidade dos *buffers*, porém, é interessante. Num primeiro raciocínio, parece óbvio que o tamanho dos *buffers* sempre terá uma posição decisiva na taxa de descarte. Isso só é verdade, porém até um certo ponto. Os resultados mostram que, se o congestionamento no comutador é grande, o tamanho dos *buffers* é irrelevante, pois eles ficam rapidamente saturados e, com o passar do tempo, as taxas de descarte acabam tornando-se similares.

As taxas de descarte por classe têm em comum com as latências o fato de que os mesmos fatores que influenciaram a taxa global também foram as que mais influenciaram as taxas por classe. Mas enquanto os valores médios de todas as latências mantiveram-se mais ou menos estáveis, os valores médios das taxas de descarte variaram muito, e de acordo com que o que cabia ao comutador fazer, as células das classes ABR e UBR foram descartadas em quantidade bem superior às células das classes CBR e VBR. Tal discrepância entre a latência e a taxa de descarte pode, a princípio, parecer um contrasenso, pois enquanto que a latência de células CBR e UBR

é bem próxima, o descarte de UBR é muito maior. Isso se explica pelo número de células CBR comutadas, que é sempre muito maior do que o de células UBR. Estas últimas não perdem tanto tempo nos *buffers*, pois logo são descartadas.

Digno de nota, no entanto, é o comportamento da taxa de descarte CBR, que, diferentemente das demais, acusou uma influência da arquitetura da matriz e uma tendência de influência da capacidade dos *buffers*. Este comportamento não tem uma razão muito clara; talvez a explicação seja que, como o descarte de células CBR foi muito pequeno, não foram gerados dados suficientes para uma análise correta por parte do *software* utilizado.

### 6.2.3 – Resumo dos resultados

Conforme mostram os resultados da análise de variância (mostrados na seção 5.3) e o diagrama de dispersão apresentado na seção 5.5, e levando-se em conta os comentários feitos nas seções 6.2.1 e 6.2.2, poderia-se resumir as maiores preocupações no que se refere a um comutador ATM em :

- ◆ Que ele tenha um barramento interno rápido o suficiente para processar células ou quadros que porventura cheguem na taxa máxima das entradas, sem se tornar um gargalo;
- ◆ Que os *buffers* sejam capazes de acomodar um tráfego extra de células (ou quadros) no caso, por exemplo, de várias entradas enviando para uma única saída. Entretanto, estes buffers não podem introduzir um retardo tal que torne a célula sem valor no destino;
- ◆ Que o equipamento implemente uma política de descarte que garanta a prioridade das células mais sensíveis ao retardo.

Acredita-se que estes influenciadores de desempenho já sejam do conhecimento dos fabricantes, pois nota-se por parte da indústria uma atenção quanto a eles. Modelos recentes de comutadores usam *buffers* por VCC (conexões de circuito virtual), *backplanes* mais rápidos do que a capacidade máxima das entradas, entre outras inovações.

## 6.3 – Importância do trabalho

Posto que a proposta da tecnologia ATM é bastante ambiciosa, visando a integrar várias tecnologias hoje muito diferentes, não é de se admirar que comutadores ATM sejam máquinas extremamente complexas. Para caracterizá-los, precisa-se fazer referência a dezenas de conceitos, siglas e padrões. Se a simples definição do que o compõe já é difícil, muito mais complicado é identificar, dentre os componentes, quais são os mais importantes.

A relevância deste trabalho reside, primeiramente, na tipificação de um comutador, ressaltando sua funcionalidade básica e variações de arquitetura. Isto contribui para dar uma maior clareza no entendimento de seu funcionamento. Além disso, dois outros aspectos devem ser destacados :

- ◆ Para o estudo do comportamento do comutador, teve-se o cuidado de simular circunstâncias que representassem a grande maioria das situações reais em que estes equipamentos são utilizados, fazendo com os resultados encontrados sejam úteis para a maior parte dos projetistas de rede;
- ◆ O uso de métodos científicos, como a simulação e a estatística, para identificar parâmetros significativos para o desempenho de um comutador, além dos valores destes parâmetros que devem ser escolhidos para que o desempenho seja o melhor possível, dá a estes resultados um alto grau de confiabilidade, se comparados com outros obtidos de fabricantes e periódicos, cujos métodos de avaliação são, em geral, bastante empíricos e discutíveis.

## 6.4 – Limitações

Como já foi salientado em outras seções, um comutador ATM é um equipamento bastante complexo. Uma tentativa de se representar todas as tarefas envolvidas no seu funcionamento resultaria num modelo computacional com tantas variáveis que tornaria a análise impraticável.

Além disso, considerando-se o contexto de uma rede, a interação entre comutadores também acrescenta uma grande complexidade à criação dos modelos.

Assim, procurou-se reproduzir nos modelos as funcionalidades “vitais” destes equipamentos, isto é, aquelas que fossem imprescindíveis para uma avaliação de desempenho. Funções como manutenção de rotas e processamento de estabelecimento e término de conexões foram agrupadas simplesmente como tráfego de controle, e a elas foi dado um único tratamento.

Uma outra ressalva importante é a questão dos tempos atribuídos aos processos representados nos modelos. Mais uma vez, salienta-se que os tempos foram estimados, com base nas informações publicadas pelos fabricantes [XYL97] [IBM97a] [3COM97]. Portanto, os resultados obtidos dos experimentos poderiam ser diferentes, caso se tivesse estimado outras durações para estes processos.

## 6.5 – Sugestões para trabalhos futuros

Dada a importância da avaliação de desempenho na área de redes de computadores em geral, e na área de ATM em particular, existe muito espaço para outras pesquisas. O trabalho aqui apresentado poderia ser estendido de várias formas, como por exemplo :

- ◆ A criação de um único modelo genérico de um comutador no *software* de simulação (aqui foram usados 6 diferentes modelos) que aceitasse características específicas (como a capacidade dos *buffers*) através de uma interface orientada a menu, desta forma possibilitando a simulação do funcionamento de comutadores com dezenas de diferentes características;
- ◆ Considerando o crescente uso do padrão *LAN Emulation* (que faz a integração entre ATM e as atuais tecnologias de redes locais), seria interessante uma avaliação de desempenho de toda uma “sessão” dentro de uma Elan, levando em conta a comunicação entre LEC e LECS, LEC e LES, LEC e BUS, e finalmente entre LECs. Esta avaliação serviria como uma boa referência para a escolha de um comutador para a interligação de redes locais;
- ◆ Por fim, é sempre importante uma comparação entre resultados obtidos por técnicas diversas. Assim, caso haja a oportunidade para tal, seria interessante a realização de testes semelhantes num comutador real através da técnica de medição.

## CAPÍTULO VII - BIBLIOGRAFIA

- [3COM97] *CoreBuilder 7000 Family*. <http://www.3com.com>, 1997.
- [3COM97a] *Scaling Performance with ATM LAN Emulation*. <http://www.3com.com>, 1997.
- [AHU95] Ahuja, R., Keshav, S., Saran, H. *Design, Implementation and Performance of a Native Mode ATM Transport Layer*. AT&T Bell Labs, 1995.
- [AND96] Andrews, Eric. *MPOA Ties All Together*. Revista Data Communications, abril 1996.
- [ATM94] *User-Network Interface Specification Version 3.1* ATM Forum, 1994.
- [ATM94a] *Introduction to ATM Forum Test Specifications*. ATM Forum, 1994.
- [ATM96] *Integrated Local Management Interface (ILMI) Specification Version 4.0*. ATM Forum, 1996.
- [ATM96a] *Traffic Management Specification Version 4.0*. ATM Forum, 1996.
- [ATM97] *The ATM Forum Glossary*. ATM Forum, 1997.
- [AZZ97] Azzam, Albert A. *A Closer Look at ATM Over ADSL*. The ATM Forum Newsletter, 1997.
- [BRA91] Bradner, Scott et al. *Benchmarking Terminology for Network Interconnection Devices*. RFC 1242, IETF, 1991.
- [BRA96] Bradner, Scott, McQuaid, Jim. *Benchmarking Methodology for Network Interconnection Devices*. RFC 1944, IETF, 1996.
- [CAT96] Catanzariti, Sergio. *ATM Testing : Standards Pave the Way*. Revista Data Communications, junho 1996.
- [CAV94] Cavanaugh, John D. *Protocol Overhead in IP/ATM Networks*. Minnesota Supercomputer Center, 1994.
- [CRA97] Craig, Robert. *Terminology for Cell/Call Benchmarking*. Internet-Draft, IETF, 1997.

- [CRO95] Crosby, Simon A. *Performance Management in ATM Networks*. Tese de Doutorado, University of Cambridge, 1995.
- [FAH97] Fahmy, Sonia. *A Survey of ATM Switching Techniques*. Department of Computer and information Science, Ohio State University, 1997.
- [FOR97] *ForeRunner 200E Adapter*. <http://www.fore.com>, 1997.
- [FOR97a] *Fore Systems LAN Backbone ATM Switch Competitive Guide*. Fore Systems, 1997.
- [FOU95] Fouquet, Yves A. et al. *ATM Performance Measurement : Throughput Bottlenecks and Technology Barriers*. Computer Systems Laboratory, National Institute of Standards and Technology, 1995.
- [GOL95] Golmie, Nada, Koenig, Alfred & Su, David. *The NIST ATM Network Simulator - Operation and Programming version 1.0*. National Institute of Standards and Technology, 1995.
- [GOY97] Goyal, Rohit et al. *UBR+ : Improving Performance of TCP over ATM-UBR Service*. ICC'97, Montreal, 1997.
- [HAL96] Hall, Brian. *Beej's Guide to Networking Programming version 1.5.1*. <http://www.ecst.csuchico.edu/~beej/guide/net>
- [HAR97] Harris, Ethan. *ATM Software - A Guide to Evaluation and Selection*. The ATM Forum Newsletter, 1997.
- [HP95] *Netperf : A Network Performance Benchmark revision 2.0, manual do usuário*. Information Networks Division, Hewlett-Packard Company, 1995.
- [HUG95] Hughes, David & Hooshmand, Kambiz. *ABR Stretches ATM Network Resources*. Revista Data Communications, abril 1995.
- [HYM96] Hymas, William J. et al. *ATM Extensions to the Socket Programming Interface in ALX 4.2*. IBM European Networking Center. <http://www.rs6000.ibm.com>
- [IBM97] *IBM Turboways 100 ATM Adapter*. <http://www.networking.ibm.com>, 1997.
- [IBM97a] *IBM 8260 Nways Multiprotocol Switching Hub*. <http://www.networking.ibm.com>, 1997
- [INT97] *Interphase 5515 PCI ATM Adapter*. <http://www.iphase.com>, 1997.

- [INT97a] *Interphase : Increasing TCP LAN Performance by Using ABR Networks*. <http://www.ipphase.com>, 1997.
- [JAI90] Jain, Raj. *Congestion Control in Computer Networks : Issues and Trends*. IEEE Network Magazine, 1990.
- [JAI91] Jain, Raj. *The Art of Computer Systems Performance Analysis, Techniques for Experimental Design, Measurement, Simulation and Modeling*. John Wiley & Sons, Inc, 1991.
- [JAI92] Jain, Raj. *Myths About Congestion Management in High-Speed Networks*. IFIP TC6 4. Conference on Information Networks and Data Communication, Helsinki, 1992.
- [JAI94] Jain, Raj. *ATM Networking : Issues and Challenges Ahead*. The Ohio State University, 1994.
- [JAI96] Jain, Raj. *Congestion Control and Traffic Management in ATM Networks : Recent Advances and a Survey*. Department of Computer and Information Science, The Ohio State University, 1996.
- [KAL96] Kalyanaraman, Shiv et al. *Performance of TCP/IP over ABR Service on ATM Networks*. Globecom 96, 1996.
- [KAL97] Kalyanaraman, Shiv et al. *Performance and Buffering Requirements of Internet Protocols over ATM ABR and UBR Services*. The Ohio State University, 1997.
- [KAL97a] Kalyanaraman, Shiv et al. *Performance of TCP/IP over ABR on ATM Backbone and with Various VBR Traffic Patterns*. ICC'97, Montreal, 1997.
- [KAR94] Kara, M. et al. *An Empirical Study of Video Application Performance over ATM and Ethernet Networks*. Proceedings of the 2. IFIP Workshop on Performance Modelling and Evaluation of ATM Networks, 1994.
- [KAR97] Kara, Mourad & Rahin, Mohammad A. *Towards a Framework for Performance Evaluation of TCP Behaviour over ATM Networks*, ICC'97, 1997.
- [KEL98] Kelton, W. D., Sadowski, Randall P. & Sadowski, Deborah A. *Simulation with Arena*. WCB McGraw-Hill, 1998.

- [LIN95] Lin, Mengjou et al. *Distributed Network Computing over Local ATM Networks*. IEEE Journal on Selected Areas in Communications Special Issue of ATM LANs. 1995.
- [LY97] Ly, Faye et al. *Definitions of Supplemental Managed Objects for ATM Management*. Internet-Draft, IETF, 1997.
- [MAH96] Mah, Bruce A. *On the Use of Quality of Service in IP over ATM*. The Tenet Group, University of California at Berkeley, 1996.
- [MEL97] Melo, Edison T. L. *redeUFSC - Infraestrutura e Serviços*. Palestra, 1997.
- [MOL94] Moldeklev, Kjersti & Gunningberg, Per. *How a large ATM MTU causes deadlocks in TCP data transfers*. Norwegian Telecom Research, 1994.
- [NETC97] *Hardcore ATM Switches for the WAN*. Revista Network Computing, edição 819, <http://techweb.cmp.com>, 1997.
- [NETC97a] *ATM Backbone Switches*. Revista Network Computing, edição 821, <http://techweb.cmp.com>, 1997.
- [NETC97b] *ATM NICs : Ready to Sail, But Not Ready to Scale*. Revista Network Computing, edição 814, <http://techweb.cmp.com>, 1997.
- [OLA97] Olabe, M. A., Ferro, A. & Espinosa, K. *Using COMNET III Simulations to Model Video Traffic and Analyze Multiplexing Performance in an ATM Network*. University of the Basque Country, 1997.
- [RAM95] Ramakrishnan, K. & Newman, Peter. *ATM Flow Control : Inside the Great Debate*. Revista Data Communications, junho 1995.
- [RYB96] Rybczynski, Tony & Narayanan, Ravi. *ATM Gets Real About Real-Time Apps*. Revista Data Communications, outubro 1996.
- [SOA95] Soares, Luiz Fernando G. et al. *Redes de Computadores, das LANs, MANs e WANs às Redes ATM*, 2a. edição. Editora Campus, 1995.
- [SIL97] Silva, Anderson O. et al. *Comutadores ATM*. Laboratório Telemídia, PUC-Rio, 1997.
- [SWA97] Swallow, George. *MPOA, VLANS and Distributed ROUTERS*. The ATM Forum Newsletter, 1997.

- [TAY96] Taylor, Martin. *Voice Over ATM : A Sound Assessment*. Revista Data Communications, dezembro 1996.
- [WOB98] Wobus, John M. *LAN Technology Scorecard*. Cell-relay mailing list, 1998.
- [XYL97] *Some Things to Consider When Selecting an ATM Switch*. Xylan, 1997.

## CAPÍTULO VIII – ANEXOS

### ANEXO A – TABELAS DE RESULTADOS OBTIDOS NO STATISTICA

#### Tabelas com teste F para as variáveis de latência

<i>FATOR</i>	<i>F</i>	<i>p</i>
Prior. classe	0,00001	0,997581
Capac. buffers	14,27868	0,000348
Intens. tráfego	71,55274	0,000001
Arquit. matriz	0,06792	0,934404
Distrib. tráfego	19,19391	0,000001

Tabela A.1 – Teste F para a variável LATÊNCIA-CBR

<i>FATOR</i>	<i>F</i>	<i>p</i>
Prior. classe	0,00002	0,996074
Capac. buffers	14,26129	0,000351
Intens. tráfego	71,63226	0,000001
Arquit. matriz	0,06571	0,936461
Distrib. tráfego	19,22948	0,000001

Tabela A.2 – Teste F para a variável LATÊNCIA-VBR

<i>FATOR</i>	<i>F</i>	<i>p</i>
Prior. classe	0,00000	0,999614
Capac. buffers	14,21820	0,000358
Intens. tráfego	71,85894	0,000001
Arquit. matriz	0,06317	0,938847
Distrib. tráfego	19,30730	0,000001

Tabela A.3 – Teste F para a variável LATÊNCIA-ABR

<i>FATOR</i>	<i>F</i>	<i>p</i>
Prior. classe	0,00003	0,995701
Capac. buffers	13,66563	0,000455
Intens. tráfego	74,41574	0,000001
Arquit. matriz	0,02781	0,972586
Distrib. tráfego	20,00937	0,000001

Tabela A.4 – Teste F para a variável LATÊNCIA-UBR

### **Tabelas com teste F para as variáveis de taxa de descarte**

<i>FATOR</i>	<i>F</i>	<i>p</i>
Prior. classe	0,00716	0,932843
Capac. buffers	0,02478	0,875409
Intens. tráfego	70,97516	0,000001
Arquit. matriz	0,31290	0,969212
Distrib. tráfego	31,81121	0,000001

Tabela A.5 – Teste F para a variável TAXA-DESCARTE-VBR

<i>FATOR</i>	<i>F</i>	<i>p</i>
Prior. classe	0,0016	0,968347
Capac. buffers	0,000001	0,999278
Intens. tráfego	120,7901	0,000001
Arquit. matriz	0,0009	0,999133
Distrib. tráfego	31,9634	0,000001

Tabela A.6 – Teste F para a variável TAXA-DESCARTE-ABR

<i>FATOR</i>	<i>F</i>	<i>p</i>
Prior. classe	0	0,995523
Capac. buffers	0,0086	0,926206
Intens. tráfego	125,5062	0,000001
Arquit. matriz	0,0150	0,985152
Distrib. tráfego	31,9575	0,000001

Tabela A.7 – Teste F para a variável TAXA-DESCARTE-UBR

**Tabelas de média e desvio padrão para a métrica Latência  
Geral por fator**

<i>FATOR PRIORIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Com prioridade	2865,211	4378,427
Sem prioridade	2866,139	4381,829

Tabela A.8 – média e desvio padrão do fator PRIORIDADE para a variável  
LATÊNCIA (em  $\mu$ s)

<i>FATOR</i> <i>CAPACIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
400 células	1679,706	2149,889
1000 células	4051,644	4349,173

Tabela A.9 – média e desvio padrão do fator CAPACIDADE para a variável  
LATÊNCIA (em  $\mu$ s)

<i>FATOR</i> <i>INTENSIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Normal	202,194	101,334
Intenso	5529,156	4875,156

Tabela A.10 – média e desvio padrão do fator INTENSIDADE para a variável  
LATÊNCIA (em  $\mu$ s)

<i>FATOR</i> <i>MATRIZ</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Memória compart.	2780,575	4304,440
Interconexão total	3024,450	4630,138
Múltiplos estágios	2792,000	4287,751

Tabela A.11 – média e desvio padrão do fator MATRIZ para a variável  
LATÊNCIA (em  $\mu$ s)

<i>FATOR DISTRIBUIÇÃO</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Interligação PABX	104,892	24,992
Videoconferência	4181,171	4798,360
Interlig. redes locais	4310,963	4840,153

Tabela A.12 – média e desvio padrão do fator DISTRIBUIÇÃO para a variável  
LATÊNCIA (em  $\mu$ s)

### **Tabelas de média e desvio padrão para a métrica Latência CBR por fator**

<i>FATOR PRIORIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Com prioridade	2860,161	4379,900
Sem prioridade	2862,078	4383,682

Tabela A.13 – média e desvio padrão do fator PRIORIDADE para a variável  
LATÊNCIA-CBR (em  $\mu$ s)

<i>FATOR CAPACIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
400 células	1671,550	2143,073
1000 células	4050,689	5558,455

Tabela A.14 – média e desvio padrão do fator CAPACIDADE para a variável  
LATÊNCIA-CBR (em  $\mu$ s)

<i>FATOR INTENSIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Normal	198,194	99,543
Intenso	5524,044	4878,805

Tabela A.15 – média e desvio padrão do fator INTENSIDADE para a variável  
LATÊNCIA-CBR (em  $\mu$ s)

<i>FATOR MATRIZ</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Memória compart.	2772,233	4300,668
Interconexão total	3025,008	4637,897
Múltiplos estágios	2786,117	4288,019

Tabela A.16 – média e desvio padrão do fator MATRIZ para a variável  
LATÊNCIA-CBR (em  $\mu$ s)

<i>FATOR DISTRIBUIÇÃO</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Interligação PABX	103,725	24,721
Videoconferência	4175,267	4802,721
Interlig. redes locais	4304,367	4843,443

Tabela A.17 – média e desvio padrão do fator DISTRIBUIÇÃO para a variável  
LATÊNCIA-CBR (em  $\mu$ s)

**Tabelas de média e desvio padrão para a métrica Latência VBR  
por fator**

<i>FATOR PRIORIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Com prioridade	2864,600	4379,264
Sem prioridade	2867,708	4381,995

Tabela A.18 – média e desvio padrão do fator PRIORIDADE para a variável  
LATÊNCIA-VBR (em  $\mu$ s)

<i>FATOR CAPACIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
400 células	1678,031	2145,248
1000 células	4054,278	5556,421

Tabela A.19 – média e desvio padrão do fator CAPACIDADE para a variável  
LATÊNCIA-VBR (em  $\mu$ s)

<i>FATOR INTENSIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Normal	203,367	100,661
Intenso	5528,942	4876,850

Tabela A.20 – média e desvio padrão do fator INTENSIDADE para a variável  
LATÊNCIA-VBR (em  $\mu$ s)

<i>FATOR MATRIZ</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Memória compart.	2778,117	4301,982
Interconexão total	3027,225	4634,349
Múltiplos estágios	2793,121	4287,090

Tabela A.21 – média e desvio padrão do fator MATRIZ para a variável  
LATÊNCIA-VBR (em  $\mu$ s)

<i>FATOR DISTRIBUIÇÃO</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Interligação PABX	107,908	25,780
Videoconferência	4179,833	4800,062
Interlig. redes locais	4310,721	4842,093

Tabela A.22 – média e desvio padrão do fator DISTRIBUIÇÃO para a variável  
LATÊNCIA-VBR (em  $\mu$ s)

### **Tabelas de média e desvio padrão para a métrica Latência ABR por fator**

<i>FATOR PRIORIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Com prioridade	2868,617	4378,972
Sem prioridade	2868,311	4383,400

Tabela A.23 – média e desvio padrão do fator PRIORIDADE para a variável  
LATÊNCIA-ABR (em  $\mu$ s)

<i>FATOR</i> <i>CAPACIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
400 células	1683,036	2152,343
1000 células	4053,892	5555,739

Tabela A.24 – média e desvio padrão do fator CAPACIDADE para a variável LATÊNCIA-ABR (em  $\mu\text{s}$ )

<i>FATOR</i> <i>INTENSIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Normal	203,489	100,603
Intenso	5533,439	4875,393

Tabela A.25 – média e desvio padrão do fator INTENSIDADE para a variável LATÊNCIA-ABR (em  $\mu\text{s}$ )

<i>FATOR</i> <i>MATRIZ</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Memória compart.	2784,342	4306,704
Interconexão total	3026,375	4629,730
Múltiplos estágios	2794,675	4289,262

Tabela A.26 – média e desvio padrão do fator MATRIZ para a variável LATÊNCIA-ABR (em  $\mu\text{s}$ )

<i>FATOR DISTRIBUIÇÃO</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Interligação PABX	106,650	25,647
Videoconferência	4186,675	4797,424
Interlig. redes locais	4312,067	4843,132

Tabela A.27 – média e desvio padrão do fator DISTRIBUIÇÃO para a variável LATÊNCIA-ABR (em  $\mu$ s)

**Tabelas de média e desvio padrão para a métrica Latência UBR por fator**

<i>FATOR PRIORIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Com prioridade	2856,200	4319,524
Sem prioridade	2852,875	4322,520

Tabela A.28 – média e desvio padrão do fator PRIORIDADE para a variável LATÊNCIA-UBR (em  $\mu$ s)

<i>FATOR CAPACIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
400 células	1718,331	2203,536
1000 células	3990,744	5461,795

Tabela A.29 – média e desvio padrão do fator CAPACIDADE para a variável LATÊNCIA-UBR (em  $\mu$ s)

<i>FATOR INTENSIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Normal	203,139	101,938
Intenso	5505,936	4782,316

Tabela A.30 – média e desvio padrão do fator INTENSIDADE para a variável  
LATÊNCIA-UBR (em  $\mu$ s)

<i>FATOR MATRIZ</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Memória compart.	2804,983	4321,456
Interconexão total	2957,029	4462,421
Múltiplos estágios	2801,600	4268,592

Tabela A.31 – média e desvio padrão do fator MATRIZ para a variável  
LATÊNCIA-UBR (em  $\mu$ s)

<i>FATOR DISTRIBUIÇÃO</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Interligação PABX	105,800	25,635
Videoconferência	4165,088	4727,839
Interlig. redes locais	4292,725	4757,919

Tabela A.32 – média e desvio padrão do fator DISTRIBUIÇÃO para a variável  
LATÊNCIA-UBR (em  $\mu$ s)

**Tabelas de média e desvio padrão para a métrica Taxa de Descarte Geral por fator**

<i>FATOR PRIORIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Com prioridade	19,1267	27,7140
Sem prioridade	19,0961	27,6816

Tabela A.33 – média e desvio padrão do fator PRIORIDADE para a variável TAXA-DESCARTE (em %)

<i>FATOR CAPACIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
400 células	19,2400	27,8785
1000 células	18,9828	27,5153

Tabela A.34 – média e desvio padrão do fator CAPACIDADE para a variável TAXA-DESCARTE (em %)

<i>FATOR INTENSIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Normal	0,0000	0,0000
Intenso	38,2228	27,9817

Tabela A.35 – média e desvio padrão do fator INTENSIDADE para a variável TAXA-DESCARTE (em %)

<i>FATOR MATRIZ</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Memória compart.	18,0433	26,3686
Interconexão total	20,1921	29,3670
Múltiplos estágios	19,0988	27,8340

Tabela A.36 – média e desvio padrão do fator MATRIZ para a variável TAXA-DESCARTE (em %)

<i>FATOR DISTRIBUIÇÃO</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Interligação PABX	0,0000	0,0000
Videoconferência	25,5513	26,1951
Interlig. redes locais	31,7829	32,5074

Tabela A.37 – média e desvio padrão do fator DISTRIBUIÇÃO para a variável TAXA-DESCARTE (em %)

### **Tabelas de média e desvio padrão para a métrica Taxa de Descarte CBR por fator**

<i>FATOR PRIORIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Com prioridade	0,2286	0,4172
Sem prioridade	0,2702	0,4922

Tabela A.38 – média e desvio padrão do fator PRIORIDADE para a variável TAXA-DESCARTE -CBR (em %)

<i>FATOR</i> <i>CAPACIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
400 células	0,3175	0,5302
1000 células	0,1813	0,3558

Tabela A.39 – média e desvio padrão do fator CAPACIDADE para a variável TAXA-DESCARTE -CBR (em %)

<i>FATOR</i> <i>INTENSIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Normal	0,0000	0,0000
Intenso	0,4988	0,5378

Tabela A.40 – média e desvio padrão do fator INTENSIDADE para a variável TAXA-DESCARTE -CBR (em %)

<i>FATOR MATRIZ</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Memória compart.	0,0975	0,1928
Interconexão total	0,4262	0,6428
Múltiplos estágios	0,2245	0,3566

Tabela A.41 – média e desvio padrão do fator MATRIZ para a variável TAXA-DESCARTE -CBR (em %)

<i>FATOR</i> <i>DISTRIBUIÇÃO</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Interligação PABX	0,0000	0,0000
Videoconferência	0,4387	0,5777
Interlig. redes locais	0,3095	0,4417

Tabela A.42 – média e desvio padrão do fator DISTRIBUIÇÃO para a variável TAXA-DESCARTE -CBR (em %)

### **Tabelas de média e desvio padrão para a métrica Taxa de Descarte VBR por fator**

<i>FATOR PRIORIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Com prioridade	5,5180	9,3747
Sem prioridade	5,4083	9,1645

Tabela A.43 – média e desvio padrão do fator PRIORIDADE para a variável TAXA-DESCARTE -VBR (em %)

<i>FATOR</i> <i>CAPACIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
400 células	5,5652	9,4556
1000 células	5,3611	9,0802

Tabela A.44 – média e desvio padrão do fator CAPACIDADE para a variável TAXA-DESCARTE -VBR (em %)

<i>FATOR INTENSIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Normal	0,0000	0,0000
Intenso	10,9263	10,5111

Tabela A.45 – média e desvio padrão do fator INTENSIDADE para a variável TAXA-DESCARTE -VBR (em %)

<i>FATOR MATRIZ</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Memória compart.	5,3141	9,0161
Interconexão total	5,6887	9,7551
Múltiplos estágios	5,3866	9,2208

Tabela A.46 – média e desvio padrão do fator MATRIZ para a variável TAXA-DESCARTE -VBR (em %)

<i>FATOR DISTRIBUIÇÃO</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Interligação PABX	0,0000	0,0000
Videoconferência	12,4075	12,6979
Interlig. redes locais	3,9820	4,0705

Tabela A.47 – média e desvio padrão do fator DISTRIBUIÇÃO para a variável TAXA-DESCARTE -VBR (em %)

**Tabelas de média e desvio padrão para a métrica Taxa de Descarte ABR por fator**

<i>FATOR PRIORIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Com prioridade	11,7208	17,0791
Sem prioridade	11,8061	17,1598

Tabela A.48 – média e desvio padrão do fator PRIORIDADE para a variável TAXA-DESCARTE -ABR (em %)

<i>FATOR CAPACIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
400 células	11,7644	17,1299
1000 células	11,7625	17,1091

Tabela A.49 – média e desvio padrão do fator CAPACIDADE para a variável TAXA-DESCARTE -ABR (em %)

<i>FATOR INTENSIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Normal	0,0000	0,0000
Intenso	23,5269	17,3635

Tabela A.50 – média e desvio padrão do fator INTENSIDADE para a variável TAXA-DESCARTE -ABR (em %)

<i>FATOR MATRIZ</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Memória compart.	11,7941	17,2385
Interconexão total	11,7004	17,1988
Múltiplos estágios	11,7958	17,2917

Tabela A.51 – média e desvio padrão do fator MATRIZ para a variável TAXA-DESCARTE -ABR (em %)

<i>FATOR DISTRIBUIÇÃO</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Interligação PABX	0,0000	0,0000
Videoconferência	15,1816	15,5152
Interlig. redes locais	20,1087	20,5426

Tabela A.52 – média e desvio padrão do fator DISTRIBUIÇÃO para a variável TAXA-DESCARTE -ABR (em %)

### **Tabelas de média e desvio padrão para a métrica Taxa de Descarte UBR por fator**

<i>FATOR PRIORIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Com prioridade	15,7541	22,7020
Sem prioridade	15,7383	22,6865

Tabela A.53 – média e desvio padrão do fator PRIORIDADE para a variável TAXA-DESCARTE -UBR (em %)

<i>FATOR</i> <i>CAPACIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
400 células	15,6155	22,5134
1000 células	15,8769	22,8728

Tabela A.54 – média e desvio padrão do fator CAPACIDADE para a variável TAXA-DESCARTE -UBR (em %)

<i>FATOR</i> <i>INTENSIDADE</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Normal	0,0000	0,0000
Intenso	31,4925	22,8035

Tabela A.55 – média e desvio padrão do fator INTENSIDADE para a variável TAXA-DESCARTE -UBR (em %)

<i>FATOR MATRIZ</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Memória compart.	16,0016	23,2245
Interconexão total	15,4191	22,3884
Múltiplos estágios	15,8179	22,9494

Tabela A.56 – média e desvio padrão do fator MATRIZ para a variável TAXA-DESCARTE -UBR (em %)

<i>FATOR DISTRIBUIÇÃO</i>	<i>MÉDIA</i>	<i>DESVIO PADRÃO</i>
Interligação PABX	0,0000	0,0000
Videoconferência	21,7637	22,2422
Interlig. redes locais	25,4750	26,0287

Tabela A.57 – média e desvio padrão do fator DISTRIBUIÇÃO para a variável  
TAXA-DESCARTE -UBR (em %)

## ANEXO B – DESCRIÇÃO DOS MODELOS

Neste anexo serão descritos os modelos criados no software ARENA, que serviram de base para a simulação do funcionamento de um comutador ATM. Os seis modelos estão aqui divididos em quatro segmentos, a fim de ilustrar suas diferenças : o pré-processamento, a entrada na matriz de comutação, a matriz de comutação propriamente dita, e o pós-processamento.

É importante salientar que o comutador modelado possui 4 interfaces, o que determina a existência de 4 entradas e 4 saídas. Este número pode ser modificado alterando-se o número de blocos ARRIVE (que representam as chegadas) e as seqüências de saída no pós-processamento.

Os atributos e variáveis que têm um destaque maior, e que portanto serão mencionados no texto são :

- ◆ *Atributo Saídas* - Nos modelos com matriz por memória compartilhada ou meio compartilhado, este atributo é um vetor de 4 posições (uma para cada saída), inicializado com zero em todas elas. No segmento de pré-processamento, as saídas às quais a célula se destina têm o seu valor alterado para 1. Nos modelos com matriz de múltiplos estágios, este atributo é uma matriz 4x3, onde as linhas representam as saídas. A primeira coluna funciona com o vetor dos outros modelos, e as colunas 2 e 3 são usadas nos nós para decidir para qual saída do nó a célula deve ser encaminhada : 0 para a saída superior, e 1 para a saída inferior.
- ◆ *Atributo Tag* - No modelo com matriz de múltiplos estágios, este atributo recebe o caminho que a célula vai fazer através da matriz, com informações extraídas do atributo Saídas.
- ◆ *Atributo CLP* - CLP significa Cell Loss Priority (Prioridade de Perda de Células). Quando este atributo tem o valor 1, a célula tem "menos valor", e portanto será descartada antes de uma célula com CLP=0. As chances de que a célula tenha o atributo CLP igual a 1 depende principalmente da classe de tráfego à qual ela pertence.

- ◆ *Variável limite* - Indica o número máximo de células que os buffers de saída têm capacidade de guardar. Quando este limite é atingido, as células seguintes provocam descarte.

## PRÉ-PROCESSAMENTO

Neste segmento o fluxo se dá da seguinte forma :

A entidade entra no sistema por uma das quatro interfaces do comutador (blocos ARRIVE). Primeiramente, verifica-se se a entidade é uma célula ou um quadro (bloco CHOOSE). No primeiro caso, é feita uma segunda verificação, determinando se é uma célula de controle ou de dados (bloco CHOOSE). As células de controle são processadas no servidor CACSM antes de continuarem. Às células de dados são aplicadas as regras de controle de tráfego (bloco INSPECT) e caso seja uma célula não-conforme, ela passa pelo bloco ASSIGN, que muda o valor do seu atributo CLP para 1.

Já os quadros que chegam entram num processo repetitivo (bloco WHILE), que, de acordo com o seu tamanho original, divide-o em um certo número de células.

As entidades oriundas destes 3 processos (células de controle, de dados e células geradas a partir da divisão do quadro) são repassadas para a etapa de determinação da multiplicidade e translação de endereços. Para a multiplicidade gera-se um número aleatório N entre 1 e 4 (já que são 4 as saídas).

Se  $N=1$ , uma saída é sorteada e marcada no vetor de endereçamento como destino.

Se  $N=4$ , todas as saídas são marcadas no vetor.

Se  $N=2$  ou  $N=3$ , o número correspondente de saídas é sorteado e o vetor é atualizado de acordo.

Definidas as saídas, a entidade passa por um bloco DELAY (simbolizando o tempo usado no processo descrito acima), o tempo perdido no pré-processamento é guardado, e a entidade é enviada para a matriz.

Os modelos com matriz de múltiplos estágios, no entanto, têm um processo adicional. Como a matriz não tem a capacidade de duplicar células, nas situações de

multicast ou broadcast, a geração de cópias das células é feita no próprio segmento de pré-processamento, através de um loop (bloco WHILE), onde são geradas tantas células quantas foram as saídas de destino.

As figuras das próximas páginas apresentam o segmento de pré-processamento dos modelos. A figura B.1 representa este segmento nos modelos de memória compartilhada com e sem prioridade de tráfego e meio compartilhado com e sem prioridade de tráfego. A figura B.2 representa este segmento nos modelos de múltiplos estágios com e sem prioridade de tráfego.

## **ENTRADA NA MATRIZ DE COMUTAÇÃO**

A maneira como se dá a entrada na matriz de comutação depende do fato de haver ou não divisão por prioridade de tráfego. Assim, para os modelos que usam prioridade de tráfego, para cada uma das quatro entradas na matriz existem quatro filas (blocos QUEUE), uma para cada classe de tráfego. Logo após, os blocos QPICK fazem a ponte entre as filas e o recurso (a memória, o meio ou os nós, dependendo da matriz). O primeiro QPICK pega aleatoriamente uma entidade entre as filas CBR, o segundo entre as filas VBR, o terceiro entre as filas ABR e o último entre as filas UBR. Em seguida, os blocos SEIZE determinam que primeiramente as células CBR tomarão o recurso, seguidas pelas células VBR, ABR e UBR. No caso do modelo com matriz de múltiplos estágios, as duas primeiras entradas são para o nó 1, e as duas outras entradas para o nó 2.

Já para os modelos que não usam prioridade de tráfego, as quatro entradas da matriz estão diretamente ligadas ao bloco SEIZE que controla a ocupação do recurso (ou os recursos, no caso do modelo com matriz de múltiplos estágios). Nesta situação, usa-se a própria fila deste bloco.

As figuras B.3 a B.8 mostram a entrada da matriz para os diferentes modelos

## **MATRIZ DE COMUTAÇÃO**

No total são 3 matrizes, já que elas independem de haver ou não divisão por prioridade de tráfego.

Para cada célula que entra na matriz é feita uma cópia. Esta cópia é usada para cálculo da taxa de descarte, como mostra a área em azul nas figuras B.9, B.10 e B.12. A seguir, uma breve descrição de cada matriz.

### ***Matriz com memória compartilhada***

Nesta matriz, são colocadas na memória tantas cópias da célula quantas forem as saídas de destino (loop WHILE). Como a própria matriz já atua como buffer das saídas, no processo de escrita já verifica-se se a memória está cheia. Se estiver, procede-se ao descarte. Caso contrário, a célula é enviada para a saída correspondente.

O descarte é feito da seguinte forma: se a célula tem o atributo CLP=1, ela é imediatamente descartada. Senão (atributo CLP=0), procura-se no buffer (no caso, a memória) - bloco SEARCH - uma célula que tenha o atributo CLP=1. Se for achada uma nesta condição, ela é descartada. Se todas as células do buffer tiverem CLP=0, descarta-se a célula atual.

Esta matriz é mostrada na figura B.9.

### ***Matriz com meio compartilhado***

Todas as células que entram na matriz são jogadas para todas as saídas (bloco DUPLICATE e os 4 blocos LEAVE).

Cada saída, então, verifica se a célula é destinada para si. Em caso negativo, a célula é descartada. Em caso positivo, a célula é duplicada. Uma cópia vai para o cálculo da taxa de descarte, e a outra é direcionada para o buffer. Se o buffer estiver cheio (verificado pelo bloco CHOOSE), procede-se ao descarte, como descrito na matriz anterior. Se não estiver, a célula é nele colocada, sem outros processamentos. Do buffer, a célula é enviada, então, para o segmento de pós-processamento.

Esta matriz é mostrada na figura B.10.

### ***Matriz com múltiplos estágios***

De uma forma aleatória, a célula entra no primeiro estágio pelo nó 1 ou pelo nó 2. Uma vez num dos nós, verifica-se o atributo TAG para decidir se a célula deve ser encaminhada para o segundo estágio pelo nó 3 (Tag[1]=0) ou pelo nó 4 (Tag[1]=1).

Nestes nós, mais uma vez o atributo Tag é examinado, para decidir se a célula deve ser enviada para a saída superior (Tag[2]=0) ou inferior (Tag[2]=1). Estes processos são mostrados na figura B.11.

Chegando na saída, como nas outras matrizes, uma cópia da célula é enviada para o cálculo da taxa de descarte, e a outra é encaminhada para o buffer. Se o buffer está cheio, há descarte, como descrito na primeira matriz. Caso contrário, a célula entra no buffer sem outros processamentos. Do buffer, a célula é enviada para o segmento de pós-processamento.

Esta etapa é mostrada na figura B.12.

## **PÓS-PROCESSAMENTO**

A célula chega em uma das quatro saídas, e é feito o pós-processamento (bloco SERVER), onde representa-se a remontagem de quadros (se for o caso) e a interação com o meio físico. Depois disso, a célula passa por blocos de contagem e medição de tempo no sistema (blocos COUNT e TALLY), e finalmente sai do comutador. Nos blocos TALLY são calculados os tempos de latência das células no comutador. Estes valores, juntamente com os de taxa de descarte, são os valores utilizados na análise estatística, apresentada no capítulo 5.

Este segmento é igual para todos os modelos, e mostrado na figura B.13.

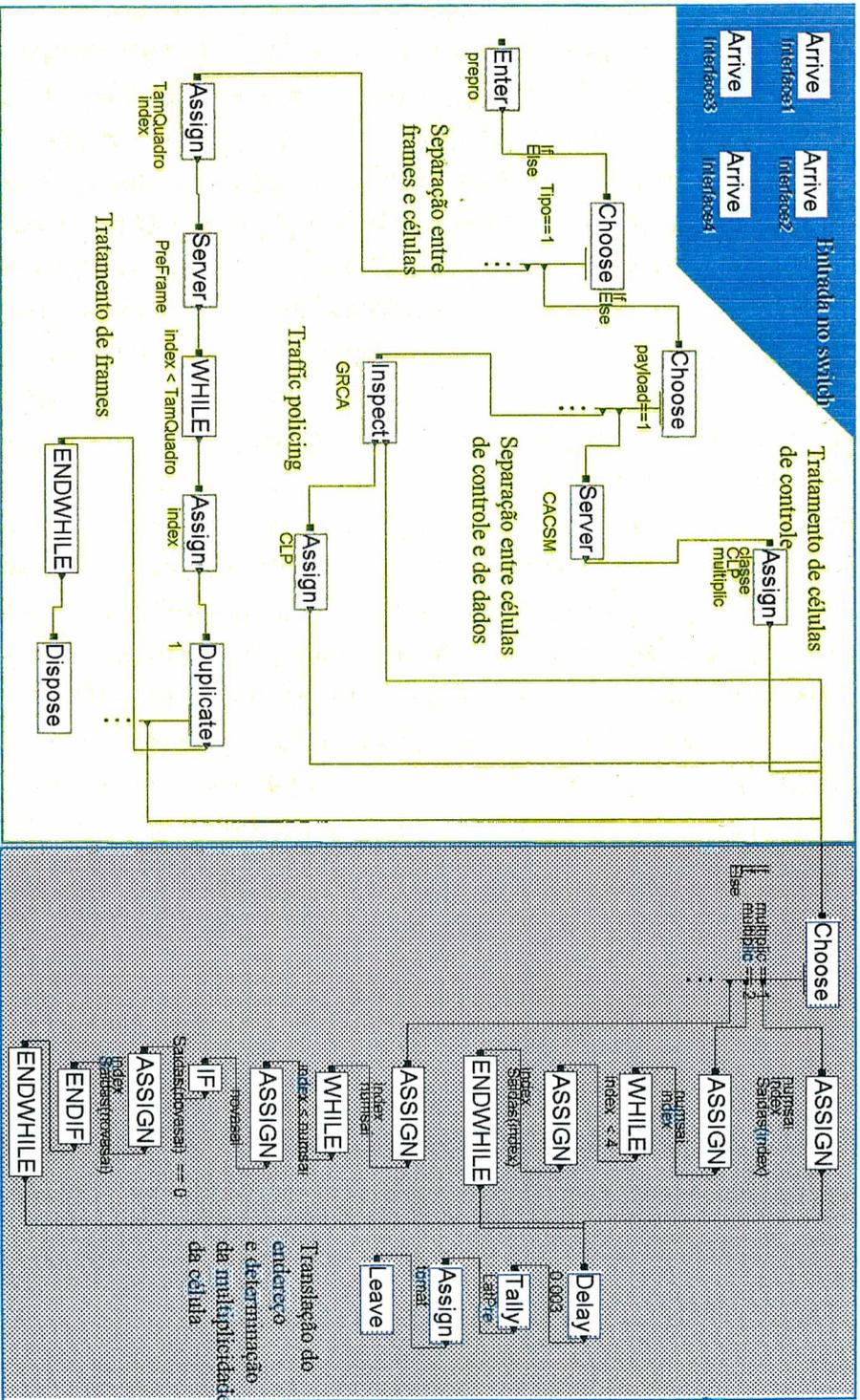


Figura B.1 - Pré-processamento nas matrizes de memória e meio compartilhados

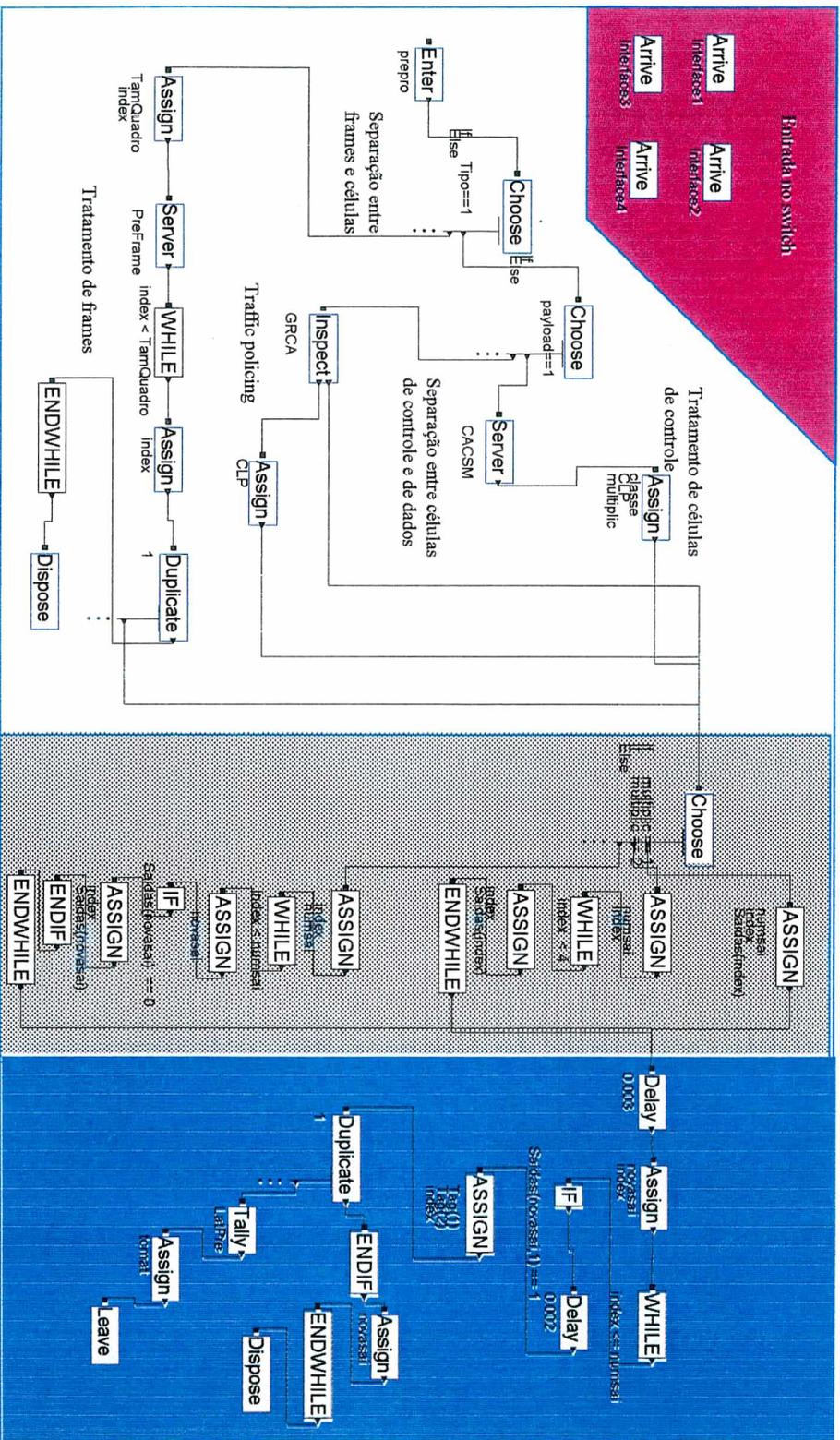


Figura B.2 - Pré-processamento da matriz com múltiplos estágios

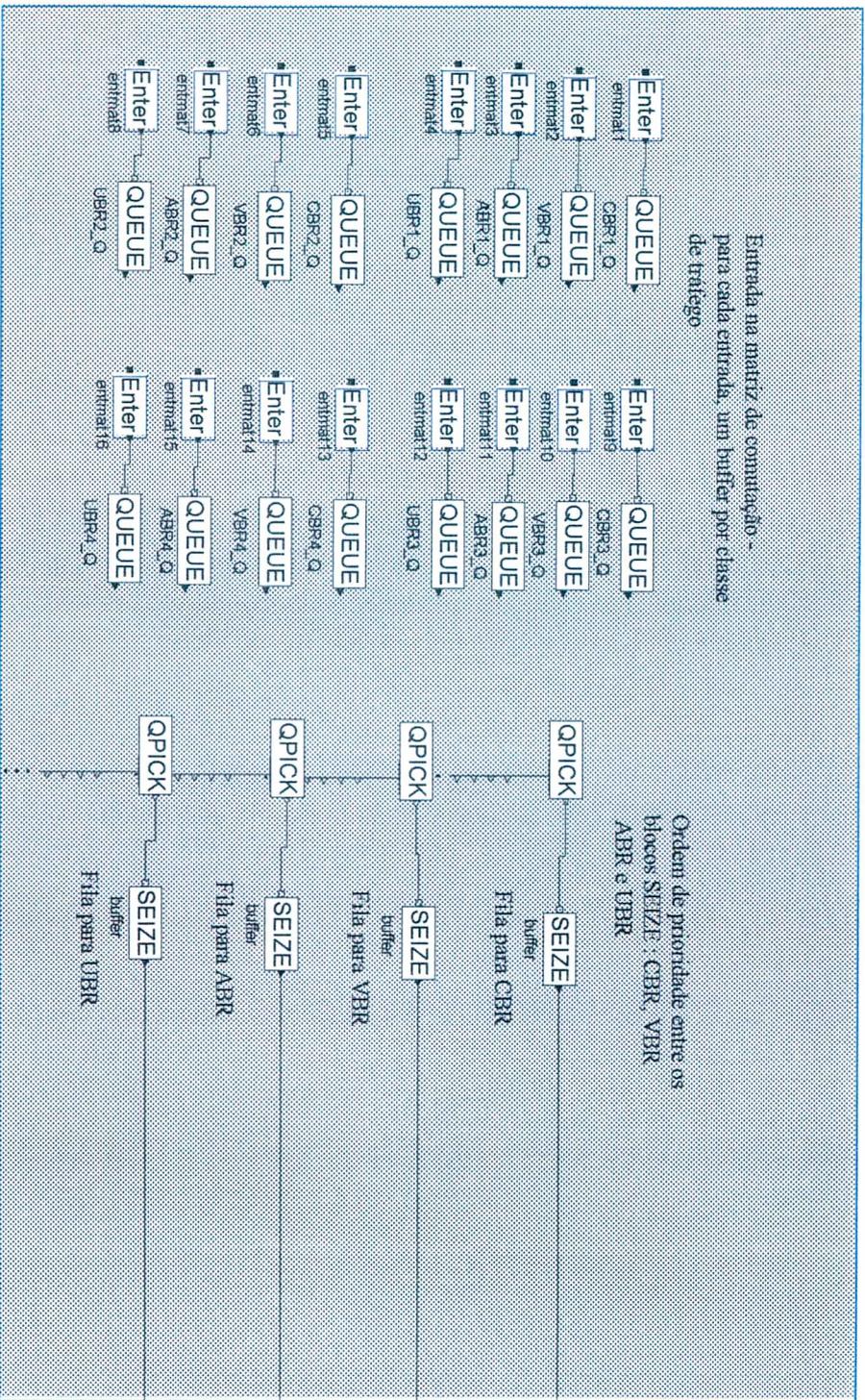


Figura B.3 - Entrada da matriz de memória compartilhada com divisão por prioridade de tráfego

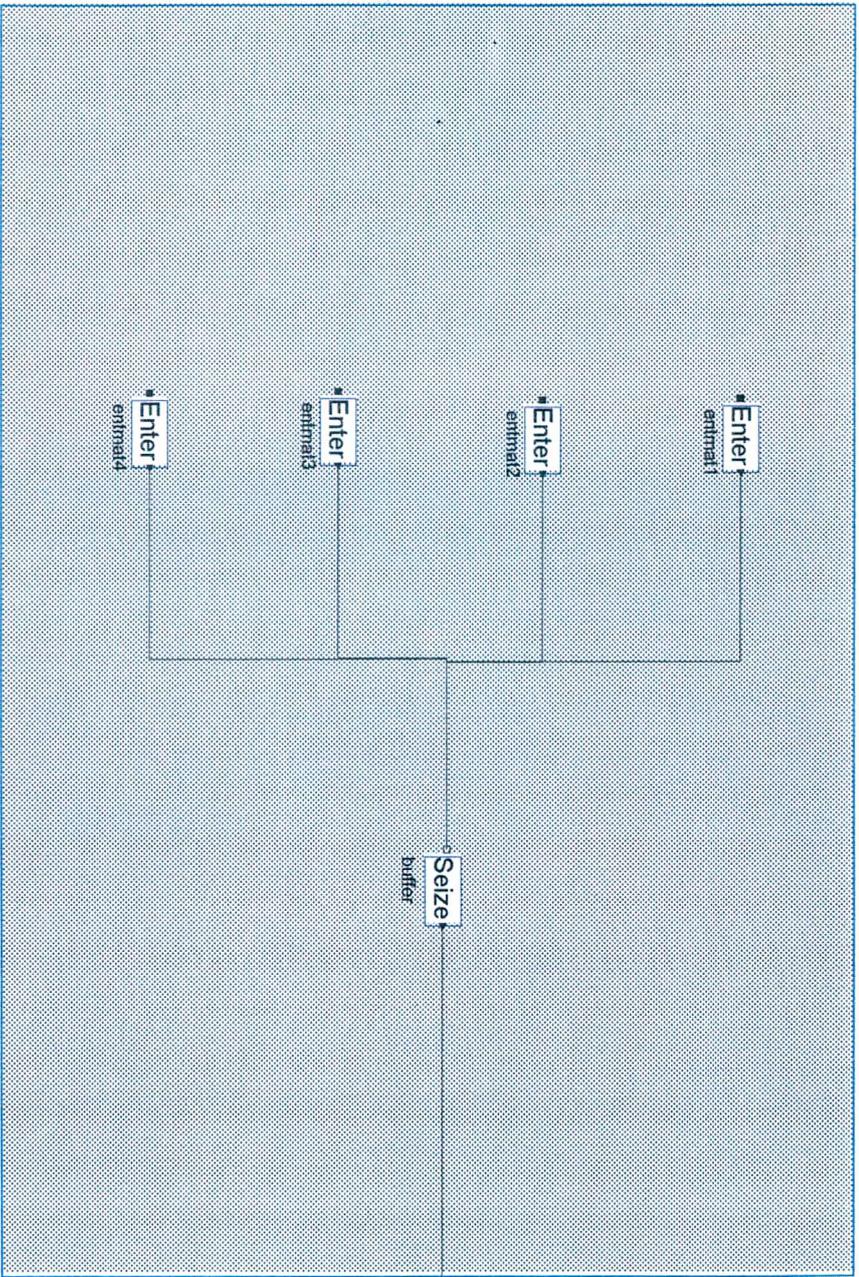


Figura B.4 - Entrada da matriz de memória compartilhada sem divisão por prioridade de tráfego

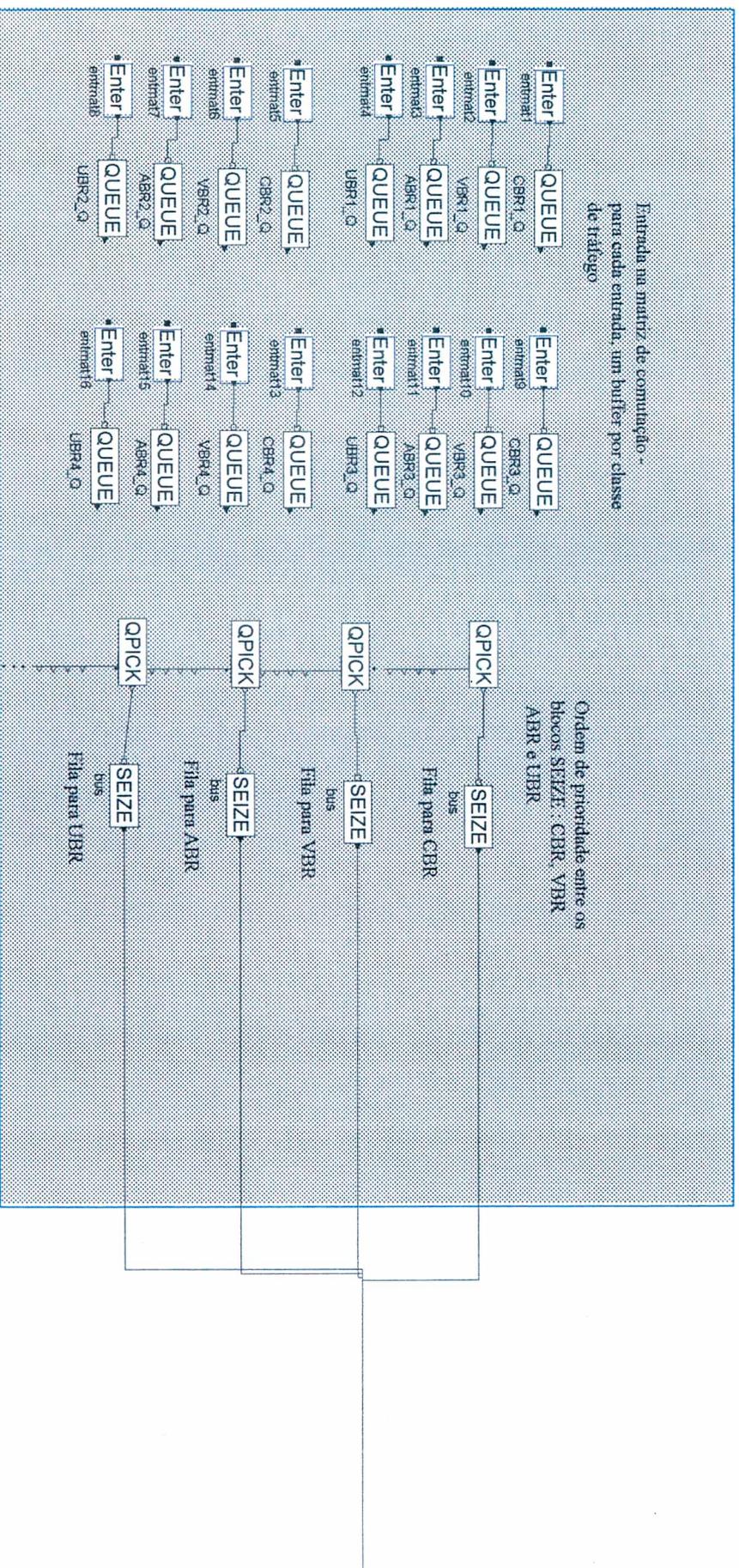


Figura B.5 - Entrada da matriz de meio compartilhado com divisão por prioridade de tráfego

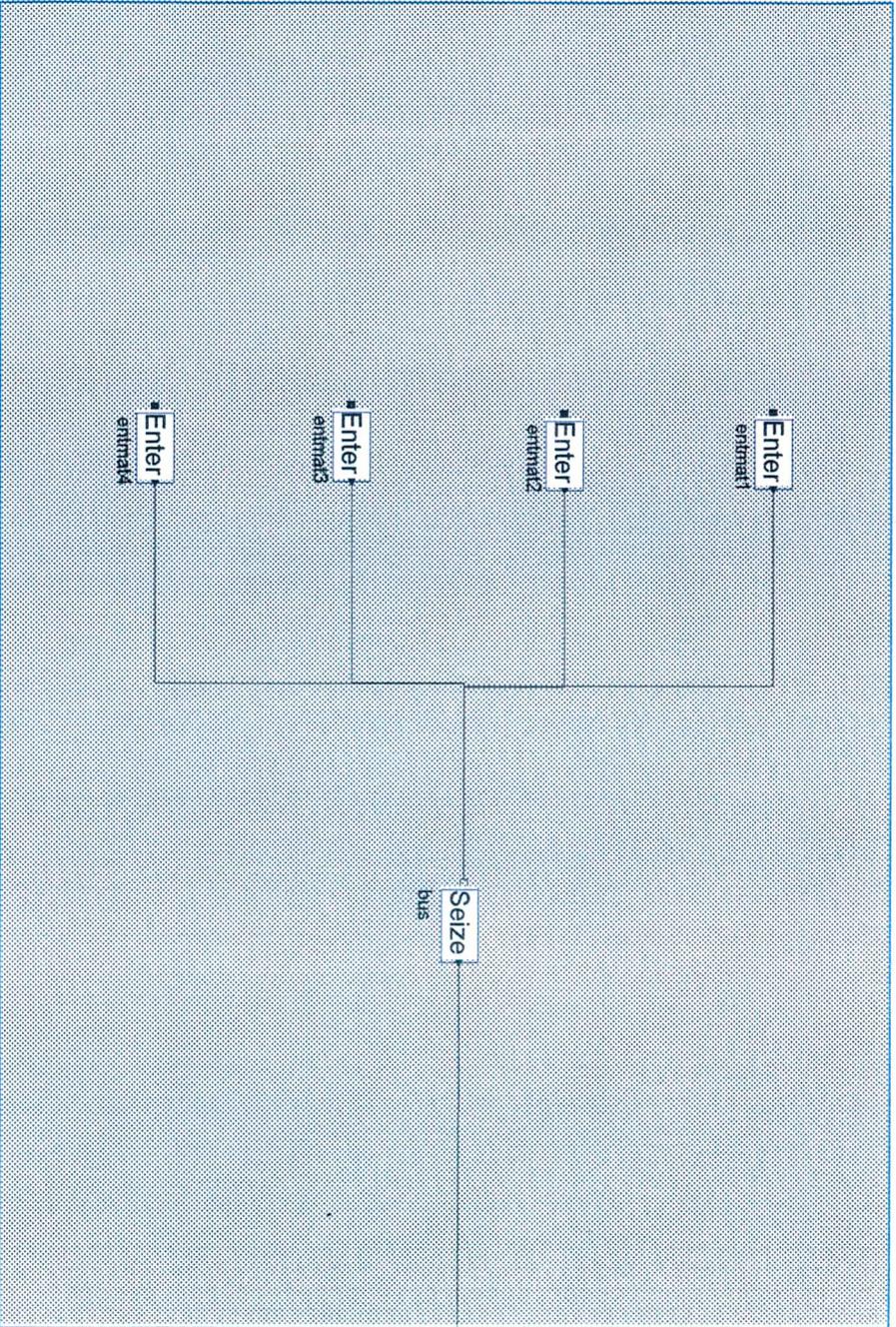


Figura B.6 - Entrada da matriz de meio compartilhado sem divisão por prioridade de tráfego

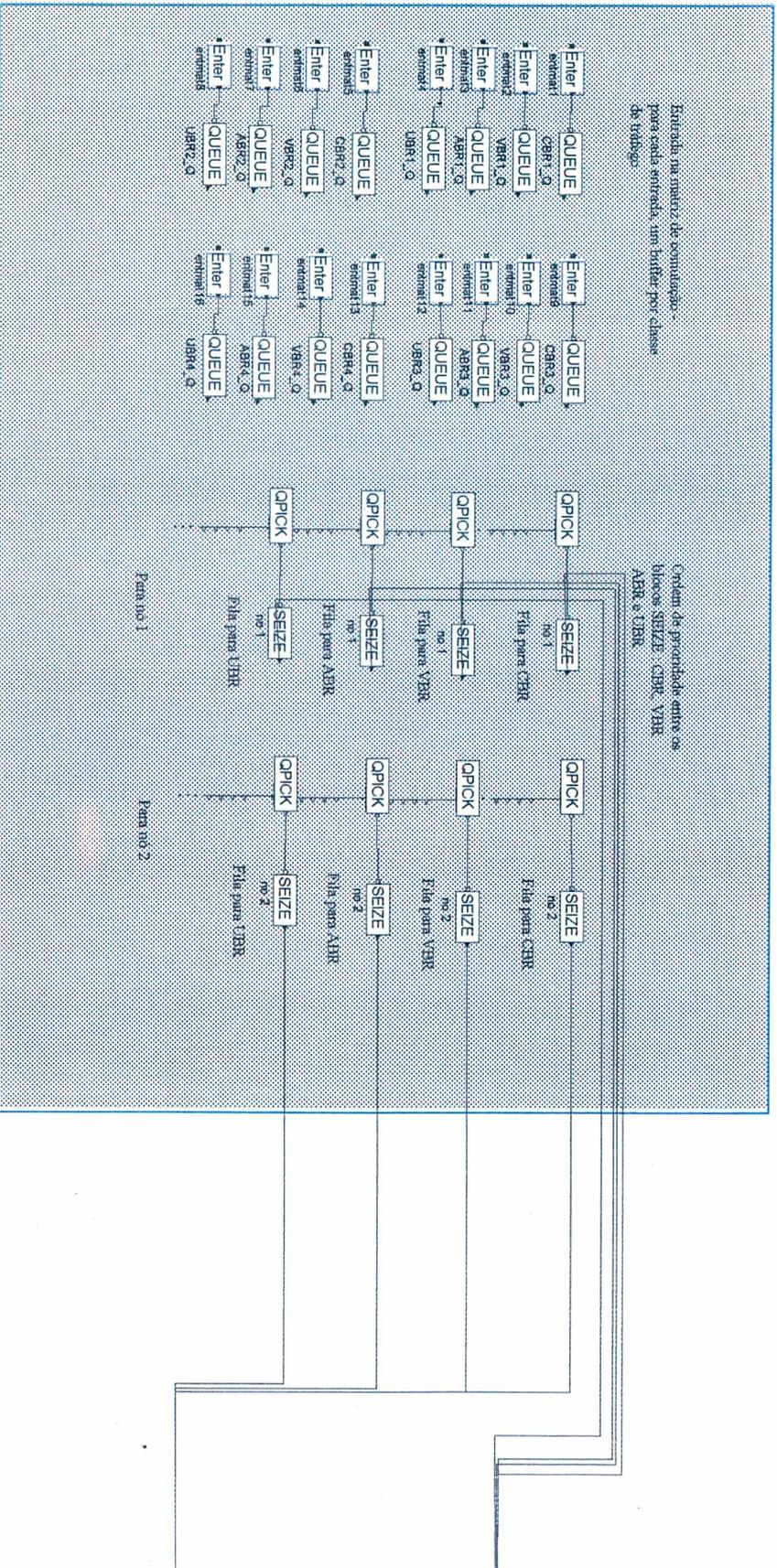


Figura B.7 - Entrada da matriz de múltiplos estágios com divisão por prioridade de tráfego

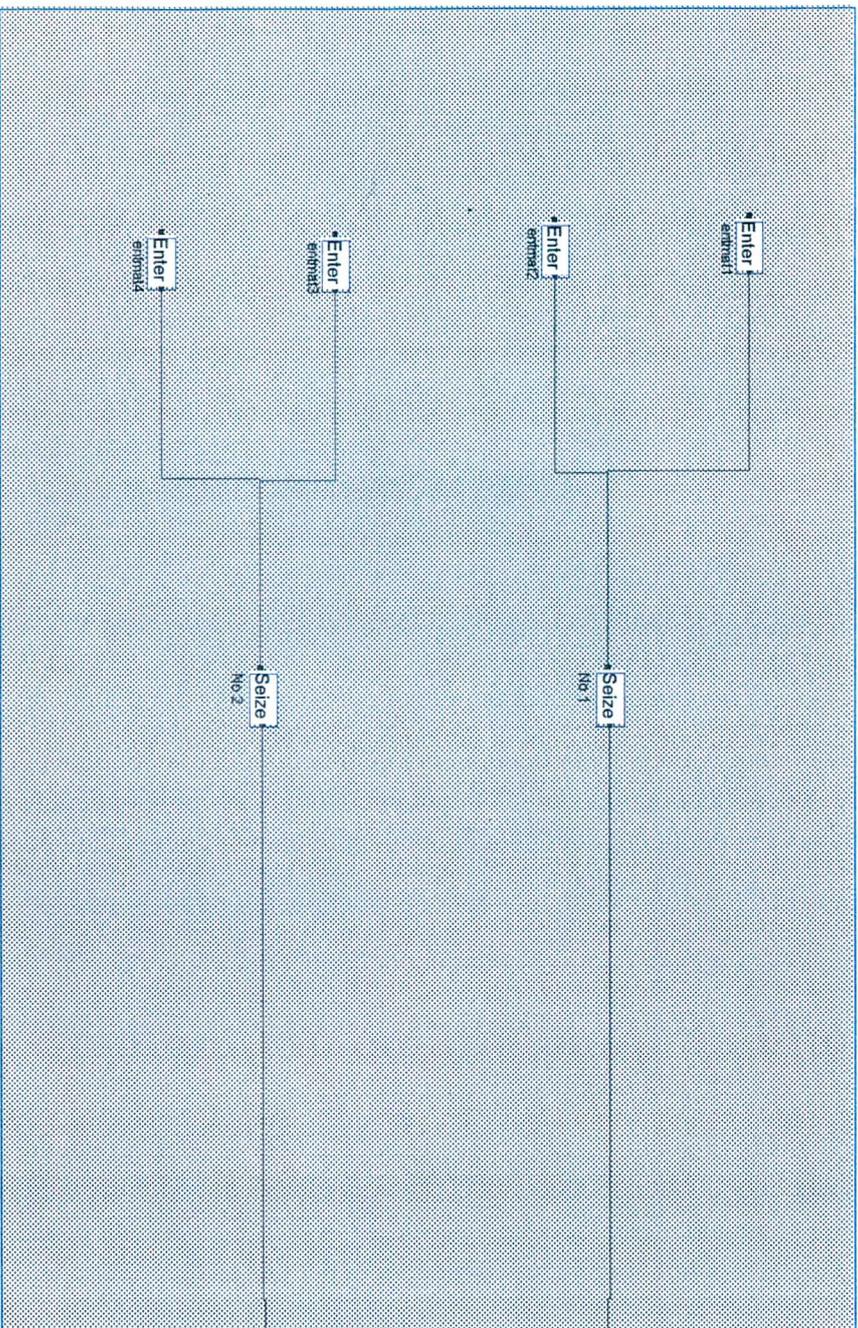


Figura B.8 - Entrada da matriz de múltiplos estágios sem divisão por prioridade de tráfego



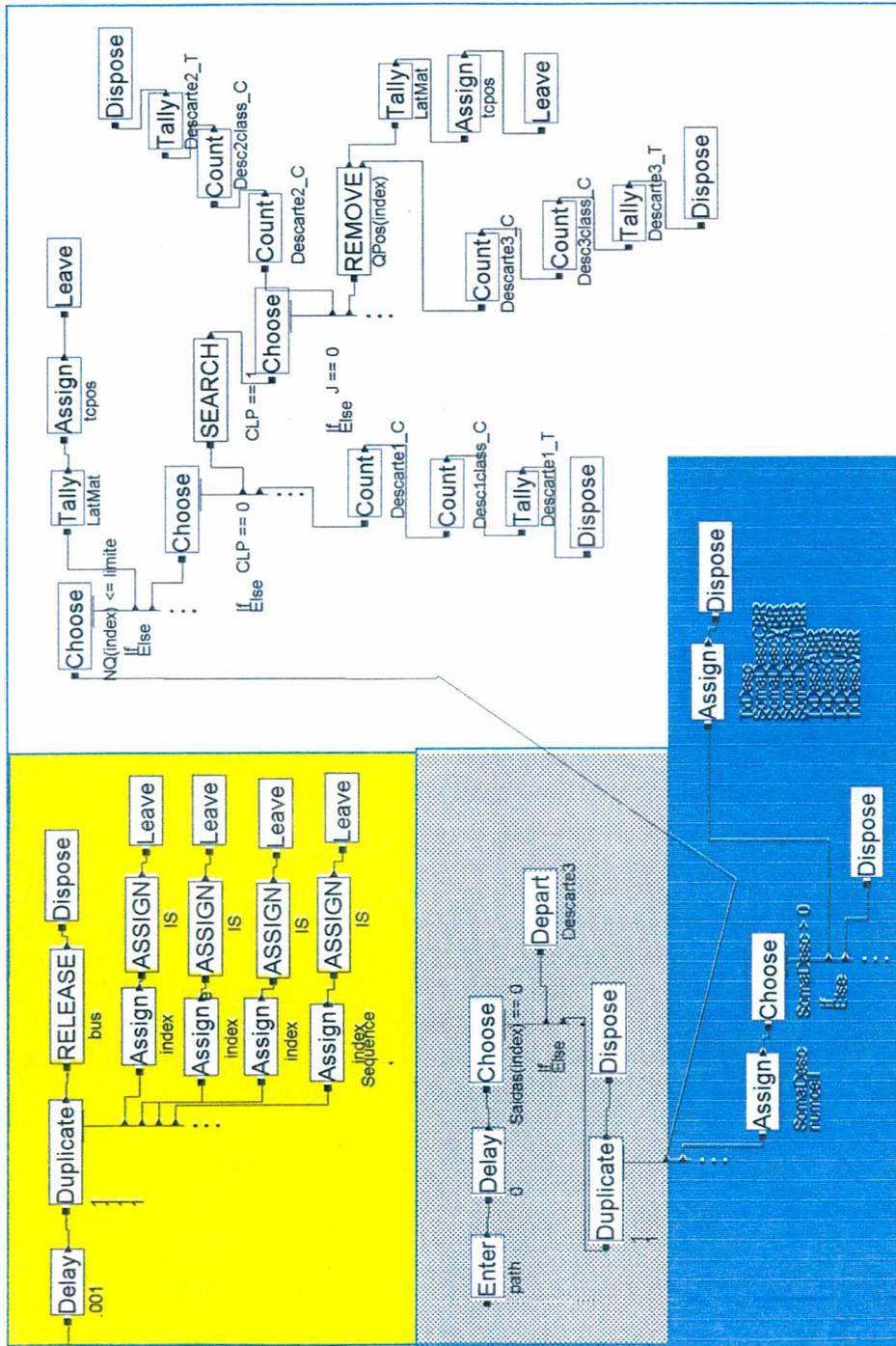


Figura B.10 - Matriz de meio compartilhado

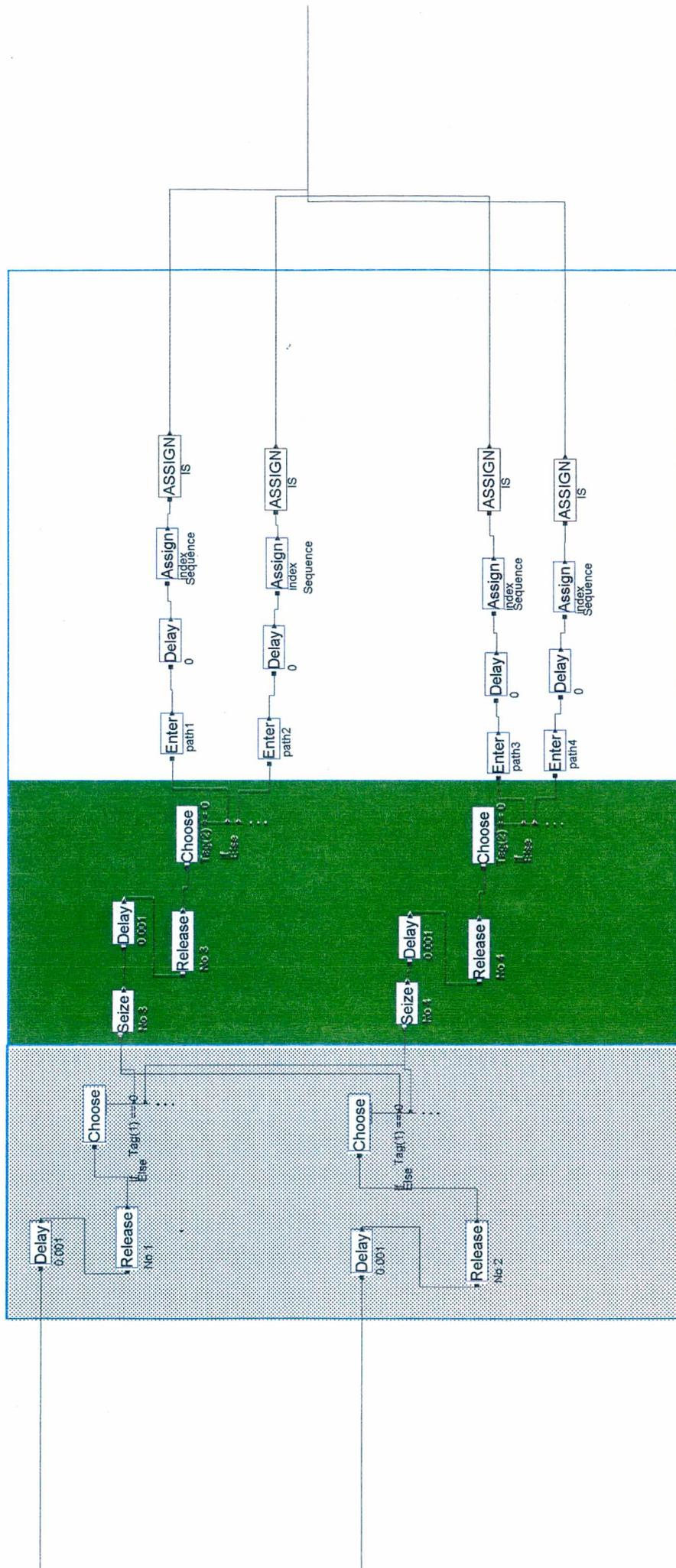


Figura B.11 - Matriz de múltiplos estágios (parte 1)

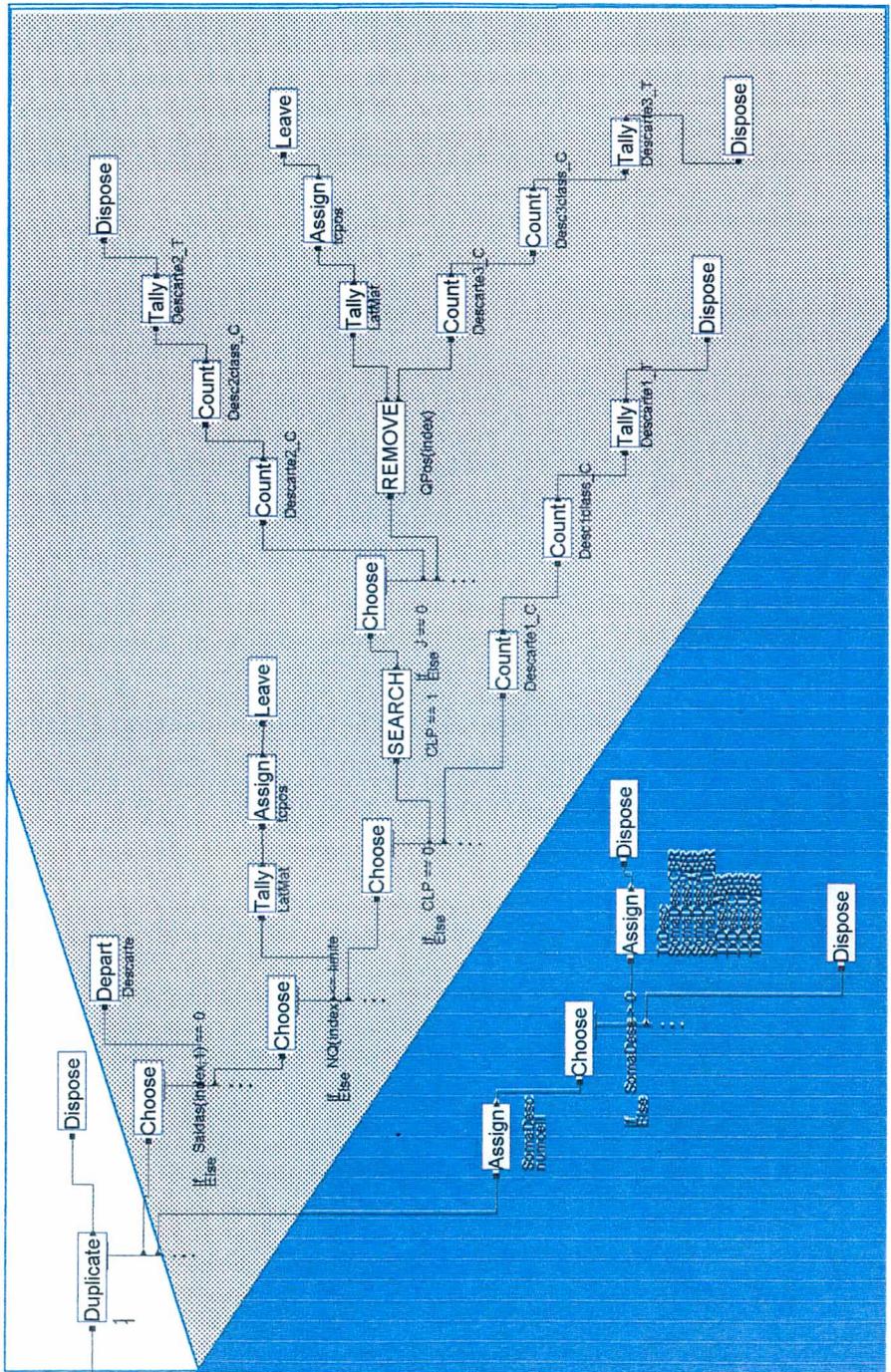


Figura B.12 - Matriz de múltiplos estágios (parte 2)

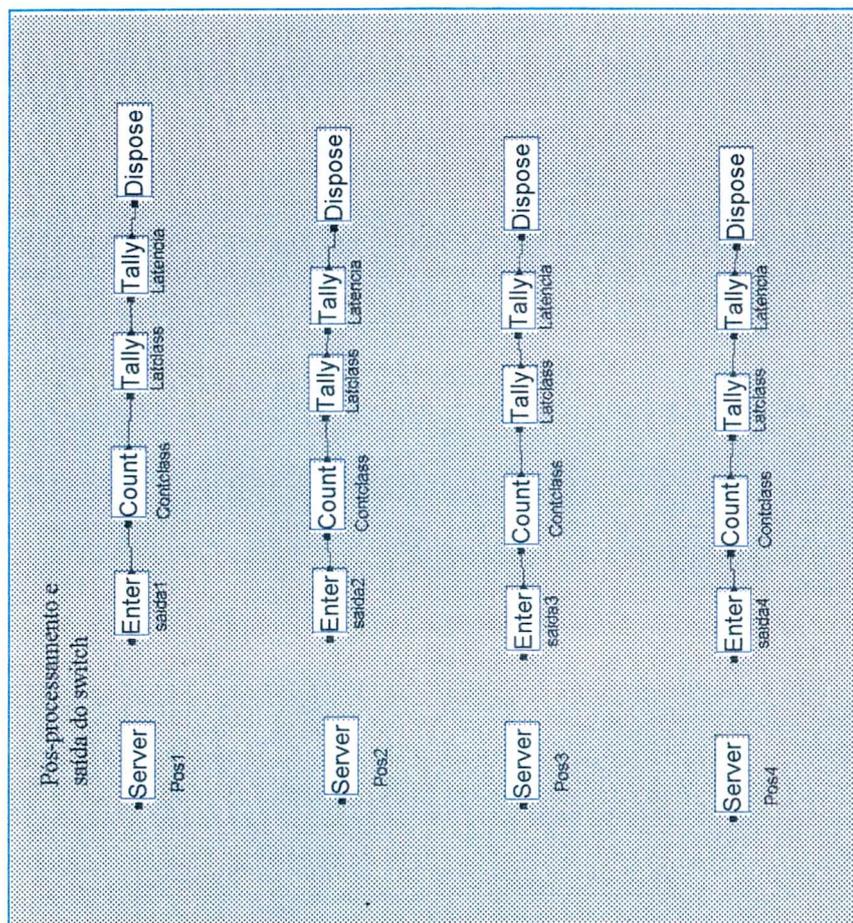


Figura B.13 - Segmento de pós-processamento