

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

**Método para
Planejamento de Capacidade
de Redes ATM baseado em Simulação**

por

ADRIANO RAMOS GONÇALVES

Dissertação submetida à avaliação,
como requisito parcial para a obtenção do grau de Mestre
em Ciência da Computação

Prof. Juergen Rochol
Orientador

Porto Alegre, março de 2001.

CIP - CATALOGAÇÃO NA PUBLICAÇÃO

Gonçalves, Adriano Ramos

Método para planejamento de capacidade de redes ATM baseado em simulação/ por Adriano Ramos Gonçalves. - Porto Alegre: PPGC da UFRGS, 2001.

126p.: il.

Dissertação (mestrado) - Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR-RS, 2001. Orientador: Rochol, Juergen.

1. ATM 2. Planejamento de capacidade 3. Dimensionamento de Recursos 4. Simulação I. Rochol, Juergen. II. Título

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitora: Profa. Wrana Panizzi

Pró-Reitor de Ensino: Prof. José Carlos Ferraz Hennemann

Pró-Reitor Adjunto de Pós-Graduação: Prof. Philippe Olivier Alexandre Navaux

Diretor do Instituto de Informática: Prof. Philippe Olivier Alexandre Navaux

Coordenador do PPGC: Prof. Carlos Alberto Heuser

Bibliotecária-Chefe do Instituto de Informática: Beatriz Regina Bastos Haro

Agradecimentos

Reconhecimento especial à dedicação e compreensão da minha esposa Míriam que é apoio indispensável em minha vida. Aos nossos filhos, Alonso e Lorenzo, que não tiveram escolha quando me debrucei sobre os livros. Aos meus pais, Vilson e Terezinha, por terem desde cedo, através de seus exemplos de vida, me encorajado em ser esforçado e justo. Ao meu irmão, Fabiano, pelo suporte computacional. Aos amigos que acompanharam meu envolvimento e torceram por mim.

E especialmente ao Professor Juergen Rochol pela paciência, disponibilidade e boa vontade em auxiliar a encontrar o melhor caminho para desenvolver o trabalho.

Agradeço também à Companhia de Processamento de Dados do Município de Porto Alegre – PROCEMPA – pela compreensão e facilidades oferecidas para que eu pudesse aproveitar a oportunidade de realizar este trabalho.

“O que nós somos é um presente da vida
para nós. O que nós seremos é um
presente que damos à vida”
Herbert de Souza (Betinho)

Sumário

Lista de Abreviaturas.....	6
Lista de Figuras	8
Lista de Tabelas.....	11
Resumo	12
Abstract.....	13
1 Introdução	14
1.1 Rede Multisserviços.....	14
1.2 Motivação.....	16
1.3 Objetivos	17
1.4 Estrutura da Dissertação.....	18
2 Desempenho em Redes ATM.....	19
2.1 Sistemas Eficientes	19
2.2 QoS e Capacidade de Atendimento em Redes de Comunicação.....	20
2.3 Princípios de Redes ATM.....	22
2.3.1 Multiplexação de Células	22
2.3.2 Contrato de Tráfego.....	24
2.3.3 Descritor de Tráfego da Conexão.....	25
2.3.4 Categorias de Serviços.....	26
2.3.5 Definição de Conformidade.....	29
2.4 Qualidade de Serviço (QoS) - Medidas de Desempenho.....	31
2.4.1 CLR - Razão de Células Perdidas.....	32
2.4.2 CTD - Atraso de Transferência de Célula	32
2.4.3 CDV - Variação do Atraso de Célula	36
2.4.4 CER - Razão de Células Erradas	39
2.4.5 CMR - Taxa de Células Inseridas Incorretamente.....	39
2.4.6 SECBR - Razão de Blocos de Células Severamente Errados	40
2.5 Controle de Tráfego - TM (<i>Traffic Management</i>)	40
2.5.1 CAC - Controle de Admissão de Conexões e NRM - Controle de Recursos da Rede	41
2.5.2 UPC/NPC - Policiamento de Tráfego.....	43
2.5.3 Formatação de Tráfego (<i>Traffic Shaping</i>)	45
3 Planejamento de Capacidade de Redes ATM.....	46
3.1 Relação Carga, Capacidade e Desempenho	46
3.2 Fases do Planejamento de Capacidade	47
3.3 Método para Dimensionamento de Recursos na Rede ATM.....	50
3.4 Modos de Alocação de Recursos	53

3.4.1	Alocação de Banda pela Taxa de Pico	54
3.4.2	Alocação de Banda pela Taxa Mínima.....	55
3.4.3	Alocação pela Banda Equivalente	55
3.5	Métodos de Alocação de Recurso por Categorias de Serviços	57
3.5.1	Alocação de Recursos para Tráfego CBR	58
3.5.2	Alocação de Recursos para Tráfego ABR.....	60
3.5.3	Alocação de Recursos para Tráfego UBR.....	61
3.5.4	Alocação de Recursos para Tráfego VBR.....	62
4	Simulação para Dimensionamento de Capacidade	64
4.1	Simulação: Vantagens e Limitações	64
4.2	Simulador ATM-HFC.....	65
4.2.1	Conceitos do Simulador	66
4.2.2	Componentes do Simulador.....	67
4.3	Método da Simulação.....	70
4.3.1	Estimação do Intervalo de Confiança: Método de Subintervalos	73
4.3.2	Estimativa de CTD	78
4.3.3	Estimativa de CDV	80
4.3.4	Estimativa de CLR	84
4.4	Simulações para Determinar a Banda Equivalente de Tráfego VBR.....	86
4.4.1	Caracterização do tráfego para o Simulador.....	86
4.4.2	Experimentos com diferentes configurações de <i>buffer</i>	88
4.4.3	Resultados Obtidos	89
4.5	Dimensionamento de Capacidade conforme Resultados Obtidos	96
4.6	Validação dos Resultados	98
5	Conclusões e Trabalhos Futuros	100
5.1	Conclusões.....	100
5.2	Trabalhos Futuros.....	102
	Anexo 1 Parâmetros de QoS Genéricos	103
	Anexo 2 Execução do Simulador	104
	Anexo 3 Relação dos Programas Alterados no NIST ATM/HFC	109
	Anexo 4 <i>Scripts</i> e Arquivos utilizados na Simulação	112
	Bibliografia.....	123

Lista de Abreviaturas

ABR	Available Bit Rate
ALN	Adaptive Logic Network
ATM	Asynchronous Transfer Mode
BECN	Backward Explicit Congestion Notification
B-ISDN	Broadband Integrated Services Digital Network
BSS	Cross-Broadband Switching Systems
BT	Burst Tolerance
BTE	Broadband Terminal Equipment
CAC	Connection Admission Control
CBR	Constant Bit Rate
CDV	Cell Delay Variation
CDVT	Cell Delay Variation Tolerance
CER	Cell Error Ratio
CLP	Cell Loss Priority
CLR	Cell Loss Ratio
CMR	Cell Misinsertion Rate
CTD	Cell Transfer Delay
DIFFSERV	Differentiated Services
EFCI	Explicit Forward Congestion Indication
EPD	Early Packet Discard
ER	Explicit Rate
FECN	Forward Explicit Congestion Notification
GCRA	Generic Cell Rate Algorithm
GFC	Generic Flow Control
HDTV	High-Definition TV
HEC	Header Error Check
HFC	Hybrid Fiber Coax
ITU-T	International Telecommunications Union - Telecommunication
LAN	Local Area Network
MaxCTD	Maximum Cell Transfer Delay
MBS	Maximum Burst Size
MCR	Minimum Cell Rate
MPEG	Motion Picture Expert Group
MPLS	MultiProtocol Label Switching
NIST	National Institute of Standards and Technology, USA
NNI	Network Network Interface
NPC	Network Parameter Control
NRM	Network Resource Management
NTSC	National Television System Committee
OAM	Operation and Maintenance
PCR	Peak Cell Rate
PDH	Plesiochronous Transmission Hierarchy
PNNI	Private Network-Network Interface
ppCDV	Peak-to-Peak Cell Delay Variation
PTI	Payload Type Indicator

PVC	Permanent Virtual Circuit
QFC	Quantum Flow Control
QoS	Quality of Service
RDSI	Rede Digital de Serviços Integrados
RDSI-FL	Rede Digital de Serviços Integrados - Faixa Larga
SCR	Sustainable Cell Rate
SDH	Synchronous Digital Hierarchy
SECB	Severely Errored Cell Blocks
SECBR	Severely Errored Cell Block Ratio
STM	Synchronous Transfer mode
SVC	Switched Virtual Circuit
TCP/IP	Transmission Control Protocol/Internet Protocol
TDM	Time Division Multiplexing
TM	Traffic Management
UBR	Unspecified Bit Rate
UNI	User Network Interface
UPC	Usage Parameter Control
VBR	Variable Bit Rate
VBR-nrt	Variable Bit Rate non-real time
VBR-rt	Variable Bit Rate real time
VCI	Virtual Channel Identifier
VPI	Virtual Path Identifier
WAN	Wide Area Network

Lista de Figuras

FIGURA 1.1 – Relação entre Fatores Internos e Externos.....	16
FIGURA 2.1 - Elementos da rede ATM	23
FIGURA 2.2 - Contrato de Tráfego	25
FIGURA 2.3 - Necessidades de QoS para diferentes aplicações	27
FIGURA 2.4 - Qualidade oferecida pela Rede ATM.....	29
FIGURA 2.5 - Conformidade de tráfego CBR e VBR.....	30
FIGURA 2.6 - Ação tomada conforme definição de conformidade.....	30
FIGURA 2.7 - Tempo de transferência da célula de um usuário a outro	32
FIGURA 2.8 - Fatores que compõem o atraso na entrega da célula	33
FIGURA 2.9 - Atraso de Enfileiramento	35
FIGURA 2.10 - Distribuição da probabilidade do CTD	36
FIGURA 2.11 - A variação no tempo de chegada da célula	37
FIGURA 2.12 - Eficiência CAC: garantia QoS vs. utilização da rede.....	42
FIGURA 2.13 - Pontos de verificação de conformidade	43
FIGURA 2.14 - Policiamento de Tráfego: analogia do balde furado.....	45
FIGURA 3.1 - Elementos do Relacionamento de Dependência	46
FIGURA 3.2 - Níveis de serviço (QoS), carga e recursos da rede [MEN85]	47
FIGURA 3.3 - Planejamento de Capacidade.....	48
FIGURA 3.4 - Comprometimento dos níveis de serviço devido à sobrecarga	49
FIGURA 3.5 - Técnicas de Previsão de Desempenho.....	49
FIGURA 3.6 - Pontos de Alocação de Recursos.....	50
FIGURA 3.7 - Exemplo de Topologia para uma Conexão	52
FIGURA 3.8 - Chegada simultânea de células.....	54
FIGURA 3.9 - Alocação pela Taxa de Pico para CBR e VBR	55
FIGURA 3.10 - BE: Banda Equivalente, $SCR \leq BE \leq PCR$	55
FIGURA 3.11 - Multiplexador Estatístico	57
FIGURA 3.12 - Variação na taxa de pico de uma conexão CBR	59
FIGURA 4.1 - Cenário de Simulação SIM_01.....	66
FIGURA 4.2 - Ações simuladas de um comutador ATM.....	67
FIGURA 4.3 - Estrutura de filas e escalonador por prioridade de categoria de serviço	68
FIGURA 4.4 - Simulação do acesso dos equipamentos terminais à rede	69
FIGURA 4.5 - Cenário de Simulação.....	70
FIGURA 4.6 - Exemplo de Curva de Desempenho obtida na simulação	72
FIGURA 4.7 - Intervalo de Confiança: Método de Replicação da Simulação	74
FIGURA 4.8 - Intervalo de Confiança: Método de Subintervalos.....	75
FIGURA 4.9 - Quantidade de Subintervalos (esquerda) e Intervalo de Confiança de 99% para Atraso (direita) em função do Tamanho do Subintervalo utilizado	76
FIGURA 4.10 - Tempo Simulado (esquerda) e Quantidade de Células Analisadas (direita) em função do Tamanho do Subintervalo utilizado	77
FIGURA 4.11 - Estimativa do CTD e CDV para uma conexão.....	83
FIGURA 4.12 - Estimativa do CLR para duas conexões simuladas.....	85
FIGURA 4.13 - CTD para Multiplexação de 1 a 5 conexões com <i>buffer</i> de 100 células (esquerda) e com <i>buffer</i> de 10.000 células (direita).....	88

FIGURA 4.14 - CTD para Multiplexação de 6, 8, 10, 15 e 25 conexões com <i>buffer</i> de 100 células (esquerda) e com <i>buffer</i> de 10.000 células (direita).....	88
FIGURA 4.15 - CLR para Multiplexação de " <i>n</i> " conexões com <i>buffer</i> de 100 células (esquerda) e com <i>buffer</i> de 10.000 células (direita).....	89
FIGURA 4.16 - CTD para Multiplexação de " <i>n</i> " conexões para transmissão de TV com <i>buffer</i> de 100 células.	90
FIGURA 4.17 - CDV para Multiplexação de " <i>n</i> " conexões para transmissão de TV com <i>buffer</i> de 100 células	90
FIGURA 4.18 - CLR para Multiplexação de " <i>n</i> " conexões para transmissão de TV com <i>buffer</i> de 100 células. À direita, gráfico ampliado mostrando CLR próximo a 10^{-4}	90
FIGURA 4.19 - CTD para Multiplexação de " <i>n</i> " conexões para transmissão de TV com <i>buffer</i> de 100 células. Por Tempo de Célula (esquerda) e por unidade de tempo (direita).	91
FIGURA 4.20 - CTD para Multiplexação de " <i>n</i> " conexões para transmissão de TV com <i>buffer</i> de 1.000 células. Por Tempo de Célula (esquerda) e por unidade de tempo (direita).	91
FIGURA 4.21 - CTD para Multiplexação de " <i>n</i> " conexões para transmissão de TV com <i>buffer</i> de 10.000 células. Por Tempo de Célula (esquerda) e por unidade de tempo (direita).....	91
FIGURA 4.22 - CTD para Multiplexação de " <i>n</i> " conexões para vídeo conferência com <i>buffer</i> de 100 células.	92
FIGURA 4.23 - CDV para Multiplexação de " <i>n</i> " conexões para vídeo conferência com <i>buffer</i> de 100 células	92
FIGURA 4.24 - CLR para Multiplexação de " <i>n</i> " conexões para vídeo conferência com <i>buffer</i> de 100 células. À direita, gráfico ampliado mostrando CLR próximo a 10^{-4}	92
FIGURA 4.25 - CTD para Multiplexação de " <i>n</i> " conexões para vídeo conferência com <i>buffer</i> de 100 células. Por Tempo de Célula (esquerda) e por unidade de tempo (direita).	93
FIGURA 4.26 - CTD para Multiplexação de " <i>n</i> " conexões para vídeo conferência com <i>buffer</i> de 1.000 células. Por Tempo de Célula (esquerda) e por unidade de tempo (direita).	93
FIGURA 4.27 - CTD para Multiplexação de " <i>n</i> " conexões para vídeo conferência com <i>buffer</i> de 10.000 células. Por Tempo de Célula (esquerda) e por unidade de tempo (direita).....	93
FIGURA 4.28 - CTD para Multiplexação de " <i>n</i> " conexões para vídeo sob demanda com <i>buffer</i> de 100 células.	94
FIGURA 4.29 - CDV para Multiplexação de " <i>n</i> " conexões para vídeo sob demanda com <i>buffer</i> de 100 células	94
FIGURA 4.30 - CLR para Multiplexação de " <i>n</i> " conexões para vídeo sob demanda com <i>buffer</i> de 100 células. À direita, gráfico ampliado mostrando CLR próximo a 10^{-4}	94
FIGURA 4.31 - CTD para Multiplexação de " <i>n</i> " conexões para vídeo sob demanda com <i>buffer</i> de 100 células. Por Tempo de Célula (esquerda) e por unidade de tempo (direita).	95
FIGURA 4.32 - CTD para Multiplexação de " <i>n</i> " conexões para vídeo sob demanda com <i>buffer</i> de 1.000 células. Por Tempo de Célula (esquerda) e por unidade de tempo (direita).	95

FIGURA 4.33 - CTD para Multiplexação de " n " conexões para vídeo sob demanda com <i>buffer</i> de 10.000 células. Por Tempo de Célula (esquerda) e por unidade de tempo (direita).	95
FIGURA 4.34 - CTD e CLR obtidos com o simulador SMURPH. Fonte:[RAM98]	98
FIGURA 4.35 - CTD e CLR obtidos com o simulador NIST ATM/HFC.....	99

Lista de Tabelas

TABELA 2.1 - Campos do cabeçalho	24
TABELA 2.2 - Descritores de Tráfego e QoS por Categoria de Serviço.....	26
TABELA 2.3 - Definições de conformidade por categoria de serviço	31
TABELA 2.4 - Tempos de Propagação.....	34
TABELA 2.5 - Tempo de transmissão de uma célula (424 bits)	34
TABELA 2.6 - Informações de comutadores disponíveis no mercado.....	34
TABELA 2.7 - Exemplos de Requisitos de CTD e CDV	36
TABELA 2.8 - Controle de Tráfego baseado em conexão e célula-a-célula	41
TABELA 3.1 - Modelo de tabela: Recursos Disponíveis x Alocados	52
TABELA 3.2 - Caracterização de Serviços.....	52
TABELA 3.3 - Caminho das Conexões	53
TABELA 3.4 - Modelo: Tabela Conexões por Porta	53
TABELA 4.1 - Relação do tempo simulado e tempo de execução	66
TABELA 4.2 - Tipos de Aplicações ATM simuladas.....	69
TABELA 4.3 - Banda Efetiva Aproximada	72
TABELA 4.4 - Serviços Simulados	86
TABELA 4.5 - Parâmetros Descritores VBR Poisson	87
TABELA 4.6 - Cenários Simulados com 100 células de <i>buffer</i>	89
TABELA 4.7 - Alocação de Recursos para Serviço de Transmissão TV com <i>Buffer</i> =100	96
TABELA 4.8 - Alocação de Recursos para Serviço de Transmissão TV com <i>Buffer</i> =1.000 e 10.000	97
TABELA 4.9 - Alocação de Recursos para Serviço de Vídeo Conferência com <i>Buffer</i> =100, 1.000 e 10.000	97
TABELA 4.10 - Alocação de Recursos para Serviço de Vídeo Sob Demanda com <i>Buffer</i> =100, 1.000 e 10.000	98
TABELA Anexo 1.1 - Parâmetro de QoS para conexões em uma porta OC-3c e OC-12	103
TABELA Anexo 1.2 - Parâmetro de QoS para conexões em uma porta DS-3	103
TABELA Anexo 1.3 - Parâmetro de QoS para conexões em uma porta DS-1.....	103
TABELA Anexo 2.1 - Fatores para combinação	107
TABELA Anexo 3.1 - Notas das Alterações Efetuadas.....	109
TABELA Anexo 4.1 - Relação dos <i>Scripts</i> e suas Funções sobre os Arquivos.....	112

Resumo

O processo de dimensionar redes de comunicação tem sido um desafio para pesquisadores e projetistas. A partir da especificação, passando pela operação, controle e gerenciamento de redes, as estimativas de comportamento do desempenho são úteis para o dimensionamento adequado dos equipamentos. O detalhamento e precisão na capacidade de avaliar o impacto de carga futura melhoram as chances de prever dificuldades em atingir metas de serviços. Com redes de banda larga, como ATM, não tem sido diferente. Pela sua concepção de oferecer garantia de qualidade para serviços com diferentes requisitos, ATM se empenha em evitar a sobrecarga da rede. De início, essa premissa é preservada naturalmente através da restrição da quantidade e tipo de conexões ingressas na rede. Portanto, a adequação dos recursos que compõem a estrutura de uma rede ATM determina o grau de disponibilidade em atender certa quantidade de serviços. A pergunta que desejamos responder é: como estimar com precisão a quantidade de serviço suportada por determinada rede ATM? O limite da rede é alcançado quando os recursos disponíveis são menores que os recursos necessários à carga de serviço. Com o emprego cada vez maior de ATM por empresas de telecomunicações, conhecer o limite da rede é estar ciente da potencialidade de negócios sem comprometimento da qualidade. É poder prever expansões evitando bloqueio de novos serviços. O processo de dimensionamento de capacidade de uma rede ATM revela a quantidade de recursos necessários para suportar determinada carga de serviço. Quando os recursos necessários forem maiores que os recursos disponíveis, o limite da rede foi alcançado. Nesse caso, são duas as possibilidades para o equilíbrio: aumentar os recursos da rede ou diminuir a carga de serviço desejado.

Esta dissertação propõe um método para dimensionamento dos recursos de uma rede ATM. A principal técnica empregada no método é a simulação do comportamento de tráfego sobre comutadores ATM. Para determinada carga de tráfego são executadas diferentes simulações variando os recursos disponíveis dentro de parâmetros prováveis. As seguintes medidas de desempenho são obtidas nas simulações como resultados estatísticos médios: razão de perda de células (CLR), atraso de transferência de células (CTD) e variação do atraso de células (CDV). Conhecendo o desempenho desejado (QoS) pela carga de serviço, o método pode determinar a quantidade necessária de recursos que satisfazem os requisitos de QoS.

A ferramenta escolhida para implementar o modelo foi o simulador orientado a eventos ATM/HFC do *National Institute of Standards and Technology* (NIST). O simulador é composto por diferentes modelos de elementos, cada qual com seus atributos, que podem ser combinados para caracterizar determinadas configurações de rede que se deseja avaliar. Os elementos podem ser desde representações de tipos de comutadores ATM até diferentes técnicas de controle de tráfego a serem utilizadas na simulação. A carga de serviço na simulação é provida por elementos modeladores que caracterizam diferentes tipos de aplicações geradoras de tráfego, permitindo arranjos de serviços CBR, VBR, ABR e UBR através de seus respectivos parâmetros descritores.

A validação do método é efetuada através da comparação dos resultados obtidos com outro trabalho similar desenvolvido utilizando simulação.

Palavras-Chave: ATM, Planejamento de Capacidade, Dimensionamento de Recursos, Simulação

TITLE:"CAPACITY PLANNING METHOD FOR ATM NETWORKS BASED ON SIMULATION"

Abstract

The process of planning communication networks has been a challenge for researchers and designers. From the specification, through the operation, control and management of networks, the behavior performance estimates are useful for the adequate equipment dimensioning. The detailing and accuracy in the capacity to evaluate the future load impact increase the possibilities to forecast difficulties in reaching goals of services. With broadband networks, as ATM, it has not been different. ATM efforts in preventing network overload by its conception to offer quality guarantee for services with different requirements. From beginning, this premise is naturally preserved through restriction of the amount and type of connections that can enter the network. Therefore, the adequacy of the resources that compose the ATM network structure determines the degree of availability in attending certain amount of services. The question that we wish to answer is: how can we estimate accurately the amount of services supported by specific ATM network?

The limit of the network is reached when the available resources are below the necessary resources to service load. With the higher use of ATM for telecommunications companies, to know the limit of network is to be aware of the potentiality without damage to the quality. It's to be able to forecast expansions to prevent new services blocking. The capacity planning process of an ATM network shows the amount of resources needed to support a specific workload. When the resource need is greater than the available resource, the network limit has been reached. In this case, there are two possibilities to reach balance: increase the network resources or lower the load of desired service.

This work is about a method for ATM network resources dimensioning. The main technique used in the method is the traffic behavior simulation over ATM switches. For specific workload, different simulations are executed and they vary according to the resources available inside the probable parameters. The following measures of performance are gotten in the simulations as average statistics results: cell loss ratio (CLR), cell transfer delay (CTD) and cell delay variation (CDV). Knowing the workload desired performance (QoS), the method can determine the necessary amount of resources that will satisfy the QoS requirements.

The chosen tool to implement the model was the event driven simulator ATM/HFC of the National Institute of Standards and Technology (NIST). The simulator is made up of different models of elements, each one with its attributes, which can be combined to characterize specific network configurations that are to be evaluated. The elements can range from representations of types of ATM switches to different techniques of traffic management to be used in the simulation. The workload in the simulation is provided by modeler elements that characterize different types of traffic generator applications, allowing sets of CBR, VBR, ABR and UBR services through their respective traffic parameters.

The method validation is carried out through the matching of the results gotten with other similar work developed using simulation.

Keywords: ATM, Capacity Planning, Resources Dimensioning, Simulation.

1 Introdução

1.1 Rede Multisserviços

Por muito tempo a principal infra-estrutura de comunicação mundial foi a rede telefônica baseada na comutação de circuitos analógicos. As conexões de voz estabelecidas nessas redes recebem toda banda definida do circuito, ou seja, o recurso para transmissão não é compartilhado. O sistema de telefonia atual com comutação de circuitos digitais utiliza a técnica de multiplexação por divisão de tempo. O meio de transmissão é alocado em fatias de tempo fixas para cada conexão de voz, ou seja, a alocação do meio para cada circuito é posicional no tempo. A vantagem nessas duas técnicas de comutação é a utilização de uma taxa constante de transmissão que possibilita um valor controlado de atraso da origem até o destino. Por outro lado, dedicar o circuito ou a fatia de tempo de um circuito causa desperdício, pois o recurso mantém-se totalmente alocado até o final da conexão passando por diversos períodos de silêncio de transmissão.

Desenvolvida inicialmente para a transmissão da voz humana, a rede telefônica se mostrou inadequada à demanda de comunicação de dados de longa distância entre computadores. A partir dessa necessidade, surge uma nova infra-estrutura de comunicação utilizando a técnica de comutação de pacotes que não requer para as conexões um circuito de transmissão dedicado. Nesse caso a conexão é lógica, o que permite que várias conexões compartilhem o meio de transmissão otimizando assim o seu uso. Os pacotes têm tamanhos variáveis e possuem identificadores da conexão a que pertencem. Através da identificação é feito o encaminhamento em cada comutador para determinar o circuito físico que leva o pacote até o destino. A tecnologia atinge, dessa forma, ganhos de utilização dos meios mas impõe, devido à multiplexação do meio, retardos variáveis na entrega dos pacotes. As redes de comunicação de dados X.25 e *Frame Relay* são exemplos do emprego da tecnologia de comutação de pacotes.

Um novo sistema de comunicação é demandado para atender eficientemente diferentes serviços. Os estudos para a integração de sistemas de comunicação para voz, vídeo e dados levaram ao desenvolvimento das redes digitais de alta velocidade para múltiplos serviços. A possibilidade de criar uma infra-estrutura única de comutação e transmissão oferecendo diversos tipos de serviços reuniu interesses de diferentes setores de atividades como telecomunicações e computação para cooperarem com o estabelecimento da nova tecnologia. O sistema proposto recebe a denominação de Rede Digital de Serviços Integrados (RDSI). Com a necessidade de maiores taxas de transmissão por novos serviços de comunicação surge especificadamente a RDSI de faixa larga (RDSI-FL). O objetivo é fornecer simultaneamente, de uma forma padronizada sobre um único acesso físico, os mais diversos serviços como: vídeo sob demanda, interconexão de redes de computadores, difusão de TV, vídeo conferência e telefonia.

O *International Telecommunications Union - Telecommunication (ITU-T)*, no final da década de 80, especifica a tecnologia ATM (*Asynchronous Transfer Mode*) como sendo a base da implementação do RDSI-FL. As pesquisas sobre ATM já vinham ocorrendo desde o início da década de 80 quando se buscava um modo de transferência

eficiente para diversos tipos de serviços de comunicação. Com o passar do tempo, o ATM aparece como alternativa não só como a tecnologia básica associada às redes de longa distância (WAN) mas também para as redes locais (LAN). ATM é uma tecnologia de comunicação que provê dutos de alta velocidade para a transferência de qualquer tipo de informação digital em pequenos pacotes de tamanho fixo, chamados células. A idéia básica é estabelecer entre dois pontos que desejam se comunicar um circuito virtual que é mantido pelos comutadores de células para levar a informação de um lado a outro. A característica marcante do ATM é o QoS (Qualidade de Serviço), que garante o desempenho predefinido que determinado serviço necessita. Isso permite suportar aplicações de tempo real que são sensíveis ao atraso como por exemplo conversas telefônicas. O atendimento à diversidade de características de tráfego e exigências distintas de grandes quantidades de serviços é possível pelo ATM graças ao controle de tráfego reunido à capacidade de multiplexação estatística do meio em altas taxas de transmissão.

O ATM é um padrão aberto normalizado pelo ITU-T, um organismo de padronização internacionalmente reconhecido. Há outras entidades envolvidas nas definições da tecnologia ATM. Uma das que merece destaque é o ATM Forum, um consórcio autônomo criado em 1991 que reúne fabricantes, fornecedores de serviços e usuários ATM que tem como objetivo clarificar os padrões desenvolvidos e propor especificações de implementações. A tendência mundial crescente é a utilização da tecnologia ATM na implantação de redes metropolitanas de alta velocidade para o suporte integrado de serviços como telefonia, videoconferência e acesso à base de dados.

Devido ao comprometimento com a qualidade dos serviços transportados e a limitação física dos recursos, como capacidade de transmissão dos enlaces, a rede ATM possui um ponto flutuante de saturação dos recursos. Em estado de saturação, a rede impede o ingresso de novos serviços ou começa a sofrer queda de desempenho devido a congestionamentos. O ponto de saturação é flutuante pois depende de fatores externos à configuração dos recursos da rede. Os três fatores externos são quantidade de tráfego, requisitos de desempenho e topologia de conexões. A combinação desses fatores especifica a carga de serviço submetida à rede. Cargas de serviços distintas aplicadas sobre determinada rede resultam em diferentes quantidades possíveis de atendimento a serviços. Portanto, o projeto de uma rede ATM deve considerar a carga de serviço fundamental para o dimensionamento dos recursos da rede, procurando sempre identificar os fatores externos com o maior detalhamento possível para adequar a rede aos serviços previstos.

Outros fatores que influenciam na capacidade de atendimento da rede são os fatores internos referentes à configuração da rede (fig. 1.1). Dentre esses estão os diferentes recursos que podem ser reunidos na configuração física da rede, como capacidade dos enlaces, capacidade de comutação, quantidade de *buffers*, topologia de enlaces e outros. São intrínsecos aos equipamentos utilizados ou caracterizam uma solução de configuração adotada para a rede. A outra parte dos fatores internos é referente à configuração lógica dos mecanismos adotados para o controle de tráfego. Novamente, combinando valores para esses fatores temos modificações na capacidade da rede. Enquanto os fatores externos consomem recursos, os fatores internos disponibilizam. O levantamento desses fatores e a avaliação de equilíbrio entre eles é o início da garantia de sucesso do projeto de uma rede ATM.

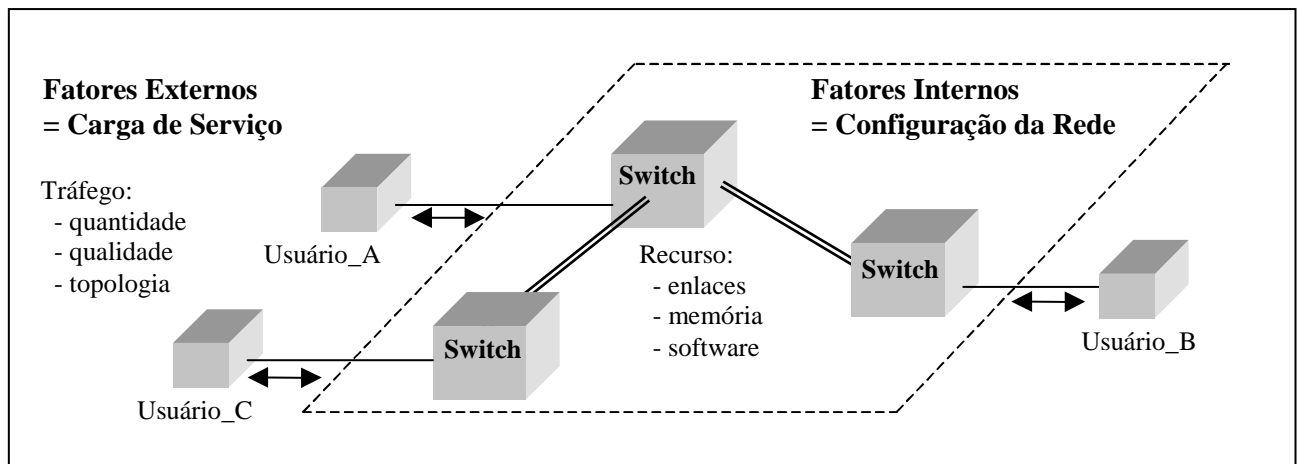


FIGURA 1.1 – Relação entre Fatores Internos e Externos

Superada a etapa de levantamento e qualificação do tráfego de serviços a serem suportados, resta avaliar o impacto dessa carga sobre a estrutura de recursos da rede. Sendo os recursos limitados, a rede possui um ponto de saturação que depende de suas características e técnicas utilizadas para atender os serviços demandados. Partindo de uma determinada carga de serviço, o comportamento de desempenho da rede depende exclusivamente das características dos fatores internos físicos e de controle. O conjunto de características físicas envolve, entre outros elementos, topologia de rede, taxa de transmissão dos enlaces, capacidade de comutação e quantidade de memória. São recursos limitados geralmente pelo custo ou inviabilidade técnica, e quanto maior a sua disponibilidade melhor serão as possibilidades de desempenho da rede. Os elementos de controle são representados por algoritmos que implementam os mecanismos de gerenciamento de tráfego, tais como admissão de conexões e tratamento de congestionamento. Os fatores de controle são fundamentais e suas configurações determinam as características de utilização da rede. Aperfeiçoamentos dos elementos de controle otimizam a utilização dos recursos físicos e conseqüentemente aumentam a capacidade de serviço suportada pela rede.

1.2 Motivação

Estudos para avaliar o desempenho ATM têm sido desenvolvidos por pesquisadores, como é visto em [PIT96]. A avaliação de desempenho, neste trabalho, trata da análise de um conjunto de medidas que procura determinar o grau de eficiência da prestação de serviços feito pela rede. Ao mesmo tempo, procura identificar pontos responsáveis por contenções (gargalos) que devam ser tratados. Fazer a avaliação de desempenho de uma rede instalada e operacional significa, na maioria dos casos, realizar medições de parâmetros de desempenho em campo por meio de equipamentos ou aplicativos de gerenciamento. Em outras situações não é possível fazer as medições em campo. Isto acontece quando estamos projetando uma rede nova ou quando estamos prevendo, sobre uma rede existente, um aumento futuro de carga. Nesse caso um dos métodos adotados é o de simulação do comportamento da rede para realizar a avaliação de desempenho.

Conforme os resultados obtidos na avaliação de desempenho de determinada carga de serviço podemos fazer o dimensionamento dos recursos da rede. Quanto mais

precisa a previsão dos resultados maior deverá ser a capacidade de configurar adequadamente e de prever dificuldades de desempenho.

A partir da especificação, passando pela operação, controle e gerenciamento de redes, as estimativas de comportamento do desempenho são úteis para o dimensionamento adequado dos equipamentos. Com redes de banda larga, como ATM, não tem sido diferente. Concebida para oferecer garantia de qualidade para serviços com diferentes requisitos, ATM se empenha em evitar a sobrecarga da rede. De início, essa premissa é preservada naturalmente através da restrição da quantidade e tipo de conexões ingressas na rede. Portanto, a adequação dos recursos que compõem a estrutura de uma rede ATM determina o grau de disponibilidade em atender certa quantidade de serviços. A pergunta que desejamos responder é: como estimar com precisão a quantidade de serviço suportada por determinada rede ATM? O limite da rede é alcançado quando os recursos disponíveis são menores que os recursos necessários à carga de serviço. Com o emprego cada vez maior de ATM por empresas de telecomunicações, conhecer o limite da rede é estar ciente da potencialidade de negócios sem comprometimento da qualidade. É poder prever expansões evitando bloqueio de novos serviços.

O processo de dimensionamento de capacidade de uma rede ATM revela a quantidade de recursos necessários para suportar determinada carga de serviço. Quando os recursos necessários forem maiores que os recursos disponíveis, o limite da rede foi alcançado. Nesse caso são duas as possibilidades para o equilíbrio: aumentar os recursos da rede ou diminuir a carga de serviço desejado.

1.3 Objetivos

Este trabalho apresenta uma proposta de método para o dimensionamento de capacidade de redes ATM que pode ser utilizado como apoio ao projeto desse tipo de rede. A proposta é baseada na avaliação de desempenho obtida pela simulação orientada a eventos do comportamento de tráfego sobre comutadores ATM. O modelo utilizado na simulação é caracterizado com elementos fundamentais que atuam no desempenho dos serviços, tais como: tamanho do buffer, modelo de tráfego, quantidade de conexões, largura de banda do enlace, e outros.

Para determinada carga de tráfego são executadas diferentes simulações variando os recursos disponíveis dentro de parâmetros prováveis. As seguintes medidas de desempenho são obtidas nas simulações como resultados estatísticos médios: razão de perda de células (CLR), atraso de transferência de células (CTD) e variação do atraso de células (CDV). Conhecendo o desempenho desejado (QoS) pela carga de serviço, o método pode determinar a quantidade necessária de recursos que satisfazem os requisitos de QoS. Diferentes configurações de redes e cargas de serviços podem ser exercitadas para revelar limites necessários de alocação de recursos que evitem dificuldades de desempenho.

O produto escolhido para implementar o modelo foi a ferramenta de simulação orientada a eventos ATM/HFC do *National Institute of Standards and Technology* (NIST). A ferramenta é composta por diferentes modelos de elementos, cada qual com seus atributos, que podem ser combinados para caracterizar determinadas configurações de rede que se deseja avaliar. Os elementos podem ser desde representações de tipos de

comutadores ATM até diferentes técnicas de controle de tráfego a serem utilizadas na simulação. A carga de serviço na simulação é provida por elementos modeladores que caracterizam diferentes tipos de aplicações geradoras de tráfego, permitindo arranjos de serviços CBR, VBR, ABR e UBR utilizando seus respectivos parâmetros descritores.

A validação do método é efetuada através da comparação dos resultados obtidos com outro trabalho similar desenvolvido utilizando simulação. Para alcançar o objetivo de dimensionamento de capacidade ATM, as seguintes metas foram traçadas:

- 1) Definir como estimar a alocação lógica de recursos para categoria de serviços CBR, VBR-rt, VBR-nrt, ABR e UBR.
- 2) Caracterizar os modelos de tráfego VBR-rt e VBR-nrt (ex.: vídeo conferência e vídeo sob demanda).
- 3) Adequar o simulador ATM/HFC.
 - disponibilizar valores estatísticos médios de CTD e CDV por conexão;
 - construção do intervalo de confiança para CTD e CDV pelo método de subintervalos de tempo de simulação;
- 4) Gerar programas para escalonar execuções de cenários de alocação de recursos.
- 5) Dimensionar recursos conforme gráficos de desempenho obtidos nas simulações.

1.4 Estrutura da Dissertação

Esta dissertação está estruturada em 5 capítulos, incluindo esta introdução, da seguinte forma:

- **2 Desempenho em redes ATM:** nesse capítulo são apresentados conceitos da rede ATM relacionados à qualidade de serviço. Também relacionamos as medidas de desempenho utilizadas para avaliar a qualidade de serviço oferecida. No final do capítulo, descrevemos alguns mecanismos de controle de tráfego que são responsáveis por garantir o QoS das conexões;
- **3 Planejamento de Capacidade de Redes ATM:** iniciamos esse capítulo comentando a relação entre carga, capacidade e desempenho em redes. Em seguida, apresentamos o método para planejamento de capacidade e os modos de alocação de recursos utilizados conforme a categoria de serviço;
- **4 Simulação para Dimensionamento de Capacidade:** aborda o uso do simulador NIST ATM/HFC para o planejamento de capacidade de serviços VBR. Resultados práticos são apresentados para demonstrar a implementação do método e validar o modelo;
- **5 Conclusões e Trabalhos Futuros:** Considerações e sugestões de melhorias que podem ser implementadas no processo de planejamento de capacidade;

Possuindo os seguintes anexos:

- **Anexo 1 Parâmetros de QoS Genéricos;**
- **Anexo 2 Execução do Simulador;**
- **Anexo 3 Relação dos Programas Alterados no NIST ATM/HFC;**
- **Anexo 4 *Scripts* e Arquivos utilizados na Simulação;**

2 Desempenho em Redes ATM

2.1 Sistemas Eficientes

Um sistema computacional é composto por diversas atividades combinadas ordenadamente para atender às necessidades dos usuários. Recursos finitos e, na maioria das vezes, custos consideráveis são utilizados nos sistemas. Não basta um sistema executar seus procedimentos corretamente e sim procurar fazê-lo de forma eficiente. Pela diversidade de resultados alcançados e de recursos utilizados, são inúmeras as possibilidades de relações indicadoras de eficiência em um sistema. Um indicador básico de eficiência em sistemas computacionais é o tempo consumido por um procedimento para produzir uma quantidade de resultado. A capacidade de transferência de dados entre dois dispositivos é um exemplo desse tipo de indicador onde o valor é dado na unidade de bits por segundo (bit/s). Os indicadores de eficiência são medidas de desempenho das características do sistema.

Consideramos a eficiência de um sistema de forma relativa. A eficiência está em alcançar determinada medida de desempenho. Certamente após um avanço tecnológico esse mesmo nível de desempenho será considerado ineficiente ou insuficiente para a demanda existente. Vejamos a evolução da comunicação de dados nos últimos anos. O desempenho máximo da capacidade de transmissão (largura de banda) passou de medidas em Kbit/s para Gbit/s, e logo estaremos falando em unidades de Terabits. Ainda desconsiderando qualquer evolução tecnológica, um sistema pode superar seu próprio desempenho após passar por uma reconfiguração para otimizá-lo. De uma forma ou de outra, o patamar de desempenho para um sistema ser considerado eficiente está sempre em gradual crescimento.

Comparar as medidas de desempenho de sistemas é o caminho para dizer qual é o mais eficiente. A comparação utiliza valores numéricos que representam as quantidades alcançadas pelas atividades do sistema, ou seja, cada medida de desempenho é feita sobre uma característica específica como quantidade processada, tempo de resposta, número de usuários, velocidade de acesso, memória utilizada, e outras tantas. A comparação é feita considerando medidas que quantificam a mesma característica de desempenho. O mínimo tolerável para as características de desempenho é determinado pelas necessidades básicas que viabilizam a utilização do sistema. Em geral pode-se exceder o desempenho mínimo esperado, mas nunca ficar aquém do necessário.

Comparando as medidas de desempenho entre sistemas, considerando a finalidade, podemos dizer qual o mais eficiente mas não qual o mais apropriado. A eficiência não é o único fator determinante na seleção de sistemas computacionais. O fato de um sistema ser mais eficiente do que outro não garante que será escolhido para ser utilizado. Outros fatores de benefício, tais como facilidade operacional, escalabilidade, disponibilidade, padronização e custo de implementação, influenciam na decisão. A evolução tecnológica para atingir níveis superiores de desempenho tem um custo a ser considerado. Sistemas podem ter desempenhos extraordinários e grandes facilidades mas terem implantação inviável devido ao custo. Os sistemas mais eficientes

também podem ser descartados por não haver necessidade prática de atingir os níveis de desempenho apresentados. Como em qualquer processo de escolha, deve ser pesada a relação entre os custos e os benefícios.

2.2 QoS e Capacidade de Atendimento em Redes de Comunicação

Como qualquer outro sistema, as redes de comunicação têm medidas de desempenho para avaliar sua eficiência. As questões referentes ao desempenho são importantes pois influenciam diretamente na quantidade e qualidade dos serviços prestados. Vejamos como exemplo a largura de banda (vazão) que restringe a quantidade de bits que são transferidos por segundo em um meio físico. Se um usuário ou um grupo de usuários demandar um desempenho de largura de banda superior à disponível, ocorrerá uma contenção e enfileiramento para utilização do meio físico, impactando a vazão de informação pretendida pelo usuário. Os recursos de rede disponibilizados de forma compartilhada são alocados temporariamente pelos usuários. Após a utilização, o recurso é liberado, tornando-o disponível para outro usuário. A quantidade máxima de usuários atendidos simultaneamente é dada pela quantidade de recursos da rede. Ultrapassando esse limite haverá um congestionamento e o usuário deverá aguardar a liberação do recurso para estabelecer a comunicação, ou seja, perda de qualidade de serviço. Como existe a alternância de usuários utilizando a rede, o compartilhamento disponibiliza o serviço de comunicação para uma quantidade maior de usuários. Assim, aproveita-se melhor a estrutura da rede reduzindo o custo por usuário.

Os recursos utilizados em uma rede de comunicação são limitados e, por questões de viabilidade, compartilhados pelos usuários. Como foi visto, recursos limitados significa uma capacidade de atendimento restrita de usuários. A capacidade está relacionada com a quantidade de recursos alocados por cada usuário. As exigências de qualidade do serviço determinam a quantidade necessária e a forma de alocação dos recursos. Portanto, qualidade de serviço e capacidade de atendimento estão inter-relacionados. O aumento da exigência de qualidade reduz a capacidade de oferecer serviço, e vice-versa. O ideal é buscar o equilíbrio entre garantir a qualidade dos serviços e maximizar a utilização dos recursos da rede. A tecnologia adotada na rede define a relação entre qualidade de serviço e capacidade de atendimento. A seguir é apresentado o comportamento dessas grandezas em duas redes de distintas finalidades: telefonia e computação.

Nas redes telefônicas, o desempenho é avaliado utilizando parâmetros como: tempo para obter sinal para discar, tempo para estabelecer a ligação, disponibilidade de troncos, qualidade do som (relação sinal/ruído, ecos,...) e confiabilidade da ligação. Esses parâmetros determinam a qualidade de serviço (QoS) prestado. A tecnologia tradicional de rede empregada para carregar voz é chamada de Comutação de Circuitos. É uma tecnologia orientada a conexão que utiliza técnicas de multiplexação para concentrar muitas conversações em um único tronco físico. A técnica de multiplexação TDM (*Time Division Multiplexing*) não requer funções complexas de controle de tráfego. Devido ao assinalamento dos usuários a intervalos fixos de tempo, a qualidade do serviço é determinística e previsível (exceto por erros de transmissão e bloqueio de conexão) e os usuários não podem afetar a qualidade de serviço de outro usuário. Não há contenção por banda ou recurso, a não ser pelo número de intervalos de tempo

disponíveis para estabelecer conexões. Entretanto, o esquema TDM não permite atingir o importante objetivo de maximizar a utilização da banda utilizada. Muitas aplicações, como transferência de arquivo e vídeo conferência, requerem banda variável durante suas transmissões. Alocação de banda acima do necessário leva a uma severa ineficiência, pois a banda não utilizada por transmissões lentas ou intermitentes poderia ser utilizada por outra conexão. Em TDM as fatias de tempo são alocadas para uma conexão independente se o usuário está ou não transmitindo algo. A forma de alocação garante a qualidade mas possibilita o desperdício de recurso. Por causa de sua alocação estática dos intervalos de tempo, o TDM não é eficiente para suportar tráfego com características de rajada. Por outro lado é eficiente para suportar tráfego de voz [FLA91].

A finalidade da rede de comunicação de dados é conectar e transportar informações entre computadores remotos, chamados usuários da rede. A eficiência está em transmitir o mais rápido possível quantidades de informações válidas entre o maior número de usuários. O parâmetro de desempenho básico utilizado em redes de computadores é a vazão (*throughput*) medida em bit/s ou octetos/s. A vazão da rede é afetada pelo congestionamento, ou seja, pela sobrecarga temporária de recursos. Congestionamento que geralmente causa o aumento de retransmissão das informações. Essas variações de sobrecarga sobre a rede são percebidas pelos usuários através do aumento do tempo de resposta, gerando insatisfações quanto à qualidade do serviço prestado. O ideal seria a rede ter um comportamento transparente, como se os usuários estivessem interligados sem o compartilhamento de recursos. Dessa forma não haveria nenhum tipo de contenção de tráfego causado pela rede. Aproximar a realidade das redes de dados a esse ideal é uma meta a ser perseguida. A tecnologia tradicional de rede projetada para carregar dados é chamada de Comutação de Pacotes. Os pacotes podem ser comutados sem estabelecer uma conexão prévia. É o que acontece com o protocolo IP. Pacotes são roteados em cada nó da rede e nenhuma banda é pré-alocada para a transferência de dados. Essa tecnologia tem a capacidade de maximizar o uso da rede mas não pode garantir qualidade de serviço porque não há qualquer tipo de alocação de recurso. Essa situação caracteriza uma rede de melhor esforço onde a largura de banda ocupada pelo usuário varia de acordo com a carga da rede a cada instante. A maior rede de computadores, a Internet, foi projetada como rede de melhor esforço, e originalmente não previa qualquer mecanismo para garantia de qualidade. A filosofia de melhor esforço tem se tornado inaceitável, pois a qualidade degrada rapidamente quando a demanda da rede cresce. Simplesmente aumentar a capacidade dos enlaces e dos equipamentos não garante qualidade indefinidamente. Propostas de mecanismos de identificação de prioridades e alocação de recursos têm sido colocadas para suprir QoS em redes TCP/IP. Podemos citar o protocolo DIFFerentiated SERVices (DIFFSERV) e MPLS (Multiprotocol Label Switching) como alternativas para implementar QoS. Detalhes dessas tecnologias são apresentadas em [MCD00].

Para serem eficientes, as soluções de redes devem maximizar a utilização dos recursos da rede enquanto atingem necessidades de qualidade de serviço (QoS) de diferentes aplicações. Nenhuma das tecnologias apresentadas, comutação de circuito e comutação de pacotes, atendem satisfatoriamente esse objetivo, ou por não usarem eficientemente os recursos, caso da comutação de circuitos, ou por não garantirem qualidade, caso da comutação de pacotes. Pela necessidade de suportar eficientemente diferentes requisitos de serviço em uma única infra-estrutura de rede, surge a tecnologia ATM baseada na comutação de células.

O inerente conflito criado pela necessidade de otimizar a utilização da rede enquanto garante diferentes QoS pode ser resolvido com a combinação de técnicas de controle de tráfego com a capacidade de multiplexação estatística do ATM. Utilizando serviços orientados à conexão, o ATM avalia a cada solicitação de estabelecimento de conexão se possui recursos suficientes para suportar nova conexão sem prejudicar as já estabelecidas. Essa é uma das funções de controle de tráfego (TM - *Traffic Management*) do ATM projetada para prevenir a rede de se tornar sobrecarregada. Mas se uma sobrecarga acontecer, as funções de controle de tráfego podem reagir no congestionamento para manter os objetivos de QoS enquanto maximiza o uso dos recursos da rede. Lidando com diferentes velocidades de acesso, o ATM pode adaptar tráfego de diferentes protocolos sobre uma única infra-estrutura de rede.

Outras tecnologias como IP e Frame Relay aspiram suportar e consistentemente disponibilizar diferentes níveis de qualidade de serviço. Para isso devem implementar técnicas de controle de tráfego. Devido ao tamanho fixo das células, a implementação dessas técnicas é mais fácil no ATM que em outras tecnologias. Para que equipamentos de diferentes fornecedores possam coexistir em uma mesma rede, alguns componentes e técnicas de controle de tráfego necessitam ser definidos de tal forma que permitam a interoperabilidade. Duas entidades são as responsáveis pela interoperabilidade do ATM: ITU-T e ATM Forum. O ITU-T estabelece o padrão formal e o ATM Forum é a organização criada por um grupo de companhias (fabricantes, provedores de serviço em rede, usuários,...) para desenvolver as especificações de interoperabilidade. A padronização das funções básicas do controle de tráfego do ATM já está completa no ITU-T e no ATM Forum.

2.3 Princípios de Redes ATM

Qualidade de serviço (QoS) tornou-se o fator chave no desenvolvimento de novas tecnologias de redes. Serviços com requisitos de QoS exigem tratamentos diferenciados. A qualidade de serviço é definida como sendo o efeito coletivo do desempenho do serviço e que determina o grau de satisfação do usuário desse serviço.

ATM é uma técnica de transmissão e multiplexação orientada a conexões que provê uma forma eficiente de garantir os requisitos de qualidade de serviço das diversas aplicações de comunicação¹. Internet, telefonia, televisão, vídeo conferência, vídeo sob demanda, tele-medicina e muitas outras aplicações vêm impondo o uso crescente de banda e o suporte simultâneo de diferentes serviços sobre a mesma rede de telecomunicações. ATM é capaz de integrar diferentes tipos de redes de comunicação em uma única e consolidada rede de banda larga.

2.3.1 Multiplexação de Células

Uma rede ATM consiste basicamente de uma série de equipamentos terminais (usuários) e um conjunto de comutadores (*switches*) interconectados por enlaces ponto-a-ponto (fig. 2.1). Os comutadores têm interfaces para conectar usuário à rede (UNI -

¹ Uma aplicação de comunicação é definida como a tarefa que requer enviar um ou mais fluxos de informações entre duas ou mais localizações. [KOW97]

User Network Interface) e para conexão entre comutadores (NNI - *Network-Network Interface*). A informação é transmitida pela rede em um formato de pacote de tamanho fixo, chamado célula ATM. Uma célula ATM é formada por 53 bytes, sendo 48 de informação (*pay-load*) e 5 de cabeçalho (*header*). O uso de células pequenas e de comprimento fixo simplifica o gerenciamento dos *buffers* e o projeto dos comutadores.

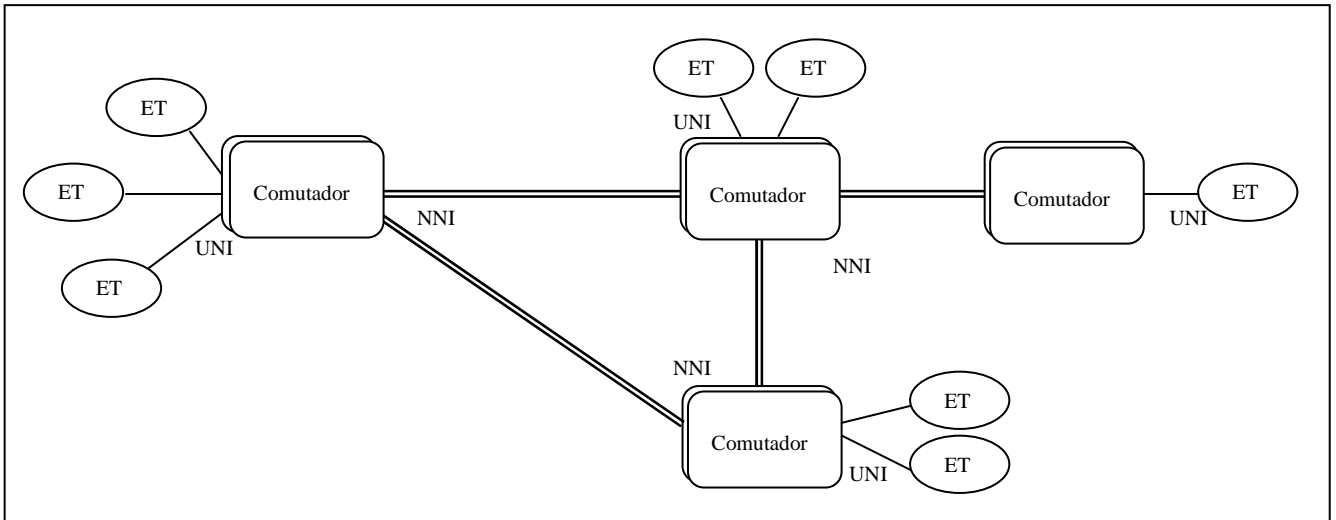


FIGURA 2.1 - Elementos da rede ATM

Os formatos da célula ATM na UNI e na NNI são mostrados na tabela 2.1. O campo GFC (*Generic Flow Control*) foi definido com o objetivo de prover o controle de fluxo na UNI para o tráfego originado no equipamento do usuário e enviado para a rede. Atualmente o GFC não está sendo utilizado. Os campos VPI (*Virtual Path Identifier*) e VCI (*Virtual Channel Identifier*) formam o rótulo da conexão utilizado pelos comutadores para encaminhar as células ao destino. O campo PTI (*Payload Type Indicator*) tem duas finalidades. A primeira, é indicar o tipo de informação contida na célula, podendo ser de usuário, sinalização ou de manutenção. A segunda finalidade é indicar se a célula recebida passou por congestionamentos durante seu percurso. O segundo bit do PTI é chamado EFCI (*explicit forward congestion indication*). O campo CLP (*Cell Loss Priority*) é formado por um único bit usado para indicar a prioridade caso haja a necessidade de descarte de células. A célula que for marcada com o CLP igual a 1 é candidata preferencial ao descarte. A marcação desse bit pode ser feita pela rede ou pela própria aplicação. No caso da rede a marcação é feita em todas as células que não estão em conformidade com o contrato estabelecido com a rede. A aplicação pode marcar determinadas células porque se essas não forem entregues não degradarão o serviço. O último campo do cabeçalho é chamado de HEC (*Header Error Check*). É utilizado para delimitação de célula e detecção de erros de transmissão que ocorreram nos campos que compõem o cabeçalho da célula. Erros que atingiram no máximo um único bit no cabeçalho podem ser corrigidos automaticamente. Excedendo o valor de um bit, a célula é descartada.

TABELA 2.1 - Campos do cabeçalho

Campo	UNI	NNI	Descrição
GFC	4 bits	----	Não tem significado fim a fim, não é utilizado
VPI	8 bits	12 bits	Identificador de caminho virtual
VCI	16 bits	16 bits	Identificador de circuito virtual
PT	3 bits	3 bits	Tipo de dados: do usuário ou de gerência
CLP	1 bit	1 bit	Indica célula sujeita a descarte, de baixa prioridade
HEC	8 bits	8 bits	Controle de erro do cabeçalho

Fonte: [CER97]

Devido às redes ATM serem orientadas a conexão, antes de qualquer transferência de dados deve ser estabelecido um circuito virtual através da rede. Os circuitos virtuais são disponibilizados de duas formas: manualmente pelos administradores da rede ou dinamicamente pelos usuários. Circuito virtual dinâmico é chamado SVC (*Switched Virtual Circuit*) e permanece ativo até a transferência de informação ser completada. Já o circuito virtual disponibilizado manualmente, chamado PVC (*Permanent Virtual Circuit*), permanece ativo independente da conexão estar sendo utilizada. Os procedimentos e mensagens que devem ser trocados entre a rede e os usuários para estabelecer SVCs são efetuados por um protocolo denominado sinalização. Genericamente, sinalização é a designação das normas para estabelecer, manter e encerrar conexões ATM. A sinalização, também chamada de fase de estabelecimento da conexão (*call setup*), precede o início da transmissão de tráfego. O ATM Forum especifica a sinalização UNI 4.0 para interfaces UNI [UNI4.0] e PNNI 1.0 para interfaces NNI [PNN1.0].

2.3.2 Contrato de Tráfego

Um usuário ATM requisita uma conexão ponto a ponto ou ponto multiponto através do protocolo de sinalização. Junto a essa requisição vem uma série de informações que caracteriza o tráfego a ser transmitido sobre a conexão e a qualidade de serviço requerida pelo usuário. Se a rede aceita atender a requisição do usuário, é firmado um acordo chamado de contrato de tráfego (fig. 2.2). Esse acordo compromete a rede em alcançar o desempenho (QoS) desejado pelo usuário para o tráfego que mantiver suas características conforme o acordado. Se alterações são necessárias, novo estabelecimento de conexão e contrato de tráfego são requeridos. O contrato de tráfego é um componente fundamental no controle de tráfego do ATM. O contrato de tráfego é composto pelas seguintes informações:

- **Parâmetros de Qualidade de Serviço requerida (QoS):**
 - Máximo atraso de transferência de célula (maxCTD);
 - Variação máxima de atraso de transferência da célula (ppCDV);
 - Probabilidade de perda de células (CLR);
- **Descritor de tráfego da conexão:**
 - Categoria de serviço;
 - Parâmetros de tráfego:
 - Taxa máxima de transmissão de células (PCR);
 - Taxa média de transmissão de células (SCR);
 - Número máximo de células transmitidas na taxa de PCR (MBS);

- Taxa mínima de transmissão de células (MCR);
- Tolerância da rede para tráfego que exceda o PCR (CDVT);
- Definição de conformidade;

Inicialmente, essas informações são utilizadas pela rede para avaliar a possibilidade de suportar a requisição do usuário sem prejuízo do nível de serviço das conexões já estabelecidas. Decidindo aceitar a nova conexão, a rede utiliza o contrato de tráfego para outra finalidade: o policiamento do fluxo recebido do usuário. Prevenindo-se da entrada de fluxos fora de conformidade, evita-se o dano que o excesso de tráfego causaria em outras conexões.

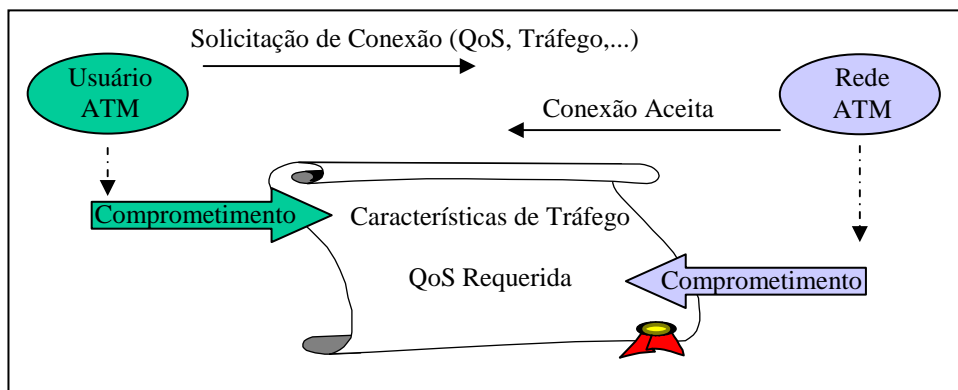


FIGURA 2.2 - Contrato de Tráfego

2.3.3 Descritor de Tráfego da Conexão

Descritor de tráfego da conexão é o nome dado ao conjunto de parâmetros originários dos usuários que especificam para a rede as características de fluxo da conexão. Durante o estabelecimento do circuito virtual, a rede utiliza esse descritor para avaliar os recursos necessários para aceitar o ingresso da conexão. Cada direção de fluxo de uma conexão pode ter seu descritor de tráfego, permitindo configurar circuitos virtuais ATM com características de tráfego assimétrico.

O descritor de tráfego da conexão, conforme [TM4.0], possui quatro parâmetros de tráfego mais o parâmetro CDVT que definem o comportamento da fonte. Esses cinco parâmetros são apresentados a seguir. Além dessas informações, o descritor de tráfego especifica também a categoria de serviço e a definição de conformidade que serão apresentados nas próximas seções deste capítulo.

PCR (*Peak Cell Rate*) - Taxa máxima de transmissão (largura de banda) da fonte medida em células por segundo.

SCR (*Sustainable Cell Rate*) - conceitualmente, representa a largura de banda média de longo prazo requerida. É a taxa média de transmissão de uma conexão. É dada em células por segundo.

MBS (*Maximum Burst Size*) - representa o número máximo de células que podem ser transmitidas na taxa de PCR. É dada em quantidade de células, mas por dedução, define o tempo máximo de duração da rajada.

MCR (*Minimum Cell Rate*) representa a largura de banda mínima requerida pela conexão e pode ser especificada como sendo quase zero. Independente do valor, uma vez aceito pela rede, esse será garantido. É dado em células por segundo.

CDVT (*Cell Delay Variation Tolerance*) - Tolerância de CDV, ou seja, variação máxima permitida para o atraso das células, usualmente informado em microssegundos. Mesmo com fontes de taxas constantes, o intervalo entre as células pode alterar-se devido aos atrasos provocados pela rede (caso de enfileiramento de tráfego). O intervalo para limitar essas alterações dentro de valores aceitáveis é dado pelo CDVT. Se o CDVT é definido como zero, significa que não há tolerância da rede caso a célula chegar antes ou depois do tempo de chegada esperado, sendo essa sujeita ao descarte. Esse parâmetro é utilizado para verificar a conformidade do tráfego.

Outros parâmetros que descrevem o comportamento do tráfego podem ser calculados a partir dos descritores informados. É o caso da tolerância de rajada (**BT** - *Burst Tolerance*), dada em segundos, utilizada para verificar a conformidade de tráfego VBR [GIR98]. O parâmetro BT é definido pela seguinte expressão:

$$BT = (MBS - 1) \times \left(\frac{1}{SCR} - \frac{1}{PCR} \right)$$

A quantidade de valores que constituem o descritor de tráfego depende do tipo de classe de serviço a ser utilizada (tab. 2.2). Definições de categorias de serviço e parâmetros de qualidade (QoS) serão vistos com detalhes nas próximas seções.

TABELA 2.2 - Descritores de Tráfego e QoS por Categoria de Serviço

Categoria de Serviço	Descritores de Tráfego					Parâmetros de QoS		
	PCR	SCR	MBS	MCR	CDVT	CLR	CDV	CTD
CBR	CT	-	-	-	CT	GR	GR	GR
VBR-rt	CT	CT	CT	-	CT	GR	GR	GR
VBR-nrt	CT	CT	CT	-	CT	GR	-	-
ABR	CT	-	-	Opc.	CT	-	-	-
UBR	CT	-	-	-	CT	-	-	-

CT - estabelecido no Contrato de tráfego

GR - Garantido pela Rede

Opc. - Opcional

2.3.4 Categorias de Serviços

Uma rede multisserviços suporta uma diversidade enorme de serviços. Apesar da diversidade, as conexões desses serviços podem ser agrupadas dentro de categorias conforme suas características de tráfego e expectativas de QoS. Dessa forma a rede pode implementar procedimentos de alocação de recursos e funções de controle de tráfego por categoria de serviço. O ATM Forum definiu um número de categorias de serviço baseada nos seus requerimentos de recursos da rede [TM4.0]. Algumas dessas categorias requerem requisitos de qualidade sensíveis ao atraso com taxas variáveis de transmissão, enquanto outras não. Alguns tipos de serviços podem tolerar um número moderado de perda de células, enquanto outros exigem perdas compatíveis com as

disponíveis em canais dedicados. Os parâmetros de QoS e de tráfego a serem declarados pelas aplicações dependem de sua categoria de serviço. O objetivo da rede é prover os recursos necessários de forma que a integridade de cada serviço seja mantida (fig. 2.3).

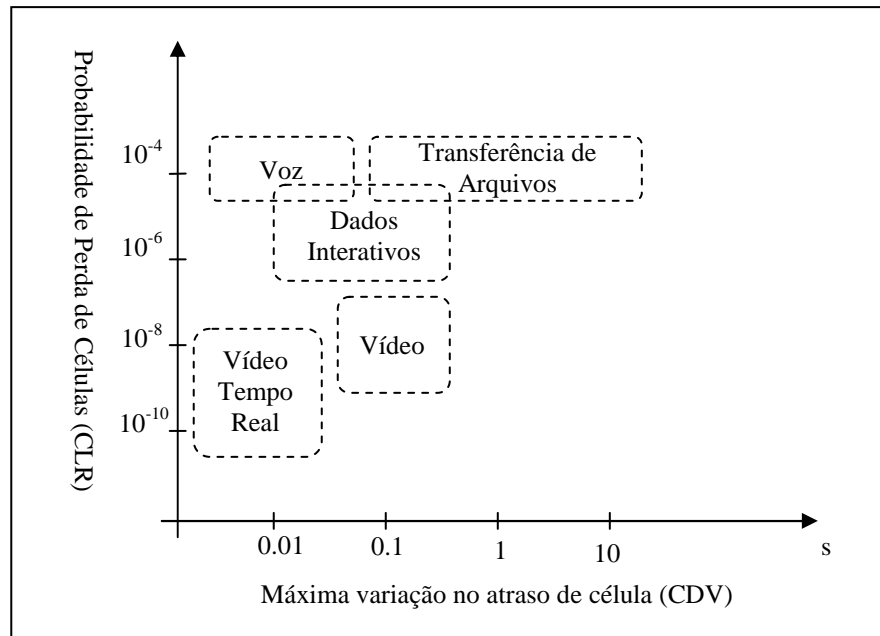


FIGURA 2.3 - Necessidades de QoS para diferentes aplicações

As categorias de serviço definidas pelo ATM Forum são:

Serviço CBR (*Constant Bit Rate*): é a categoria de serviço usada por aplicações e conexões que requerem uma quantidade fixa (constante) de largura de banda. A rede utiliza o parâmetro PCR estabelecido pelo usuário para determinar os recursos que estarão alocados enquanto durar a conexão. Nesse caso, todas as células estão sujeitas ao mesmo atraso. A fonte de tráfego pode emitir células abaixo da taxa de PCR, porém a largura de banda não utilizada não poderá ser aproveitada por outra conexão, o que acarretará desperdício de recursos da rede. A variação de atraso é praticamente zero. Desde que a fonte não ultrapasse o valor de PCR, o QoS é garantido. O controle de congestionamento é alcançado geralmente por meio do CAC (Connection Admission Control) na fase de estabelecimento da conexão. A rede aceita a requisição de chamada somente se pode prover a banda solicitada. Perda de células, ou grandes atrasos, tem um impacto negativo substancial nesse tipo de serviço. Incluídos nessa categoria de serviço estão aplicações sensíveis ao atraso e de tempo-real tais como voz, vídeo e emulação de circuitos de dados, como PDH E1 (2 Mbit/s).

Serviço VBR-rt (*Variable Bit Rate real-time*): é a categoria de serviço utilizada por aplicações cuja fonte gera tráfego com taxa variável de transmissão. A alocação de banda e recursos é feita considerando a taxa média de transmissão (SCR), a taxa máxima (PCR) e o tamanho máximo de rajada (MBS). A banda alocada é menor que o PCR e maior ou igual a SCR. A natureza de serviço de tempo real VBR-rt requer baixo atraso (CTD), baixa variação no atraso (CDV) e pequena taxa de perda de células (CLR), como a categoria CBR. Esses parâmetros são passados pelos usuários no estabelecimento da conexão. As fontes do serviço VBR geram tráfego, durante períodos

de atividade, na taxa máxima PCR. A duração dos períodos de transmissão e de silêncio é caracterizada pelo SCR. Conforme a relação PCR / SCR aumenta, maior é o comportamento de rajada² da fonte. A alternância randômica entre períodos de atividade e silêncio das fontes VBR minimiza a probabilidade de duas fontes colidirem nos períodos de pico de transmissão. A variação na taxa de transmissão permite diversas fontes VBR serem multiplexadas estatisticamente sobre os mesmos recursos otimizando a utilização da rede. Dessa forma, os serviços VBR-rt e VBR-nrt são projetados para os usuários compartilharem enlaces ATM de forma a propiciar o ganho estatístico na utilização do meio³. Aplicações multimídia de tempo real, como vídeo conferência, são exemplos dessa categoria de serviço.

Serviço VBR-nrt (*Variable Bit Rate non-real-time*): como VBR-rt, essa categoria de serviço requer largura de banda variável durante a duração da conexão. Entretanto, sendo a natureza não de tempo real, não há restrições quanto a atrasos (CTD) ou variações no atraso (CDV). Porém, é requerida uma baixa taxa de perda de células (CLR). Portanto, serviços VBR-nrt podem ser amplamente *bufferizados*. A rede determina os recursos alocados pelo PCR, SCR e MBS informados pelo usuário. Aplicações como correio eletrônico com imagem e vídeo sob demanda são exemplos para essa categoria.

Serviço ABR (*Available Bit Rate*): a categoria de serviço ABR possui um mecanismo adaptativo que modifica a taxa de transmissão conforme a largura de banda disponível na rede. A largura de banda dos enlaces físicos não utilizada pelas outras conexões fica automaticamente disponível para os serviços ABR. Portanto, a taxa de transmissão nessa categoria de serviço é dinâmica conforme a demanda da aplicação e a disponibilidade da rede. A rede utiliza células chamadas RM (Resource Management) para notificar a fonte de tráfego ABR sobre o estado da rede. Quando a fonte recebe uma célula RM indicando congestionamento, essa deve imediatamente reduzir sua taxa de transmissão. Recebendo células RM indicando não haver congestionamento, a fonte aumenta a utilização da banda. Essa interação entre as células RM e a fonte ABR resulta em um apropriado ajuste na taxa de transmissão para evitar congestionamentos. Para ter certeza que a fonte está obedecendo as requisições de redução das células RM, é necessário o policiamento do tráfego por parte da rede. O serviço ABR garante uma taxa mínima de transmissão (MCR) e pode ser limitado por uma taxa máxima (PCR). O mecanismo de controle de fluxo na fonte de tráfego minimiza o potencial de perda de células. Esse serviço foi projetado para acomodar tráfego de dados que requer uma garantia específica em relação à taxa de perda de células (CLR), no entanto não possui requisitos quanto ao atraso de transferência (CTD) ou sua variação (CDV). Exemplo de aplicação que pode utilizar esse serviço é a transferência de arquivos.

Serviço UBR (*Unspecified Bit Rate*): a categoria de serviço UBR foi projetada tendo em mente um modelo muito similar ao das redes locais tradicionais onde não existe previsão de taxas de tráfego e de QoS desejadas. É muito simples em termos de controle de tráfego. Conexões UBR compartilham a largura de banda sem utilizar

² Transmissão de bloco de células com taxa fixa de transmissão. Os blocos são separados por períodos de inatividade da fonte. Quanto maior a duração dos períodos de inatividade em relação aos períodos de transmissão, maior é o comportamento de rajada.

³ O ganho estatístico está na proporção do número de conexões suportadas pela multiplexação estatística em relação ao número de conexões suportadas pela multiplexação determinística (pela taxa de pico de transmissão). A multiplexação estatística será detalhada no capítulo seguinte.

qualquer mecanismo de controle de congestionamento. O usuário não informa nenhum tipo de parâmetro descritor de tráfego. Portanto, a rede não garante nenhuma qualidade de serviço (QoS). A banda disponível para serviços UBR é a restante na rede depois de atender todas as outras conexões. É como no serviço ABR, com a diferença de não possuir qualquer controle de fluxo. Células transmitidas acima da banda disponível são descartadas. As informações são transmitidas pelas fontes UBR entendendo que a rede fará o melhor para levá-las ao destino, porém isso pode não acontecer. Protocolos superiores à rede ATM, como TCP, devem se preocupar em retransmitir as informações perdidas. Esse comportamento caracteriza o serviço UBR como sendo de melhor esforço (*best-effort*). Mecanismos de descarte de células seletivos foram propostos, como EPD (Early Packet Discard) [RF95], para minimizar a retransmissão de pacotes devido ao descarte de células. O serviço UBR é atrativo porque é muito simples de ser suportado na rede, não há reserva de recursos e as fontes não necessitam conhecer suas características de tráfego. Nas aplicações sem pressão de entrega que querem fazer seu próprio controle de erros e de fluxo, a UBR é uma escolha razoável. Exemplo de aplicação de serviços UBR são transferência de arquivos e correio eletrônico.

Devido à possibilidade de oferecer os requisitos mínimos solicitados pelas conexões, o ATM pode ser utilizado para suportar todos os diferentes tipos de tráfego de telecomunicações. A figura 2.4 mostra os requisitos de QoS e o mecanismo de controle de fluxo oferecido para as categorias de serviços.

	CBR	VBR-rt	VBR-nrt	ABR	UBR
CLR	Especificado				Não Esp.
CTD	Especificado			Não Especificado	
CDV	Especificado		Não Especificado		
Controle Fluxo	Não			Sim	Não

FIGURA 2.4 - Qualidade oferecida pela Rede ATM

Para uma dada categoria de serviço, uma rede pode oferecer uma ou mais classes de serviço. Uma classe de serviço pode especificar valores de QoS e limitar o intervalo de alguns parâmetros descritores de tráfego. Dessa forma é possível oferecer diferentes níveis de serviços e tarifas. Quanto maior a qualidade, maior o valor a pagar pelo serviço.

2.3.5 Definição de Conformidade

A qualidade de serviço (QoS) é assegurada para todo o tráfego de células que estiver em conformidade com o contrato de tráfego. A rede faz o monitoramento das conexões considerando determinados parâmetros descritores de tráfego, como PCR ou SCR (fig. 2.5). Os parâmetros utilizados são determinados pela categoria de serviço da conexão. O monitoramento pode agir somente sobre o fluxo de células do usuário com direito à QoS (CLP=0) ou agir indiscriminadamente sobre todo o fluxo agregado (CLP=0 + CLP=1). As células que passam pelo monitoramento e não satisfazem as

condições de tráfego são consideradas não conformes e a rede deve descartá-las ou marcá-las (CLP=1), conforme a definição de conformidade passada pelo usuário no estabelecimento da conexão (fig. 2.6).

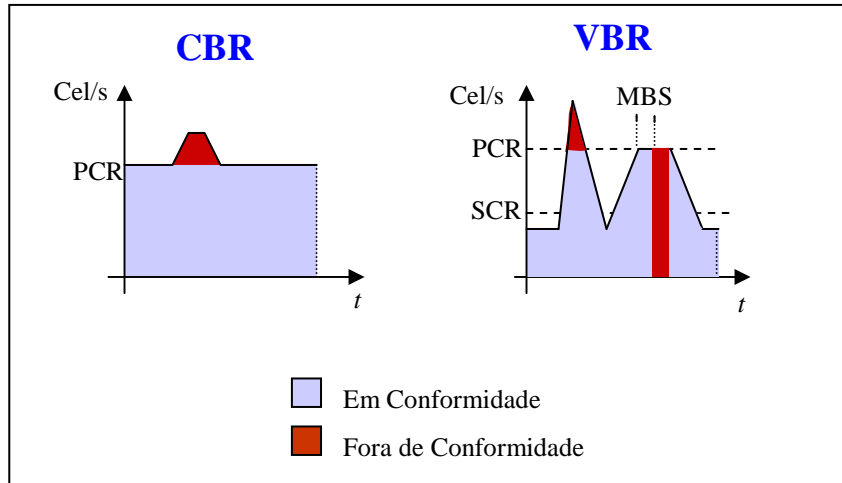


FIGURA 2.5 - Conformidade de tráfego CBR e VBR

Conforme visto, para fazer o monitoramento é necessário determinar os parâmetros descritores de tráfego a serem utilizados, o tipo de fluxo (CLP=0 ou CLP=0+1) a ser analisado e a ação a ser tomada sobre as células não conformes. Essas especificações são feitas durante o estabelecimento da conexão através do parâmetro de usuário chamado definição de conformidade.

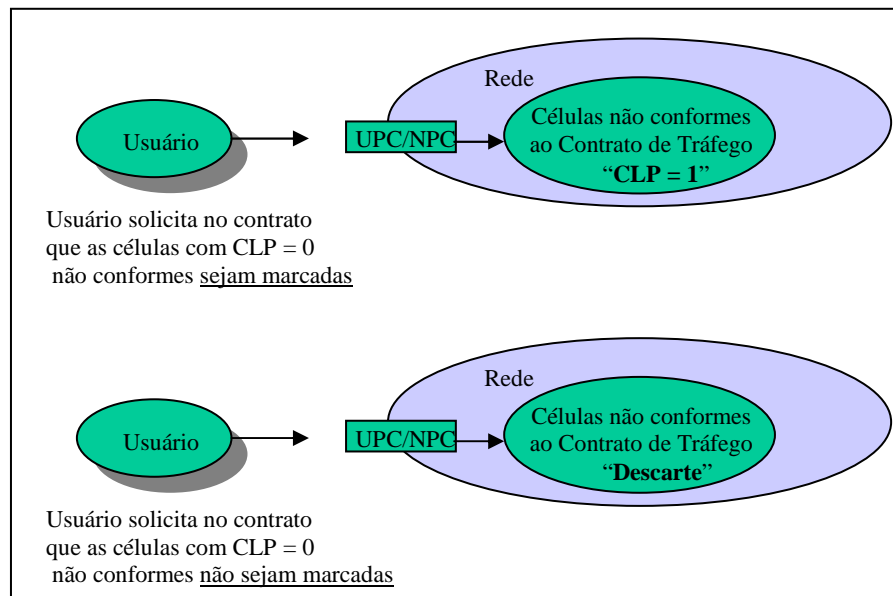


FIGURA 2.6 - Ação tomada conforme definição de conformidade

A definição de conformidade determina a forma de selecionar para quais células serão garantidos os parâmetros de QoS requeridos e como as células não conformes serão tratadas, ver tabela 2.3. Células com CLP=1 tornam-se inelegíveis para as garantias de QoS, mas opcionalmente elas ainda podem alcançar o destino se a rede não sofrer congestionamento. Quando a garantia de QoS é aplicada ao tráfego agregado

(CLP=0+1), células com CLP=0 e CLP=1 são tratadas igualmente, sem diferença de prioridade. Esse caso é chamado de definição de conformidade com CLP transparente.

TABELA 2.3 - Definições de conformidade por categoria de serviço

Definição de Conformidade	Categoria de Serviço	Fluxo Monitorado			Ação Células Não Conforme	QoS Garantida		
		PCR	SCR	MCR		CLR	Max-CTD	P2P-CDV
CBR.1	CBR	0+1	Não	Não	Descarte	0+1	0+1	0+1
VBR.1	VBR-rt/nrt	0+1	0+1	Não	Descarte	0+1	0+1(rt)	0+1(rt)
VBR.2	VBR-rt/nrt	0+1	0	Não	Descarte	0	0 (rt)	0 (rt)
VBR.3	VBR-rt/nrt	0+1	0	Não	Marca	0	0 (rt)	0 (rt)
ABR.1	ABR	0	Não	0	Descarte	0	Não	Não
UBR.1	UBR	0+1	Não	Não	Descarte	Não	Não	Não
UBR.2	UBR	0+1	Não	Não	Marca	Não	Não	Não

Fonte: [GIR98]

2.4 Qualidade de Serviço (QoS) - Medidas de Desempenho

Existe um conjunto de informações que define o desempenho fim a fim desejado por uma aplicação sobre a rede ATM. São chamadas métricas de qualidade de serviço (QoS). A Qualidade de Serviço é definida na recomendação E.800 do ITU-T como sendo o efeito coletivo do desempenho do serviço e que determina o grau de satisfação do usuário desse serviço. A rede utiliza esses valores para determinar se a conexão pode ou não ser admitida. Seis parâmetros de QoS são usados para medir o desempenho da rede para uma dada conexão [I.356]. Três desses podem ser negociados pela aplicação com a rede, são eles:

- Razão de Células Perdidas (CLR);
- Atraso de Transferência de Célula (CTD);
- Variação de Atraso de Transferência da Célula (CDV).

Os três outros parâmetros não são negociáveis e são características da configuração da rede, são eles:

- Razão de Células Erradas (CER);
- Taxa de Células Inseridas Incorretamente (CMR);
- Razão de Blocos de Células Severamente Errados (SECBR).

São diversos os possíveis resultados da transferência de uma célula numa rede ATM. Basicamente são os seguintes:

- Transferência com sucesso: a célula é recebida sem erros dentro de um tempo máximo especificado;
- Célula com erro: a célula é recebida dentro do intervalo de tempo especificado, mas o conteúdo foi alterado ou o cabeçalho é inválido;
- Célula perdida: a célula não chegou dentro do tempo especificado ou foi descartada;
- Célula intrusa: a célula recebida não faz parte dessa conexão, foi inserida por alteração errônea do cabeçalho.

Os aspectos de qualidade de serviço podem ser observados e medidos no ponto no qual o serviço é acessado pelo usuário. A seguir detalharemos a forma como são considerados e medidos os parâmetros de QoS.

2.4.1 CLR - Razão de Células Perdidas

A perda de células ocorre por causa do esgotamento de espaço em filas (*buffers*) devido a chegadas simultâneas de diferentes conexões. O escalonador do CAC e as estratégias de filas podem ter um efeito na quantidade de células perdidas (descartadas). As células também podem ser descartadas por erros de transmissão que afetaram o cabeçalho da célula e não puderam ser corrigidos. A razão de células perdidas em uma conexão é dada por:

$$CLR = \frac{\text{Células Perdidas}}{\text{Total de Células Transmitidas}}$$

Onde,

Células Perdidas incluem:

- Células conformes que não alcançaram o destino, ou seja, que foram descartadas;
- Células que foram recebidas com cabeçalho inválido;
- Células entregadas tarde demais. São células com tamanho atraso que perdem a utilidade para o caso de tráfego em tempo real.

Total de Células Transmitidas é o número de células conformes transmitidas no período. O CLR não conta as células que não estão em conformidade com os descritores de tráfego assumidos pela aplicação com a rede.

O intervalo de tempo de medição não é padronizado, mas, geralmente, é representado como o período de duração da conexão.

2.4.2 CTD - Atraso de Transferência de Célula

A medida de tempo para a transferência de célula é definida como o período consumido para levar a célula de um usuário ao outro. Detalhando, é o tempo que começa a contar a partir da saída do primeiro bit da célula na origem até a chegada do último bit da célula no destino. Os pontos de referência de saída e de chegada são as interfaces UNI dos equipamentos do usuário (fig. 2.7).

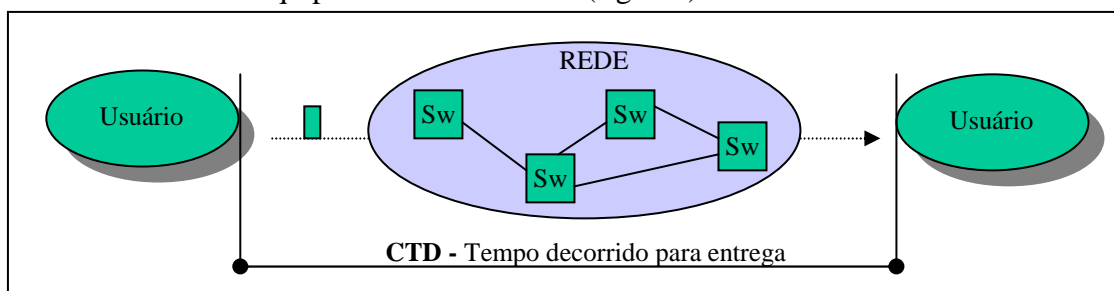


FIGURA 2.7 - Tempo de transferência da célula de um usuário a outro

O atraso para a transferência de células de uma mesma conexão não é fixo. Diversos componentes de retardo são responsáveis pelo atraso da célula. São eles retardo de propagação, retardo de transmissão, retardo de comutação e retardo de enfileiramento (fig. 2.8).

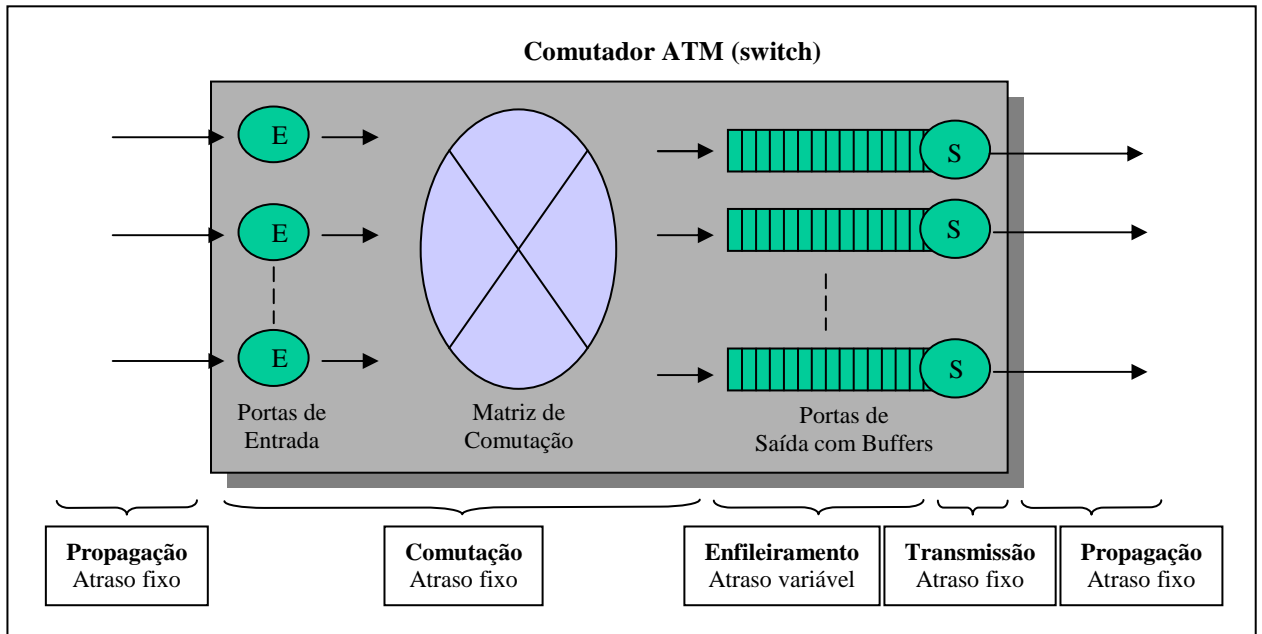


FIGURA 2.8 - Fatores que compõem o atraso na entrega da célula

O atraso para levar uma célula de uma ponta a outra em uma conexão é dado como:

$$CTD = D + W_i$$

Onde,

D é a constante resultante dos atrasos de propagação, transmissão e comutação.

W_i é o componente variável do atraso que reflete o tempo que a célula permaneceu em filas para ser transmitida.

O atraso através de uma rede é o resultado dos retardos causados pelos elementos presentes no caminho da célula. Existem elementos cujos retardos são constantes para todas as células que o atravessam. São os retardos causados pela propagação, transmissão e comutação. Portanto, o menor valor de CTD para qualquer célula de uma mesma conexão é dado pela soma dos atrasos fixos. O atraso de uma célula possui um componente variável dependente do tráfego encontrado no caminho. Células oriundas de diversas fontes competem por recursos comuns limitados, espaço no *buffer* do multiplexador, o que causa o retardo de enfileiramento. As filas são encontradas nas portas de saída dos equipamentos dos usuários e nos comutadores da rede. As células que permanecem mais tempo em filas aguardando sua vez para serem transmitidas terão um atraso maior do que aquelas que encontraram filas menores.

A seguir são apresentados os fatores componentes do CTD.

- **Atraso de propagação:** ocorre devido ao tempo utilizado para propagar o sinal no meio físico. O tempo de propagação depende do tipo de meio físico utilizado e a distância percorrida pelo sinal. É fixo para determinado caminho que a célula percorre. O retardo de propagação final é o somatório de todos os enlaces por onde passou. A tabela 2.4 mostra atrasos conforme o meio de propagação.

TABELA 2.4 - Tempos de Propagação

Meio de Propagação	Tempo de Propagação
UTP	4 μ s/Km
Cabo Coaxial	4 μ s/Km
Fibra	5 μ s/Km
Satélite	360 ms/36.000 Km

Fonte: [Onv95]

- **Atraso de transmissão:** é o tempo necessário para completar a transmissão de uma quantidade de bits para um enlace. Como o tamanho da célula é fixo (53 bytes), o retardo devido à transmissão depende unicamente da largura de banda do enlace. A tabela 2.5 mostra o tempo necessário para inserir uma célula na rede com diferentes interfaces.

TABELA 2.5 - Tempo de transmissão de uma célula (424 bits)

Interface	Taxa (Mbit/s)	Taxa Real (Mbit/s) ⁴	Atraso (μ s) = μ s/célula
PDH E1	2,048	2,048	207
PDH E3	34,368	33,730	12,5
SDH STM-1	155,52	149,76	2,83
SDH STM-16	2.488,32	2.396,16	0,17

- **Atraso de comutação:** é o tempo total que o *switch* leva para comutar uma célula de uma porta de entrada para outra de saída. O procedimento é analisar o cabeçalho da célula, consultar tabelas de comutação para determinar a porta de saída e entregar a célula na fila da porta de saída. O tempo gasto nesse procedimento, conhecido por latência interna, depende da arquitetura da matriz de comutação implementada no equipamento. A tabela 2.6 traz a comparação da latência interna e de outros parâmetros de desempenho alcançados por comutadores disponíveis atualmente no mercado.

TABELA 2.6 - Informações de comutadores disponíveis no mercado

Fabricante	Switch	(I) Latência Interna	(II) Latência de Chamada	(III) Chamada/s	(IV) Quantidade VCs	(VI) Matriz de Comutação
3Com	CB9000	10 μ s	500 ms	62	36.000	30 Gbit/s
Cabletron	Ssw 6500	11 μ s	8 ms	250	64.000	10 Gbit/s
Marconi	ASX-1000	12 μ s	2 ms	385	25.000	10 Gbit/s

Fonte: Documentação dos fabricantes, 1999/2000.

⁴ Excluído o overhead da estrutura de quadro do PDH e SDH.

Onde na tabela,

- I - Tempo interno para comutar uma célula. Retardo de comutação.
- II - Tempo para estabelecer uma conexão, considerando um *switch*.
- III - Quantidade máxima de estabelecimento de conexões por segundo.
- IV - Quantidade máxima suportada de circuitos virtuais simultâneos.
- VI - Capacidade de comutação *full-duplex*.

- **Atraso de enfileiramento:** é o somatório do tempo de permanência da célula em cada ponto de fila (*buffer*) ao longo do caminho percorrido. Os *buffers* são utilizados para resolver a contenção quando mais de uma célula chega simultaneamente em uma porta de saída. Os valores de atraso de enfileiramento variam de uma célula para outra conforme a carga no ponto de fila. Por ter essa característica variável, é o principal componente a ser considerado na avaliação de atraso em redes de comutação de células. O atraso de enfileiramento não depende unicamente da quantidade de células correntemente no *buffer*, mas também da capacidade de transmissão da porta de saída da fila (fig. 2.9). Portanto, o tempo máximo possível de permanência na fila de uma célula é dado pela seguinte expressão:

$$\max CTDq = B \times \frac{1}{TE}$$

Onde,

$\max CTDq$ é o valor máximo de atraso para essa fila.

B é a quantidade possível de células que a fila pode armazenar.

TE é a taxa de transmissão do enlace, dado em células por segundo.

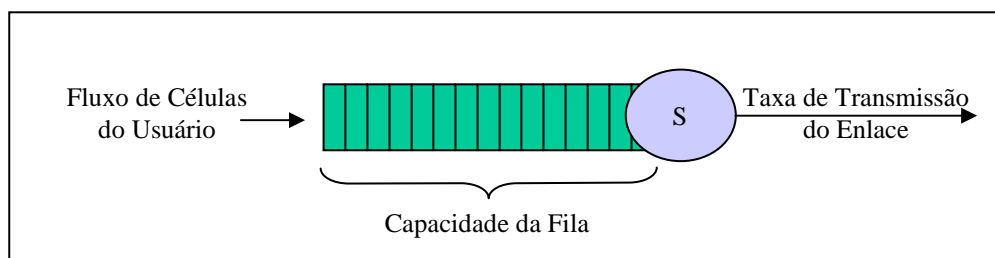


FIGURA 2.9 - Atraso de Enfileiramento

As exigências de máximo CTD como parâmetro de QoS pode variar de uma aplicação para outra. O CTD pode afetar negativamente aplicações com interatividade, como vídeo conferência. Com grandes atrasos, a conversação em uma vídeo conferência tende a colidir, forçando os participantes a utilizar longas pausas, o que descaracteriza uma conversação normal. A tabela 2.7 mostra valores de atraso tolerados para diferentes aplicações.

TABELA 2.7 - Exemplos de Requisitos de CTD e CDV

Aplicação	Taxa de Pico	Atraso (CTD)	Variação do Atraso (CDV)
Voz Comprimida	16 Kbit/s	30 ms	130 ms
Vídeo Conferência	64 Kbit/s	250 ms	130 ms
Vídeo Conf. MPEG	256 Kbit/s	7 ms	9,1 ms
Vídeo NTSC MPEG	1,5 Mbit/s	5 ms	6,5 ms
Vídeo HDTV	20 Mbit/s	0,8 ms	1 ms

Fonte: [Onv95]

O máximo CTD (maxCTD) especificado para uma conexão é definido como uma garantia estatística usando o parâmetro de probabilidade α , onde α é um valor muito pequeno, na ordem de 10^{-6} . A rede tem que garantir o maxCTD com uma probabilidade de $1 - \alpha$. O máximo CTD representa, dessa forma, o limite na função de densidade de probabilidade, onde as células com atrasos superiores são consideradas perdidas ou inúteis (fig. 2.10). Portanto, a probabilidade da célula exceder o máximo CTD não pode ser superior ao valor de CLR (probabilidade de perda requerida). O parâmetro α pode ser visto como a probabilidade de perda de células que pode ser tolerado pela aplicação devido a chegadas muito atrasadas. Na figura 2.10, o atraso mínimo é a soma dos componentes do atraso de característica fixa: transmissão, propagação e comutação.

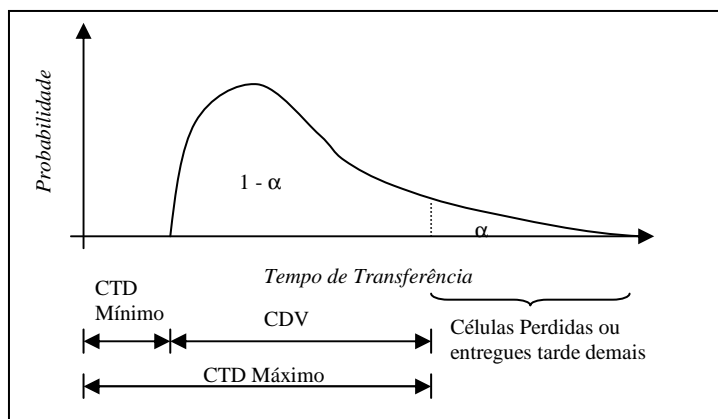


FIGURA 2.10 - Distribuição da probabilidade do CTD

2.4.3 CDV - Variação do Atraso de Célula

Devido à multiplexação estatística do ATM, o tempo de permanência em filas (*buffers*) varia de uma célula para outra. Essa variação é referida como CDV (*Cell Delay Variation*), também chamada de *jitter*. Um fluxo de células com taxa de transmissão constante chegando em uma fila congestionada para transmissão pode resultar em uma seqüência de rajadas de células na saída.

Conceitualmente, o CDV é dado como a diferença entre o atraso experimentado por duas células consecutivas, como mostra a expressão:

$$CDV = (D + W_{i+1}) - (D + W_i)$$

Onde,

D é a constante que inclui os atrasos de propagação, transmissão e comutação. É um valor igual para todas as células da conexão em questão.

W_i é o valor de atraso resultante das filas por onde passou a célula "i" da conexão.

W_{i+1} é um componente do atraso variável da célula que segue a de número "i". É um valor independente do W_i .

CDV é medido em unidades de tempo e pode ter resultados positivos e negativos. Um valor positivo ocorre quando o tempo entre as células decrementa (fig. 2.11). E um valor negativo ocorre quando o tempo entre células incrementa. O CDV é zero quando não há modificação no tempo de atraso das células.

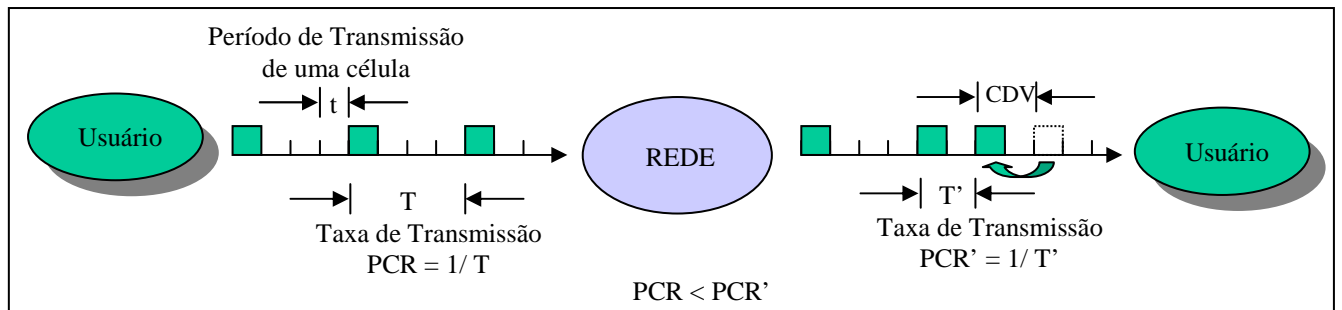


FIGURA 2.11 - A variação no tempo de chegada da célula

O CDV é utilizado como requisito de QoS no estabelecimento da conexão. O valor informado é definido como a diferença do maior e menor CTD entre todas as células da conexão. Esse parâmetro é chamado de CDV de pico a pico (ppCDV). Segue a fórmula:

$$ppCDV = CTD_{\max} - CTD_{\min}$$

Essa métrica permite avaliar a variação máxima possível entre o tempo de chegada de duas células consecutivas que foram espaçadas durante o percurso.

Como medida de desempenho da rede, o CDV pode ser avaliado célula a célula de duas formas: CDV de um ponto e CDV de dois pontos.

O CDV de um ponto mede a variabilidade da chegada da célula em relação ao PCR da fonte. Isso inclui a variabilidade da fonte e o efeito cumulativo da variabilidade introduzida pela rede [I.356]. Cada tempo de chegada de célula é comparado com seu tempo de referência, o qual é representado pelo tempo de chegada da célula anterior mais o tempo entre células esperado ($1/PCR$). Segue a expressão para cálculo do CDV de um ponto:

$$CDV1P_k = c_k - a_k$$

Onde,

$CDVIP_k$ é o CDV de um ponto da célula k ;
 c_k é o tempo de chegada de referência da célula k ;
 a_k é o tempo de chegada real da célula k .

O tempo de referência (c_k) é definido como segue, sendo $c_1 = a_1$.

Se $c_k \leq a_k$, então:

$$c_{k+1} = a_k + \frac{1}{PCR}$$

Senão:

$$c_{k+1} = c_k + \frac{1}{PCR}$$

O CDV de dois pontos mede a variabilidade do tempo de chegada das células com referência à variabilidade gerada pela fonte. Esse inclui somente a variabilidade introduzida pela rede. A medida é tomada entre dois pontos (MP1 e MP2) de referência na rede, por exemplo, as UNI de entrada e saída. Para a célula k , o CDV de dois pontos é dado pela seguinte expressão:

$$CDV2P_k = CTD_k - CTD_{ref}$$

Onde,

$CDV2P_k$ é o CDV de dois pontos da célula k ;
 CTD_k é o CTD da célula k , medido entre o ponto MP2 e MP1;
 CTD_{ref} é o CTD de referência que é definido como sendo o CTD calculado para a primeira célula da conexão.

Visto que as células de usuário não carregam a hora de saída, é muito difícil de medir o CDV de dois pontos. Essa definição ainda está em estudo no ITU-T.

O parâmetro de qualidade CDV é facilmente confundido com a tolerância na variação de atraso de células (CDVT), o qual é um descritor de tráfego para estabelecimento e policiamento da conexão. O CDVT é o valor máximo tolerado de CDV para uma dada conexão, ultrapassando esse valor o tráfego é considerado não conforme.

O CDV tem um impacto negativo no desempenho das aplicações que não se adaptam bem às modificações na taxa de recebimento das células. Por exemplo, um decodificador que está mostrando um vídeo com 30 quadros por segundo espera receber da rede um fluxo consistente. Entretanto, se a taxa de recebimento decrementa, o decodificador deverá reduzir a quantidade de quadros por segundo, prejudicando a qualidade do vídeo. A tabela 2.7, na seção anterior, mostra valores de variação de atraso tolerados para diferentes aplicações.

2.4.4 CER - Razão de Células Erradas

A razão de células erradas é influenciada pelas características físicas do meio, como distância e técnica de transmissão utilizada. CER é definido para uma conexão como segue:

$$CER = \frac{\text{Células Erradas}}{\text{Total de Células}}$$

Onde,

Células Erradas são todas aquelas que sofreram erros de transmissões que modificaram o cabeçalho ou a carga útil da célula e não foi possível recuperá-las através de técnicas de correção.

Total de Células incluem as células transmitidas com sucesso e as células erradas.

2.4.5 CMR - Taxa de Células Inseridas Incorretamente

As alterações dos campos VPI e VCI no cabeçalho da célula devido a erros de transmissão podem gerar o ingresso equivocado de células no fluxo de outra conexão. A situação ocorre na combinação de dois eventos. Quando a modificação do VPI/VCI pelo erro não é detectada e existe uma conexão com os mesmos valores de VPI e VCI resultantes. Caso não exista uma conexão com esses valores, a célula é descartada. Por definição, uma célula inserida incorretamente é uma célula recebida no destino que não foi transmitida na origem da conexão.

CMR é calculado para uma conexão como segue:

$$CMR = \frac{\text{Células Erradas}}{\text{Intervalo de Tempo}}$$

Onde,

Células Erradas são células carregadas por canais virtuais e que originalmente não pertenciam a eles.

Note que esse parâmetro é definido como taxa em vez de razão, pois o mecanismo que produz a inserção equivocada é independente do número de células transmitidas sobre a conexão. Não há relação entre as células transmitidas com as células inseridas incorretamente.

2.4.6 SECBR - Razão de Blocos de Células Severamente Errados

Esse parâmetro de desempenho fornece um meio de quantificar surtos de falhas em transferência de células e evitar que tais surtos influenciem os valores para CER, CLR e CMR. Portanto, as células de um bloco severamente errado são desconsideradas nos cálculos dos parâmetros de desempenho.

Entende-se por bloco de células a seqüência de N células de usuários transmitidas consecutivamente em uma conexão. Os blocos são limitados por células de OAM (*Operation and Maintenance*) [I.610]. A quantidade de células que formam o bloco é dada pela expressão:

$$N = \frac{PCR}{25}$$

Onde,

PCR é a taxa máxima de transmissão da conexão em células por segundo.

O valor de N é arredondado para a próxima potência superior de 2.

Bloco de Células Severamente Errado (SECB) é o bloco recebido com mais que M células erradas, perdidas ou inseridas incorretamente. O valor de M é aproximadamente 3% do tamanho do bloco, conforme expressão:

$$M = \frac{N}{32}$$

Onde, N é o tamanho do bloco para a conexão em questão.

Por fim, a razão de blocos de células severamente errada (SECBR) é definida como segue para uma conexão:

$$\text{SECBR} = \frac{\text{Blocos Severamente Errados}}{\text{Total de Blocos Transmitidos}}$$

2.5 Controle de Tráfego - TM (*Traffic Management*)

Para garantir qualidade de serviço para aplicações multimídia, a reserva de recursos é necessária. Adicionalmente, controle de admissão de conexões é requerido para impedir que novas conexões ingressem na rede quando a carga de serviço se torna demasiada, preservando, dessa forma, a qualidade de serviço das conexões já estabelecidas na rede. Podemos dizer que os principais objetivos do controle de tráfego da camada ATM são:

- Prover previsível comportamento para auxiliar a atingir os objetivos de qualidade de serviço;
- Minimizar congestionamento na rede;
- Maximizar o uso eficiente dos recursos da rede.

O método de serviço orientado à conexão é essencial na implementação de um mecanismo para reserva de recurso e admissão de conexões para garantir QoS. Isso é alcançado através da inclusão de informações tais como largura de banda e requisitos de QoS na mensagem recebida pela rede na fase de estabelecimento de conexão. A rede, então, deve determinar se possui recursos suficientes para aceitar a nova conexão. Os recursos da rede incluem largura de banda de transmissão e capacidade de *buffer* disponível no caminho percorrido pela conexão.

Entretanto, reserva de recurso e controle de admissão, embora necessários, não são suficientes para garantir QoS para cada aplicação. Eles lidam com a garantia de QoS durante o processo de estabelecimento da conexão. Outros mecanismos são necessários para garantir QoS durante a fase de transferência de células entre a origem e o destino.

Os mecanismos de controle de tráfego dividem-se em duas categorias: os baseados em conexão e os baseados em célula-a-célula. Os mecanismos de controle baseados em conexão são executados durante a fase de estabelecimento da conexão (tab. 2.8). Os baseados em célula-a-célula atuam nos processos de recebimento, tratamento e envio de cada célula das conexões, ou seja, na fase de transferência.

TABELA 2.8 - Controle de Tráfego baseado em conexão e célula-a-célula

Mecanismo de Controle de Tráfego	Atuação
Controle de Admissão de Conexões (CAC)	Estabelecimento da conexão
Controle de Recursos de Rede (NRM)	Estabelecimento da conexão
Policiamento de Tráfego (UPC/NPC)	Transferência de célula
Formatação de Tráfego	Transferência de célula
Controle de Fluxo ABR	Transferência de célula
Controle de Filas e Escalonador	Transferência de célula
Descarte Seletivo	Transferência de célula

2.5.1 CAC - Controle de Admissão de Conexões e NRM - Controle de Recursos da Rede

Uma conexão ATM atravessa um conjunto de nós de comutação que possuem internamente filas (*buffers*) para transmissão das células pelos enlaces. Para estabelecer a conexão por esses nós de comutação é necessário reservar recursos das filas para garantir a qualidade de serviço contratada. Os recursos da fila compreendem quantidade de espaço para receber células (tamanho do buffer) e a largura de banda requerida para servir a fila. O conjunto de procedimentos que determinam se a rede admite uma nova conexão são chamados CAC (*Connection Admission Control*).

CAC é executado pelos comutadores ATM quando recebem solicitações de estabelecimento de conexões. O CAC utiliza os descritores de tráfego para fazer a melhor estimativa quanto à capacidade necessária para suportar o tráfego. Então o CAC solicita ao controlador de recursos da rede (NRM - *Network Resource Management*) que verifique se existem recursos adequados na rede para suportar a requisição de qualidade de serviço. Com a informação do NRM, o CAC avalia se a nova requisição não afetará as conexões correntes já estabelecidas. Para cada nova conexão aceita, O NRM faz a alocação dos recursos ao longo do caminho. A avaliação do CAC/NRM é decomposta para cada enlace (ponto de fila) que forma o caminho da origem ao destino da conexão.

Os procedimentos adotados pelo CAC dependem da categoria de serviço (CBR, VBR, ABR e UBR) da conexão, pois esses possuem critérios de QoS diferenciados. Um eficiente CAC produz o máximo de ganho estatístico sem violar o QoS desejado.

De forma geral podemos descrever os procedimentos do CAC da seguinte forma:

- Posicionar os descritores de tráfego da conexão dentro de um modelo apropriado de tráfego. Como cada categoria de serviço tem seus específicos descritores de tráfego, diferentes modelos de tráfego são necessários para implementar o CAC;
- Utilizar o modelo de tráfego junto ao modelo de fila utilizado pelo comutador para estimar se há recursos suficientes para aceitar a conexão;
- Se disponíveis, alocar recursos e admitir a conexão.

Portanto, a eficiência do CAC depende muito da capacidade dos modelos de tráfego e de fila representarem com fidelidade a realidade de comportamento desses elementos. A precisão de alguns dos modelos, como VBR, intensifica a complexidade computacional. Esse é um limitante para implementar CAC nos comutadores, pois a decisão de aceitar ou não uma conexão é feita em tempo real. O tempo de execução do CAC afeta diretamente o tempo utilizado para o estabelecimento de uma conexão. Quanto menor o tempo do CAC maior será o desempenho do comutador em estabelecer conexões por segundo, sendo essa uma das medidas utilizadas para comparar o desempenho dos equipamentos. A tabela 2.6 mostra esses valores para alguns fabricantes.

Não havendo tempo suficiente para atingir o nível desejado de complexidade e precisão, é comum adotar modelos de tráfego conservadores. Esses modelos, em nome da garantia de QoS, fazem alocações de recursos superiores aos realmente necessários. O impacto dessa medida é o aumento da possibilidade de recusar novas conexões devido à estimativa exagerada de recursos alocados pelas conexões admitidas. O grau de conservadorismo dos modelos determina o nível de eficiência do CAC, ou seja, a capacidade de garantir o QoS das conexões evitando a subutilização da rede (fig. 2.12).

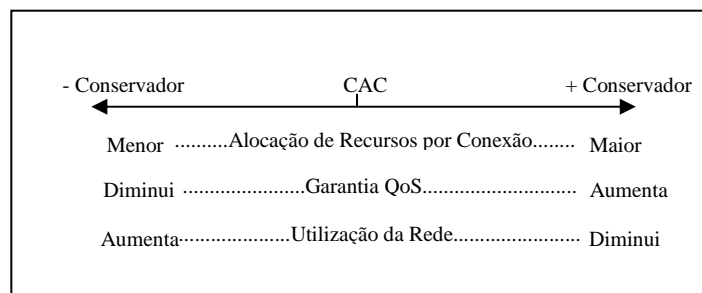


FIGURA 2.12 - Eficiência CAC: garantia QoS vs. utilização da rede

Em geral, os modelos para alocação de recursos assumem os parâmetros da fonte de tráfego conforme os parâmetros declarados no descritor. Algumas fontes que não conhecem seus parâmetros com antecedência tendem a declarar seu descritor de tráfego com alguma margem de segurança para evitar perda de qualidade. Essa situação, mais o comportamento não estático das fontes, pode causar a subutilização da rede. Para contornar esse inconveniente foram desenvolvidos procedimentos de CAC para

alocação adaptativa de recursos baseados em medições em tempo-real. Esses procedimentos medem certos recursos em tempo-real e tentam estimar se os objetivos de QoS poderão ser mantidos se uma nova conexão for admitida. [COU95] propõe um procedimento para monitorar o tráfego em um ponto de fila e tomar decisões de admissões de mais conexões baseado na fração de células descartadas na fila. O método em [DJM97] usa tanto os descritores de tráfego como a medição de tráfego para estimar a banda requerida pela conexão. Para estimar é utilizado um filtro linear de Kalman descrito em [DZI97]. O CAC dinâmico em [SAI91] utiliza a medida de número de células recebidas durante um intervalo de tempo e os descritores de tráfego para estimar o CLR. [BEN97] propôs um algoritmo de lógica *Fuzzy* para prever o CLR. Técnicas de CAC menos conservadoras baseadas em heurística ou em técnicas de medidas em tempo real podem resultar em redução da possibilidade contínua de atingir o QoS para todas as conexões devido à constante necessidade de ajustes na alocação dos recursos.

Os algoritmos do CAC não são especificados pelo ATM Forum ou ITU-T pois a implementação depende da arquitetura de fila e do escalonador empregado pelo fabricante no comutador ATM. Ainda mais, não é necessário ter a mesma implementação de CAC em todos os comutadores da rede para atingir o QoS de ponta a ponta da conexão. A avaliação e alocação de recursos se referem exclusivamente a cada equipamento.

2.5.2 UPC/NPC - Policiamento de Tráfego

A alocação de recursos pelo CAC não é suficiente para garantir o QoS das conexões. Para manter a qualidade de serviço assumida com os contratos de tráfego que foram estabelecidos, a rede ATM necessita monitorar o tráfego dos usuários.

O monitoramento do tráfego transmitido na rede é referenciado como policiamento de tráfego. No policiamento, o comutador ATM verifica o fluxo de entrada de células para certificar-se que o usuário está respeitando o contrato de tráfego estabelecido. Esse policiamento reside no ponto de acesso do usuário à rede, ou seja, na interface UNI. Na interconexão de redes públicas e privadas, a rede privada é considerada um usuário e, portanto, o policiamento é feito na interface de entrada da rede pública (fig. 2.13).

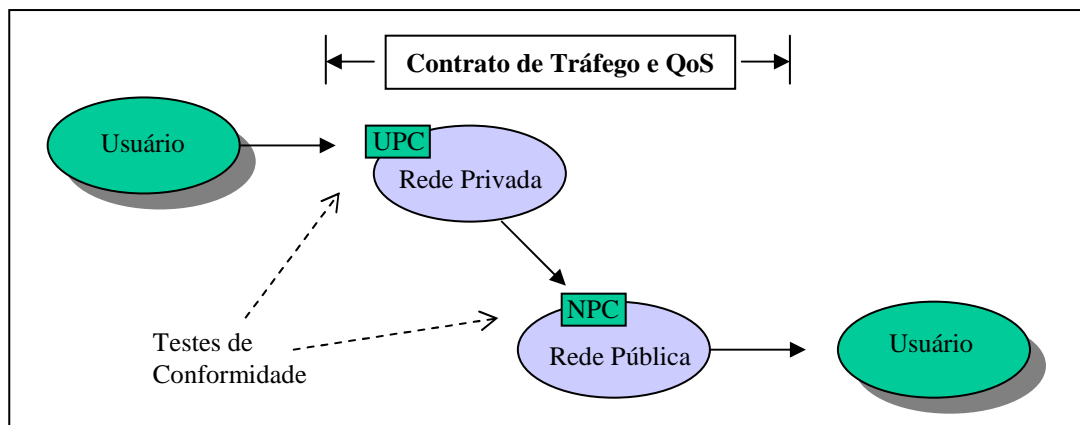


FIGURA 2.13 - Pontos de verificação de conformidade

Algumas razões podem levar o usuário a não cumprir com os parâmetros de tráfego negociados. Podemos citar o desconhecimento do usuário em estimar com precisão a carga requerida. Os recursos de rede estimados pelo CAC são baseados no perfil de tráfego informado pelo usuário. Um excesso no fluxo de células do usuário obriga a rede a dispor recursos que estariam reservados para outras conexões. Sem o devido policiamento, fluxos mal comportados⁵ prejudicariam o comprometimento de desempenho da rede com os usuários.

Usuário com fluxo de tráfego bem comportado é aquele cujas células respeitam os valores de transmissão estabelecidos com a rede. Por isso, o policiamento é feito célula a célula. Para cada célula que chega é verificado se está ou não em conformidade aos valores acordados. Essa verificação é chamada também de monitoramento de conformidade.

Células conformes são aquelas que têm um comportamento de acordo com os descritores de tráfego do contrato feito com a rede. Células não conformes são as células que descumprem o contrato estabelecido com a rede, ou seja, tem um comportamento fora do esperado. Células não conformes perdem a garantia de QoS e podem ser descartadas pela rede.

Quando a rede aceita estabelecer um circuito virtual, está de acordo em transportar todas as células que estão em conformidade com o contrato de tráfego. A verificação da conformidade é feita pelo mecanismo chamado UPC (*Usage Parameter Control*) e envolve vários aspectos de validação do tráfego do circuito virtual. O UPC é baseado em variações de implementações do algoritmo GCRA (*Generic Cell Rate Algorithm*) conhecido pela analogia do balde furado (*leaky bucket*) [ONV95][TM4.0]. O UPC tem as seguintes funções:

- verificar a validade dos valores de VPI e VCI no cabeçalho das células;
- monitorar as células para determinar se estão em conformidade com os descritores de tráfego;
- tomar ações sobre as células não conformes: descartá-las ou marcá-las deixando-as seguir o fluxo (fig. 2.14).

A opção de marcar as células serve para identificá-las como células candidatas (frente às não marcadas) ao descarte caso haja necessidade. O termo marcar refere-se a alterar o bit CLP (*Cell Loss Priority*) do cabeçalho da célula para o valor 1, indicando que essa célula tem baixa prioridade em relação às com CLP=0. É importante notar que o mecanismo UPC não implementa contenção para monitorar o tráfego. Portanto, não causa nenhuma variação de atraso. A única modificação nas características do fluxo é a remoção de células ou alteração da prioridade de descarte da célula (CLP para 1). A função de policiamento deve detectar o mais rápido possível fontes de tráfego não conforme. As devidas ações devem ser tomadas para minimizar o potencial de dano em outras conexões causado pelo excesso de tráfego. Isso deve ser feito de forma transparente para os usuários cujo tráfego está em conformidade.

⁵ Fluxos mal comportados são aqueles que excedem seus parâmetros de tráfegos acordados com a rede no estabelecimento da conexão.

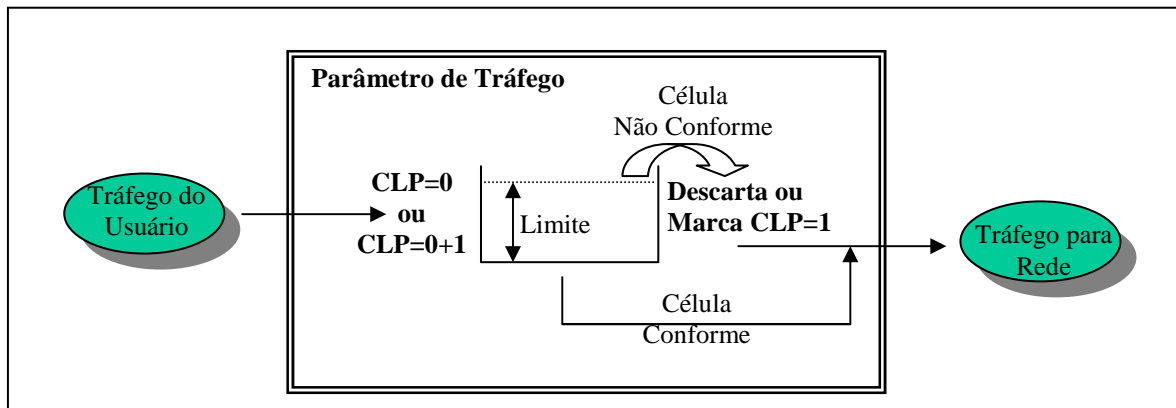


FIGURA 2.14 - Policiamento de Tráfego: analogia do balde furado

2.5.3 Formatação de Tráfego (*Traffic Shaping*)

A quantidade reservada de largura de banda para uma conexão considera suas taxas de utilização média e máxima de bits por segundos. Fontes com característica de rajada, ou seja, com grande diferença entre a média e o pico de transmissão, têm longos períodos de baixa atividade seguidos por transmissões à taxa de pico. Aproveitando o comportamento desse tipo de tráfego é possível reduzir a taxa de pico através do enfileiramento (*bufferização*) das células antes delas entrarem na rede, de forma que a taxa de saída dessa fila seja menor que a taxa de pico. Esse procedimento é chamado de formatação de tráfego (*traffic shaping*) e é implementado em um ponto de acesso à rede. São dois os propósitos do *traffic shaping*: economizar largura de banda e conformar o tráfego para evitar descarte de células. Para ter certeza que as células de uma fonte de tráfego estarão conformes, um usuário pode utilizar um software formatador de tráfego antes que esse dê ingresso na rede ATM.

O *traffic shaping* altera as características do fluxo de células para alcançar melhor eficiência da rede e ainda satisfazer o QoS requerido. Exemplos dessas funções incluem redução de PCR, redução do tamanho de rajada, remoção de CDV e espaçamento entre células. Todos esses mecanismos têm o efeito de alterar a característica do tráfego e introduzir uma quantidade nominal de atraso. Portanto, tráfego sensível ao atraso, como serviço de voz, restringe a capacidade de redução da taxa de pico provida pelo *traffic shaping*.

3 Planejamento de Capacidade de Redes ATM

3.1 Relação Carga, Capacidade e Desempenho

Em resumo, uma rede provê a habilidade de comunicar informações entre usuários, com o objetivo de prover um serviço efetivo a custos razoáveis. Não é eficiente manter recursos permanentemente dedicados aos usuários. Existe a necessidade de compartilhar e dispor aos usuários meios de acessar esses recursos quando requisitados. Uma rede é composta por elementos físicos (*switches*, equipamentos terminais, enlaces de transmissão), cada qual com sua capacidade finita de transmissão de informações. Se a capacidade não é adequada para suportar a carga de serviços demandada pelos usuários, o desempenho da rede em transmitir as informações será prejudicado. Existe uma relação de dependência entre carga, capacidade e desempenho de uma rede (fig. 3.1). A carga é a quantidade de tráfego a ser carregada na rede. A capacidade da rede é dada pelos recursos que a constituem. E o desempenho representa a qualidade de serviço oferecida pela rede aos usuários.

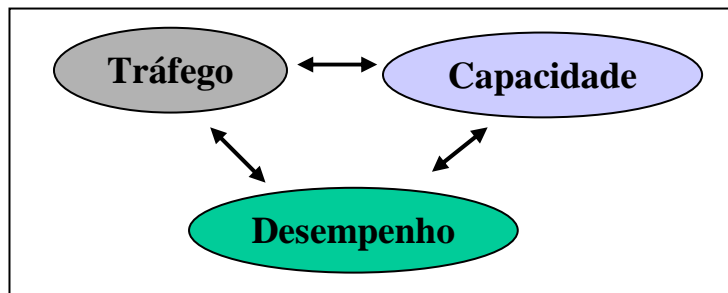


FIGURA 3.1 - Elementos do Relacionamento de Dependência

Alterações em qualquer um desses três elementos influenciam o comportamento dos outros dois. Podemos fixar um elemento em determinada quantidade para avaliar o comportamento dos outros dois elementos entre si. Por exemplo, fixemos a capacidade (quantidade de recursos) de uma rede de comunicação. Aumentando continuamente a quantidade de carga (tráfego) sobre a rede, atingiremos em determinado momento o esgotamento dos recursos ocasionando a redução no desempenho da rede percebido em questões de tempo de resposta, congestionamento e indisponibilidade do serviço (fig. 3.2). Os usuários percebem e avaliam os serviços de rede através dos níveis de serviço. Associado ao nível de serviço está o custo a ser pago pelo usuário. Portanto, de comum acordo com os usuários, a instalação deve definir quais são os níveis de serviço desejáveis e quais são intoleráveis. Exemplos de outros procedimentos que utilizam esse relacionamento são:

- Dimensionamento de recursos: para especificar a capacidade requerida da rede precisamos avaliar a carga demandada e os níveis de desempenho esperados;
- Avaliação de performance: para determinar o desempenho da rede medimos o comportamento da carga sobre determinada capacidade de rede. Analisar o desempenho de um particular projeto de rede tem como

ênfase variar o tráfego e medir o desempenho para dada capacidade de rede;

- Admissão de chamadas: para determinar a carga suportada pela rede avaliamos se a quantidade de recursos disponível permite alcançar o desempenho desejado. O CAC em redes ATM parte da capacidade de recursos e dos requisitos de desempenho para determinar quanto e que combinação de tráfego pode ser aceita pela rede.

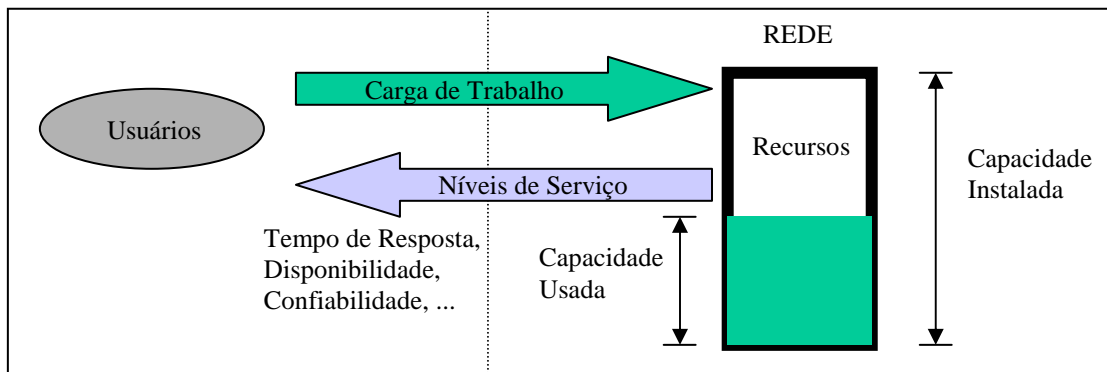


FIGURA 3.2 - Níveis de serviço (QoS), carga e recursos da rede [MEN85]

Os usuários, em geral, percebem os serviços de informática através do tempo de resposta, da disponibilidade do sistema e da facilidade de uso. A última está associada ao software utilizado, enquanto as outras duas estão relacionadas ao dimensionamento de recursos, isto é, ao planejamento de capacidade.

O Planejamento de Capacidade é baseado no relacionamento entre esses três elementos básicos: carga (tráfego), capacidade (recursos) e desempenho (QoS).

3.2 Fases do Planejamento de Capacidade

O sistema de computação atende às solicitações dos usuários com um certo desempenho, que pode ser definido quantitativamente através de várias medidas como, por exemplo, tempo de resposta, taxa de processamento, índice de disponibilidade, e outros. O desempenho de um sistema de computação resulta da interação da carga de trabalho com os recursos que compõem o sistema. A capacidade de um sistema de computação é definida como sendo a carga de trabalho que o sistema pode processar, sem ultrapassar os limites de desempenho estabelecidos pelos níveis de serviço da instalação [STE81]. O planejamento de capacidade é um processo para determinar, no tempo preciso, a quantidade adequada de recursos para atender a carga de trabalho dentro de níveis de serviços propostos (fig. 3.3).

O processo de planejamento de capacidade para configuração de uma rede pode ser decomposto em quatro fases:

- Caracterização da carga de trabalho;
- Definição dos níveis de serviço (QoS);
- Previsão de desempenho sobre os recursos disponíveis;
- Se necessário, adequação dos recursos.

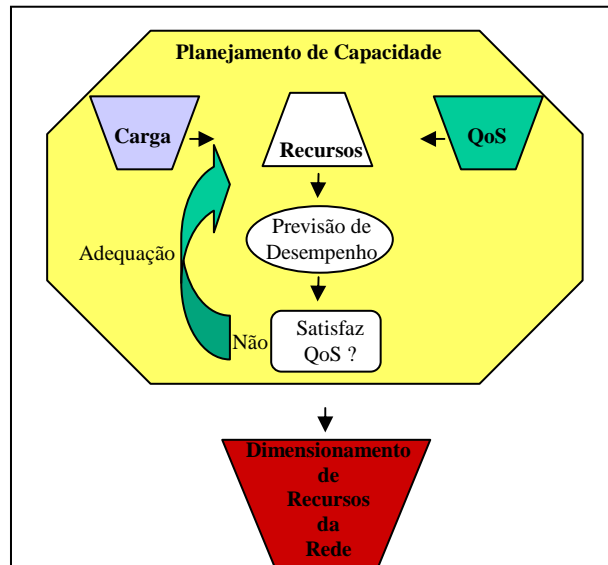


FIGURA 3.3 - Planejamento de Capacidade

Os recursos básicos da rede ATM são a capacidade de *buffers* nas portas, a largura de banda dos enlaces e a capacidade de comutação dos *switches*. Esses recursos devem ser dimensionados conforme a carga de trabalho. A unidade básica de carga de trabalho a ser considerada aqui é a conexão ATM definida em determinada categoria de serviço⁶. As conexões presentes em uma rede são heterogêneas. Essas diferenças se manifestam no consumo de recursos da rede. Inicialmente, é necessário identificar e agrupar as conexões conforme a similaridade de seus parâmetros de tráfego e requisitos de QoS. O objetivo desse agrupamento é obter conjuntos de serviços o mais homogêneos possível. Dessa forma podemos realizar procedimentos de alocação de recursos e avaliação de desempenho por grupo de conexões.

Tratando-se de redes ATM, a definição dos níveis de serviço para cada conexão é dada pelos parâmetros de QoS, maxCTD, ppCDV e CLR, conforme valores requisitados pelo usuário. São valores esperados de desempenho da rede que devem ser atingidos para cada serviço. Lembrando que das categorias do ATM Forum, a única que não faz requisito algum de QoS é a UBR.

O processo de dimensionar os recursos de uma rede consiste em utilizar técnicas que permitam prever o desempenho desse sistema frente a novas situações de carga e serviços. Uma vez de posse das estimativas de cargas e dos níveis de serviços desejáveis, deve-se estimar quando as demandas por recursos excederão a capacidade instalada da rede. O ponto de exaustão do sistema ocorrerá quando, em função das demandas previstas, o nível de serviço for considerado insatisfatório (fig. 3.4). A fase de previsão de desempenho deve relacionar o impacto da carga prevista sobre os níveis de serviços e a utilização corrente dos recursos. A questão final do planejamento de capacidade é estimar o ponto em que a demanda de recursos excederá a capacidade instalada de recursos. O processo requer uma ferramenta para a previsão de desempenho. A avaliação de um sistema pode ser a combinação de diversas medidas de desempenho alcançadas durante a realização de suas atividades. Métodos para a

⁶ Categorias conforme ATM Forum: CBR, rt-VBR, nrt-VBR, ABR e UBR.

avaliação de desempenho estão divididos em duas categorias: técnicas de medição e técnicas de previsão.

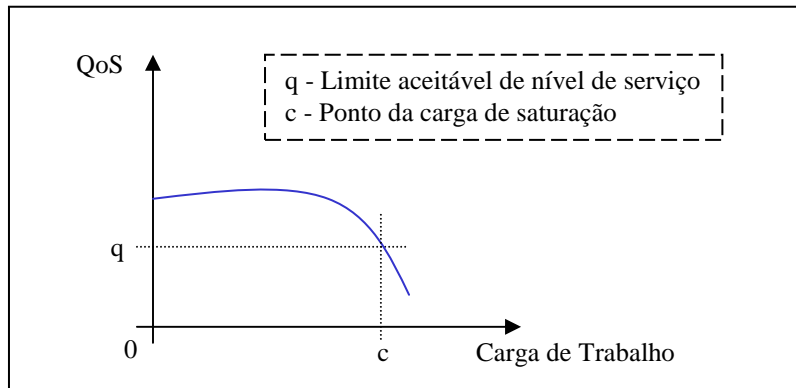


FIGURA 3.4 - Comprometimento dos níveis de serviço devido à sobrecarga

As técnicas de medição são utilizadas quando é possível fazer medições sobre a rede nas condições de tráfego que se deseja. A técnica de medição requer que redes reais sejam avaliadas por experimentação. A vantagem da medição direta da rede é que nenhum detalhe da rede é excluído: a real utilização da rede está sendo monitorada e medida. Existem inconvenientes no caso de redes em produção, quando não é possível interromper os serviços para fazer experimentos de carga. Portanto, a rede não pode ser exercitada no seu limite de carga sem causar queda de desempenho para os usuários que a estão utilizando. Nesse caso, o tipo de tráfego usual disponível restringe as condições de experimento da rede. Em situações de projeto de rede, quando se deseja avaliar o desempenho da futura rede ou em situações futuras de tráfego, as técnicas de medição real não são possíveis porque a rede não está implementada ou porque a carga ainda não existe para ser avaliada. Utilizam-se, então, técnicas de previsão através de modelos analíticos ou simulados. Comparando modelo analítico e simulação, os principais fatores a considerar são a precisão dos resultados, o tempo para obtê-los e o custo de utilizar o método (fig.3.5). A vantagem das soluções analíticas é que elas podem ser usadas razoavelmente de forma rápida. Entretanto, a necessidade de resolver o modelo restringe a abrangência do sistema ou das características de tráfego que podem ser incluídas. Portanto a solução analítica é utilizada com frequência para produzir uma aproximação de um sistema, com resultados sendo produzidos relativamente de forma rápida e com custo acessível.

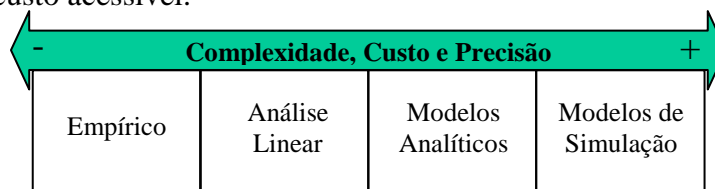


FIGURA 3.5 - Técnicas de Previsão de Desempenho

Redes de maior complexidade podem ser investigadas usando simulação. Sistemas podem ser modelados no nível de precisão desejado permitindo analisar as alternativas de configuração, levando-se em conta aspectos como custos, considerações de confiabilidade, carga projetada e níveis de serviço. Muitas vezes a simulação é o método mais adequado devido à natureza do problema e porque técnicas analíticas tornam-se muito difíceis de serem tratadas. Entretanto, a simulação pode ter um alto custo de desenvolvimento e execução. Simulações de eventos raros, como perda de células ATM, consomem um tempo considerável de execução do simulador.

3.3 Método para Dimensionamento de Recursos na Rede ATM

Os principais recursos de uma rede ATM são a capacidade das filas de células e a largura de banda dos enlaces. As células são armazenadas em filas enquanto esperam para serem transmitidas na largura de banda dos enlaces. As filas são conhecidas por *buffers* e a capacidade é dada em quantidade de células. A capacidade da largura de banda dos enlaces é dada geralmente em Mbit/s podendo ser transformada em células por segundo (cel/s). A alocação de quantidade de *buffers* e de largura de banda é determinada pela combinação dos parâmetros de tráfego e dos requisitos de QoS das conexões que compartilham esses recursos (fig. 3.6).

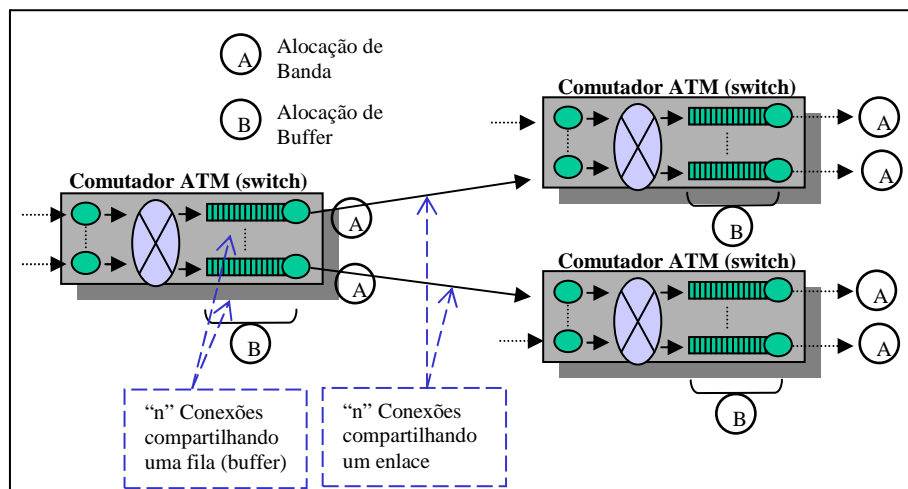


FIGURA 3.6 - Pontos de Alocação de Recursos

Como foi visto, o planejamento de capacidade é um processo que tem como objetivo prever a quantidade necessária de recursos para suportar dada quantidade de tráfego. Portanto, o processo de planejamento de capacidade é importante para o projeto e dimensionamento de novas redes. Da mesma maneira, no caso de redes já instaladas, é possível, através do planejamento de capacidade, responder se a rede irá suportar um aumento na carga de tráfego. Para realizar o planejamento de capacidade de uma rede ATM, propomos um método com os seguintes procedimentos:

- 1) Determinar a quantidade de recursos disponíveis na rede:
 - ❑ Montar uma tabela por porta dos comutadores com capacidade de *buffer* e largura de banda conforme informações do fornecedor do equipamento;
 - ❑ Denominaremos essa tabela de "Recursos Disponíveis x Alocados", conforme modelo (tab. 3.1);
 - ❑ No caso de projeto de rede, a tabela de recursos disponíveis estará em branco, visto que a configuração dos equipamentos será de acordo com o resultado do método de planejamento de capacidade.
- 2) Caracterizar o tráfego da rede:
 - ❑ Os recursos demandados são resultado das características de tráfego e requisitos de QoS das conexões;
 - ❑ Portanto, é necessário descrever cada conexão presente na rede em termos de seus descritores de tráfego e requisitos de QoS. Usaremos, para organizar essas informações, a tabela denominada "Caracterização

de Serviços" (tab. 3.2). Os campos da tabela são preenchidos com informações provenientes dos implementadores dos serviços, como desenvolvedores de aplicações, administradores de rede, e outros;

- ❑ Dessa forma, todas as aplicações conhecidas que a rede deverá suportar estarão devidamente caracterizadas por categoria de serviço.
- 3) Determinar a topologia de conexões:
- ❑ A alocação de recursos é feita por porta conforme a carga de conexões que a utiliza;
 - ❑ Portanto, é necessário esquematizar a topologia física de conexões, identificando as portas de comutadores por onde passa cada conexão. Utilizamos a tabela denominada "Caminho de Conexões" (tab. 3.3) para organizar essas informações;
 - ❑ Caminhos alternativos devem ser considerados também. O ATM trabalha com protocolos de roteamento de conexões (PNNI) que permitem o balanceamento de carga entre caminhos redundantes. Portanto, se os recursos estão escassos em um caminho, deve-se avaliar também os caminhos alternativos.
- 4) Determinar a quantidade de recursos demandados para cada conexão ou grupo de conexões que compõem o tráfego:
- ❑ Cada categoria de serviço possui um método específico para determinar a quantidade de recursos necessários para satisfazer o QoS da conexão. Os métodos para cada categoria de serviço serão detalhados nas próximas seções deste capítulo;
 - ❑ Como já foi visto, a alocação de recursos é feita por porta conforme a carga de conexões que a utiliza;
 - ❑ Portanto, seguindo as duas premissas anteriores, devemos agrupar as conexões por categoria de serviço do ATM Forum para cada porta de comutador. A partir das informações de caminho da tabela 3.3, podemos organizar a quantidade e tipo de conexões por porta. Assim, teremos para cada porta de comutador uma tabela com, no máximo, 5 linhas, denominada "Conexões por Porta" (tab. 3.4). Cada categoria de serviço ocupa uma linha;
 - ❑ A partir das informações da tabela "Conexões por Porta", é possível determinar o perfil de tráfego presente na porta para cada categoria de serviço. Os métodos de alocação de recursos utilizam essas informações para dimensionar a quantidade de recursos necessária para cada grupo de categoria de serviço.
- 5) Alocar os recursos demandados nos locais onde são necessários, diminuindo-os dos recursos disponíveis:
- ❑ O total de recursos demandados por porta é a soma de todos os recursos necessários para cada categoria de serviço da tabela "Conexões por Porta";
 - ❑ O total de recursos necessários para cada porta de comutador deve ser transportado para a tabela "Recursos Disponíveis x Alocados" (tab. 3.1), caracterizando a quantidade de recursos alocados;
 - ❑ Para cada porta dos comutadores na tabela de "Recursos Disponíveis x Alocados", os recursos alocados devem ser menores ou iguais aos

recursos disponíveis. Para tanto, o resultado da subtração de recursos disponíveis menos os recursos alocados, para cada porta, é colocado na coluna de contabilização;

- E, por fim, é preciso verificar se os recursos demandados foram plenamente satisfeitos. Se em algum ponto a quantidade de recursos for insuficiente (campo contabilização com valor negativo), a reconfiguração da carga de tráfego ou dos recursos disponíveis nessa porta é necessária.

A figura 3.7 mostra a topologia de conexão utilizada para exemplificar a utilização das tabelas dos procedimentos de planejamento de capacidade relacionados anteriormente.

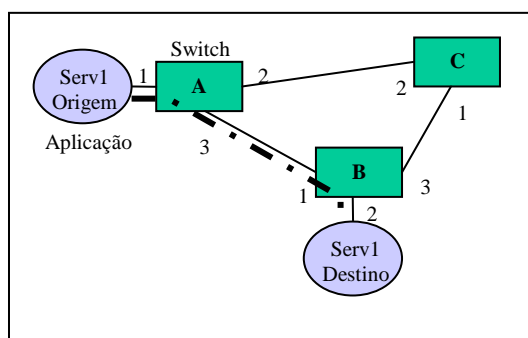


FIGURA 3.7 - Exemplo de Topologia para uma Conexão

TABELA 3.1 - Modelo de tabela: Recursos Disponíveis x Alocados

Comutador	Porta	Recursos Disponíveis		Recursos Alocados		Contabilização	
		Buffer (células)	Banda (Mbit/s)	Buffer (células)	Banda (Mbit/s)	Buffer (células)	Banda (Mbit/s)
Switch A	A1	X	Y	Z	K	X - Z	Y - K
	A2
	A3	10000	149 ⁷	200	7	800	142
Switch B	B1
Switch C	C1
.....

TABELA 3.2 - Caracterização de Serviços

Id. Serviço ⁸	Aplicação	Categoria de Serviço	Descritores de Tráfego ⁹					Requisitos QoS ¹⁰		
			PCR	SCR	MBS	MCR	CDVT	CLR	CTD	CDV
Serv.1	Emulação de Circuito	CBR	1							
Serv.2	Emulação de Circuito	CBR	2							
Serv.3	Vídeo Conferência	VBR-rt	5	1	300					
Serv.4	Vídeo Sob Demanda	VBR-nrt	1,5	1	1000				5	6,5
Serv.5	Conexão de LAN	UBR	-	-	-	-	-	-	-	-
....

⁷ Enlace STM-1 (155 Mbit/s) descontado *overhead* do SDH.

⁸ Serviços com transmissões assimétricas (parâmetros de tráfegos ou requisitos de QoS diferentes para cada sentido da conexão) devem ser especificados separadamente. Normalmente a alocação de recursos é idêntica nos dois sentidos da conexão.

⁹ Valores genéricos para o exemplo. Informações de PCR, SCR e MCR dadas em Mbit/s.

¹⁰ Requisitos de CTD e CDV conforme [RUS93].

TABELA 3.3 - Caminho das Conexões

Id. Conexões	Id. Serviço	Cat. Serviço	Caminho ¹¹	Caminhos Alternativos
Cx.1	Serv.1	CBR	A1,A3,B1,B2	A1,A2,C2,C1,B3,B2
Cx.2	Serv.1	CBR
Cx.3	Serv.2	CBR	..A3...
Cx.4	Serv.3	VBR-nrt	..A3..
....

TABELA 3.4 - Modelo: Tabela Conexões por Porta

PORTA: A3			
Categoria de serviço	Id. Conexões	Recursos Necessários	
		Buffers	Banda
CBR	Cx.1, Cx.3	50	3
rt-VBR	Cx.4	150	4
nrt-VBR		
ABR		
UBR		
Total Alocado :		200	7

3.4 Modos de Alocação de Recursos

O controlador de admissões de chamadas (CAC) do ATM possui funções características do processo de planejamento de capacidade. A cada solicitação de conexão feita à rede, o CAC avalia a quantidade de recursos necessários para suportar a nova demanda nos níveis de QoS impostos. Uma vez que os recursos requeridos pela conexão são avaliados, a função do CAC é verificar se os recursos disponíveis no caminho considerado possuem valor igual ou maior aos recursos demandados. Se não houver recursos suficientes, a conexão é rejeitada. O CAC utiliza um algoritmo de alocação lógica de recursos por conexão. Entenda-se a alocação lógica como associar certa quantidade de banda para uma conexão, porém não há em ATM nenhum mecanismo que possa garantir essa alocação de forma física. Enquanto o princípio de compartilhamento de recursos tem a vantagem da multiplexação estatística, o conceito de alocação lógica de banda foi criado para tirar proveito das simples e bem conhecidas técnicas desenvolvidas para as redes de circuitos comutados, tais como CAC e mecanismos de roteamento.

A alocação lógica de banda para uma conexão pode ser avaliada em função dos parâmetros de tráfego da fonte, da capacidade do enlace (largura de banda), do tamanho do *buffer*, dos requisitos de QoS e da combinação (multiplexação) de tráfego sobre o enlace.

O tamanho das filas tem um impacto significativo na capacidade dos comutadores e na qualidade dos serviços. Para conexões CBR, o tamanho das filas deve ser pequeno e o enfileiramento, em geral, deve ser evitado. Mantendo pequeno o tamanho da fila, a quantidade total de tempo despendido pela célula aguardando ser

¹¹ Caminho corresponde ao conjunto de portas por onde passa o tráfego nos dois sentidos da conexão (ida e volta). A ordem das portas não é relevante. O importante é mapear a quantidade de tráfego em cada porta.

transmitida é reduzido. Isto tem um impacto positivo em reduzir o CTD e CDV. Aplicações usando conexões CBR e VBR-rt, como voz e vídeo conferência, podem preferir descartar uma célula a permitir que a célula fique um longo período de tempo na fila.

A alocação lógica de banda pode ser feita de diferentes modos. Entretanto, em todos os casos o objetivo comum é que mesmo quando toda a capacidade do enlace for alocada a qualidade de serviço comprometida não pode ser violada para nenhuma conexão. Os modos de alocação lógica de banda são:

- ❑ pela Taxa de Pico (PCR);
- ❑ pela Taxa Mínima (MCR);
- ❑ pela Banda Equivalente.

A seguir são apresentadas as vantagens e desvantagens desses modos de alocação lógica.

3.4.1 Alocação de Banda pela Taxa de Pico

O esquema mais simples de alocação lógica de banda é utilizar a taxa de pico da conexão como a quantidade de banda a ser reservada no ponto de fila. As vantagens dessa opção são:

- ❑ o valor para alocação de banda é diretamente pego de um dos descritores de tráfego da conexão (PCR);
- ❑ o mecanismo de policiamento da conexão é simples de ser implementado;
- ❑ a alocação não depende da característica de tráfego das outras conexões.

Portanto, a implementação dessa opção é muito simples, tanto do ponto de vista do usuário como da rede. A alocação de *buffer* também é simples. Para garantir taxas baixas de CLR, o *buffer* necessita ter o tamanho igual ao número de conexões que o compartilham. A alocação de *buffer* é feita de forma a acomodar a chegada simultânea de células de diferentes conexões (fig. 3.8). A qualidade de serviço em termos de CTD e CDV também é garantida para todas essas conexões visto que o *buffer* é de tamanho pequeno.

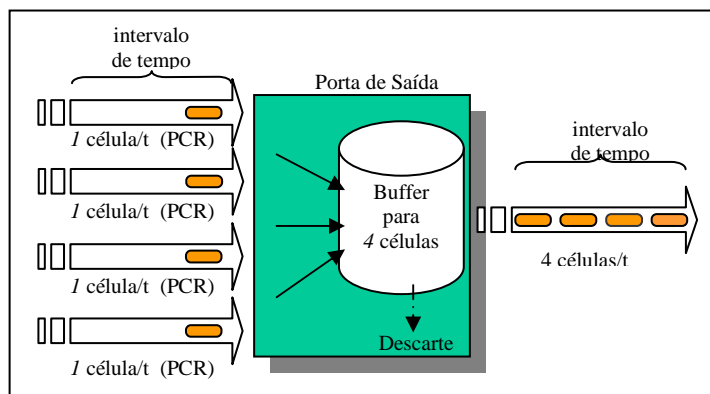


FIGURA 3.8 - Chegada simultânea de células

A desvantagem óbvia desse modo de alocação é que os recursos da rede podem ser extremamente subutilizados quando a fonte das conexões tem características de

tráfego com taxa de transmissão variável (fig. 3.9). A banda disponível das conexões que não estão transmitindo à taxa PCR poderia ser aproveitada por outras conexões. A alocação pela taxa de pico não tira proveito da capacidade de multiplexação estatística do ATM.

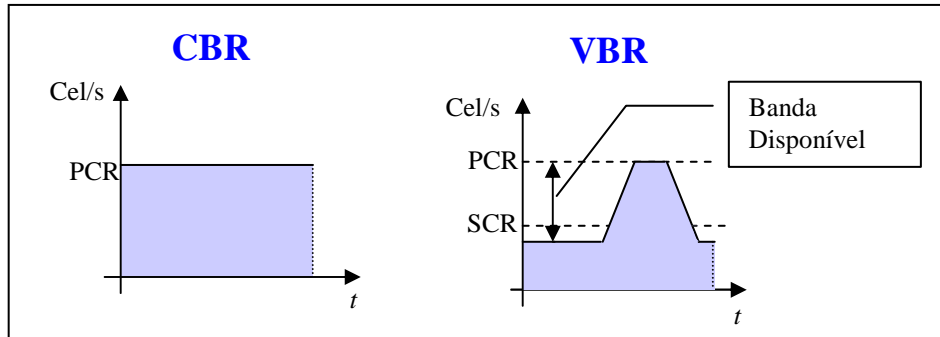


FIGURA 3.9 - Alocação pela Taxa de Pico para CBR e VBR

3.4.2 Alocação de Banda pela Taxa Mínima

A alocação pela taxa mínima é adequada para serviços de dados com fluxos controlados. Nesse caso, a rede aloca para a conexão a banda correspondente à taxa mínima de transmissão de células. O tráfego excedente ainda pode ser carregado pela rede desde que haja banda disponível no momento. A qualidade de serviço é garantida para as células que são transmitidas até a taxa mínima. Portanto, o tráfego excedente deve ser controlado pela fonte com algum mecanismo de informação baseado no estado de congestionamento da rede. A vantagem desse esquema de alocação é seu potencial para a disponibilização dinâmica dos recursos da rede, tornando-os menos ociosos. A desvantagem é que o tráfego excedente ao mínimo alocado não tem garantia de QoS.

3.4.3 Alocação pela Banda Equivalente

Visto que a taxa máxima de transmissão de uma conexão é o PCR, ignorando o CDV adquirido, a rede não necessita alocar mais do que o PCR para satisfazer o QoS da conexão. Entretanto, desde que a conexão não envie células continuamente, é possível alocar uma banda menor quando diversas conexões são multiplexadas num ponto de fila. Esse valor resultante da multiplexação estatística que satisfaz o QoS é chamado de *banda equivalente*. A banda equivalente se encontra entre a taxa de pico e a taxa média de transmissão da conexão (fig. 3.10).

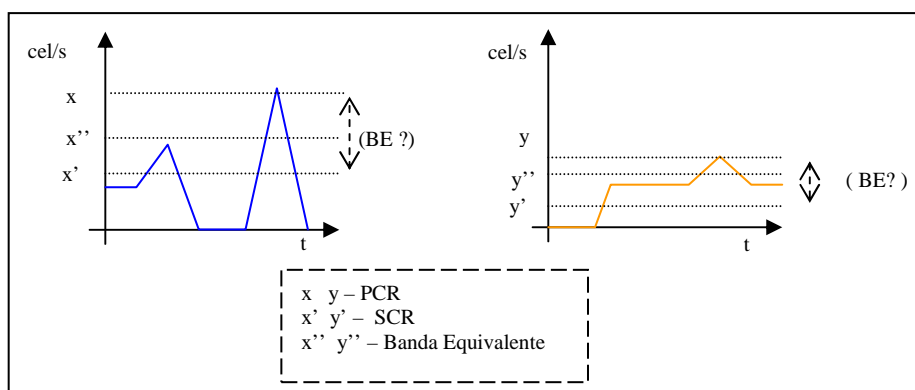


FIGURA 3.10 - BE: Banda Equivalente, $SCR \leq BE \leq PCR$

A soma das taxas de picos de todas as conexões é maior que a capacidade do enlace, enquanto que a soma de todas as bandas equivalentes é menor ou igual à capacidade do enlace. A eficiência na utilização da banda do enlace aumenta quando a banda equivalente das conexões se aproxima de suas taxas médias, e piora quando se aproxima dos picos das conexões. Em geral, a alocação pela banda equivalente permite um número maior de conexões do que a alocação pelo pico e, portanto, um melhor aproveitamento dos recursos da rede. O ganho de multiplexação estatística (SGM - *Statistical Multiplexing Gain*) no enlace pode ser definido pela seguinte expressão:

$$SGM = \frac{\sum PCR}{\sum BE}$$

Onde,

ΣPCR é a soma das taxas de pico de todas as conexões no enlace;
 ΣBE é a soma das bandas equivalentes de todas as conexões no enlace.

O ganho de multiplexação estatística é determinado por características das conexões. Geralmente quando o número de conexões aumenta sobre um enlace, o fluxo agregado resultante tende a ser mais constante (com menor explosividade de rajadas). Isso porque os fluxos de rajada das conexões individuais estatisticamente tendem a se intercalar. Nesse caso, a banda equivalente do fluxo agregado se aproxima da taxa média de utilização do enlace, aumentando o ganho da multiplexação estatística, ou seja, a eficiência na utilização do enlace. Esse efeito não é alcançado quando se multiplexam conexões com pequenas rajadas, ou seja a taxa de pico é muito próxima da taxa média de transmissão. Há pouca área para o aproveitamento da multiplexação estatística e a banda equivalente se aproxima da taxa de pico. O mesmo acontece com a multiplexação de fontes de alta taxa de transmissão com fontes de baixa taxa. O efeito de compensação das fontes de baixa sobre as de alta para tornar a banda equivalente próxima da média de transmissão pode ser insignificante. O ganho é geralmente uma função do tamanho da fila, das características do tráfego e os objetivos de QoS das conexões que estão sendo multiplexadas. A banda equivalente também é denominada de banda efetiva ou capacidade equivalente.

A alocação pela banda equivalente é um dos métodos propostos na literatura para o CAC. Ela consiste em determinar, através do estudo preciso da multiplexação estatística das fontes, a capacidade efetiva (capacidade estatística) requerida para cada uma delas. Na prática, uma nova conexão só é aceita se a rede dispuser de recursos para suportar a QoS requerida em cada porta de comutador (enlace) desde a origem até o destino por onde passa a conexão.

O dimensionamento da banda equivalente é feito através de métodos analíticos ou simulação. As simulações normalmente levam um tempo proibitivo para determinar a banda equivalente em relação ao tempo que o CAC, em um sistema real, dispõe para decidir sobre a aceitação da chamada. No entanto, as simulações são importantes como referência para comparar as diversas propostas de métodos para o CAC [MON94] e [PER96].

Os métodos propostos focam seus estudos, normalmente, no multiplexador estatístico existente em cada porta de saída do comutador ATM (fig. 3.11). Podemos simplificar o modelo desse multiplexador como sendo um *buffer* de comprimento K , dado em números de células, sendo alimentado por N fontes de tráfego distintas com a mesma taxa média B_m (bit/s) e mesma taxa de pico B_p (bit/s), compartilhando um canal de capacidade W (bit/s).

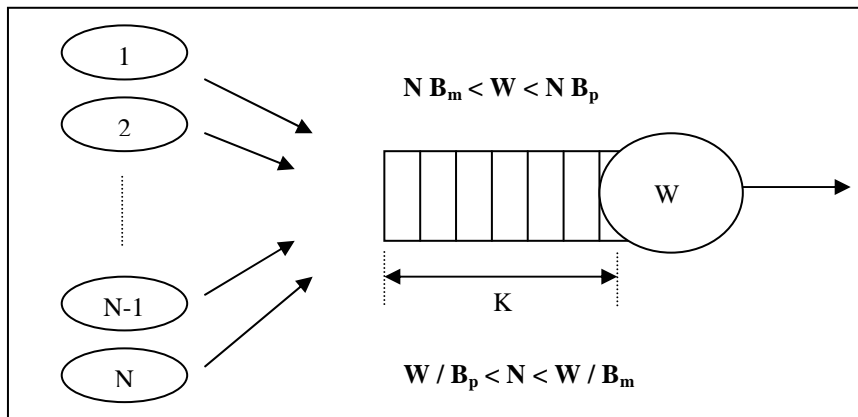


FIGURA 3.11 - Multiplexador Estatístico

Nesse modelo podemos ter os seguintes problemas para resolver:

- Problema de alocação de capacidade (W);
- Problema do dimensionamento de memória (K);
- Problema da determinação do número de conexões máximas (N).

O problema de alocação de capacidade consiste de: dada uma mistura de N fontes distintas que alimentam um multiplexador com *buffer* de tamanho K , deseja-se encontrar a capacidade W do canal que deve ser alocada a essa mistura de tráfego de modo a satisfazer determinada QoS (taxa de perda de células, atraso máximo e variação do atraso). No problema do dimensionamento de memória (*buffer*), estamos interessados em determinar o valor de K que satisfaz uma dada QoS, dada uma mistura de N fontes distintas num canal de capacidade W . O problema da determinação do número de máximo de conexões consiste em determinar o número máximo N de fontes distintas que podem compartilhar o multiplexador com capacidade W e K *buffer* e ainda satisfazer determinada QoS.

3.5 Métodos de Alocação de Recurso por Categorias de Serviços

Os modos de alocação pela taxa de pico, pela taxa mínima ou pela banda equivalente, como visto, são utilizados conforme as vantagens que proporcionam para determinado tipo de tráfego. Em ATM os tipos de tráfego são agrupados por similaridade em categorias de serviços.

As alocações de recursos das conexões conforme a categoria são apresentadas a seguir:

- Conexões CBR são baseadas no modo de alocação pela taxa de pico;

- Conexões VBR são baseadas no modo de alocação pela banda equivalente;
- Conexões ABR são baseadas no modo de alocação pela taxa mínima.

Devido a sua característica de não ter requisitos de QoS, conexões da categoria UBR não alocam recursos na rede. Pela natureza dessas conexões, elas são transportadas quando há disponibilidade de recurso na rede.

Portanto, conforme as características de tráfego de cada categoria de serviço e o método de alocação apropriado, podemos avaliar a quantidade de recursos necessários em cada porta de comutador da rede ATM.

3.5.1 Alocação de Recursos para Tráfego CBR

A fonte de uma conexão CBR envia uma célula periodicamente a cada $1/PCR$ intervalo de tempo. A alocação de recursos para tráfego CBR é baseada na taxa de pico (PCR) de cada conexão. Portanto, a banda necessária em uma porta de comutador para um grupo de conexões CBR é dado por:

$$BA_{CBR} = \sum_{i=1}^n PCR_i$$

Onde,

BA_{CBR} é a banda alocada para as conexões CBR;

PCR_i é a taxa de pico da conexão "i".

n é o número de conexões do tipo CBR nessa porta;

Quando as células atravessam diversos pontos de fila, a periodicidade de um tráfego CBR será perdida. As células podem se aproximar ou se dispersar devido à ação de outro tráfego sobre a mesma fila. Quando o tempo entre células diminui (quando se aproximam uma da outra) a taxa efetiva de transmissão aumenta em relação ao PCR declarado da fonte (fig. 3.12). Dessa forma, alocar o PCR declarado pode não ser suficiente para garantir o QoS da conexão. Essas variações não podem ultrapassar o máximo CDV (*Cell Delay Variation*) comprometido pela rede. O CDV pode ser inserido em maior grau quando são multiplexados na mesma fila conexões de tráfego VBR-rt. Mesmo sem essa variação, a chegada simultânea de células (congestionamento em escala de célula¹²) devido à multiplexação de conexões pode causar a perda de células em filas pequenas.

Para resolver a questão de alocação de banda para conexões CBR, levando em conta CDV e a chegada simultânea de células, devemos calcular o tamanho da fila (*buffer*) compartilhada pelas conexões.

¹² Congestionamento em escala de célula ocorre em buffers pequenos devido à chegada simultânea de células de diferentes conexões. O outro tipo de congestionamento é o congestionamento em escala de rajada que ocorre tipicamente em buffers grandes devido à chegada de rajadas de células de diferentes conexões.

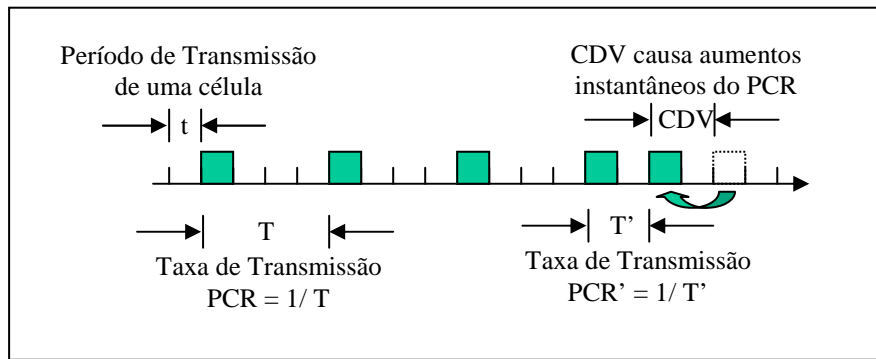


FIGURA 3.12 - Variação na taxa de pico de uma conexão CBR

Cada conexão é policiada na entrada da rede usando o algoritmo GCRA (*Generic Cell Rate Algorithms*) que utiliza como parâmetros, no caso de tráfego CBR, PCR e CDVT. O CDVT serve como limite aceitável de CDV pela rede. Qualquer valor superior a esse será punido pelo policiamento (UPC) e não receberá garantia de qualidade de serviço. Portanto, o tamanho da maior rajada possível é dado pela seguinte equação:

$$MBS = 1 + \left\lceil \frac{CDVT}{\frac{1}{PCR} - \frac{1}{TE}} \right\rceil$$

Onde,

MBS é o tamanho da rajada para o pior caso de variação na taxa de transmissão. É dado em quantidade de células;

$CDVT$ é o parâmetro informado pela fonte com sendo a tolerância máxima para a variação do atraso. É dado em microssegundos;

PCR é a taxa máxima de transmissão da conexão em células por segundo;

TE é a taxa de transmissão do enlace que atende a porta do comutador em questão. É dado em células por segundo.

Portanto, considerando o CDV e o congestionamento em escala de célula, para o cálculo do recurso tamanho de *buffer* é necessário satisfazer as seguintes expressões:

$$BU_{CBR} \geq n + \sum_{i=1}^n MBS_i$$

e

$$BU_{CBR} \leq \max CTDq * TE$$

Onde,

MBS_i é o tamanho máximo de rajada da conexão "i".

"n" é a quantidade de conexões CBR nessa porta.

$\max CTDq$ é o valor máximo de atraso permitido para essa fila em questão. Transformado em unidade de segundos para compatibilizar com o valor de TE .

TE é a taxa de transmissão do enlace, dado em células por segundo.

BU_{CBR} é o tamanho da fila necessária para satisfazer QoS, ou seja, razão de perda de células (CLR) próximo a zero e valor máximo de CTD.

O $maxCTDq$ é a parcela de atraso de fila dessa porta no $maxCTD$ de ponta a ponta da conexão. O atraso na entrega da célula é acumulativo. O $maxCTDq$ de todas as portas mais os componentes fixos de atraso (propagação, transmissão e comutação) não podem ultrapassar o valor de $maxCTD$ requisitado pela conexão. Em nosso método para dimensionamento de recursos utilizamos valores máximos de CTD por porta baseados nos requisitos genéricos definidos em [BEL95] e [BEL96]. A tabela completa dos requisitos genéricos de desempenho está no anexo 1.

Através das equações de cálculo de BA_{CBR} e BU_{CBR} apresentadas acima, é feita a alocação dos recursos para a conexão CBR. Estimar o pior caso de rajadas possível garante que não ocorrerá perda de células. Entretanto, essa premissa de alocação pode ser considerada muito pessimista devido à possibilidade de subutilização dos recursos. Outro modo é considerar a conexão CBR como fosse VBR e tratá-la com as técnicas de alocação para conexões VBR. Nesse caso, os parâmetros VBR de determinada conexão CBR seriam derivados da seguinte forma:

$$\begin{aligned} SCR' &= PCR \\ PCR' &= TE \\ MBS' &= 1 + \left[\frac{CDVT}{\frac{1}{PCR} - \frac{1}{TE}} \right] \end{aligned}$$

Onde,

SCR' é a taxa média de transmissão de uma conexão VBR.

PCR' é a taxa de pico de transmissão de uma conexão VBR. É dado em células por segundo;

PCR é a taxa de pico da conexão CBR. É dado em células por segundo;

MBS é o tamanho da rajada para o pior caso de variação na taxa de transmissão. É dado em células;

TE é a taxa de transmissão do enlace, dado em células por segundo.

Outros métodos de alocação de recursos utilizados para conexões CBR simplesmente consideram desprezível o valor de CDV e se preocupam apenas com a questão de congestionamento em escala de célula (chegada simultânea de células). Esses métodos são apresentados em [RMV96].

3.5.2 Alocação de Recursos para Tráfego ABR

O tráfego para conexões de categoria ABR tem a garantia de uma taxa mínima de banda dada pelo parâmetro de tráfego MCR. Portanto, a alocação de recursos para tráfego ABR é baseada na taxa mínima.

A banda necessária alocada em uma porta de comutador para um grupo de conexões ABR é dado por:

$$BA_{ABR} = \sum_{i=1}^n MCR_i$$

Onde,

BA_{ABR} é a banda alocada para o conjunto das conexões ABR;
 MCR_i é a taxa mínima de transmissão da conexão "i";
 n é o número de conexões do tipo ABR nessa porta.

A categoria ABR não oferece garantia de CLR, no entanto a taxa de perda de células deve ser mantida baixa quando a conexão transmite dentro de valores do MCR. Da mesma forma, tráfego ABR não possui restrições quanto ao CTD e CDV, o que possibilita ter *buffers* grandes para acomodar a chegada simultânea de células e rajadas de células causadas pela multiplexação de conexões. O dimensionamento da capacidade de *buffer* para tráfegos de dado (categoria de serviço ABR e UBR) tem como objetivo restringir a probabilidade de perda de células dentro de valores desejáveis pela rede. A seguinte expressão é uma aproximação para o dimensionamento de *buffer* em comutadores servindo fluxos de tráfego Markoviano (modelado como M/M/1/B) em função da carga do enlace e da desejada probabilidade de perda [MCD00]:

$$BU_{ABR} \cong \frac{-\ln(\varepsilon)}{1 - \rho} \quad \rho = \frac{\sum_{i=1}^n MCR_i}{BD}$$

Onde,

BU_{ABR} é o tamanho da fila necessária para o conjunto das conexões ABR;
 ε é o valor de probabilidade de perda que se deseja alcançar. Exemplo: 10^{-4} ;
 ρ é o nível de carga de utilização do enlace pelas conexões ABR. Varia de 0 a 1, sendo 1 correspondente a 100% de utilização da banda disponível para as conexões ABR;
 MCR_i é a taxa mínima de transmissão da conexão "i";
 n é o número de conexões do tipo ABR nessa porta;
 BD é a banda disponível para as conexões ABR. É dada como a diferença entre a capacidade do enlace e o total de banda alocada pelas conexões CBR e VBR.

Dessa forma podemos alocar o *buffer* para garantir a probabilidade de perda ε para as conexões ABR na taxa mínima de transmissão.

3.5.3 Alocação de Recursos para Tráfego UBR

O tráfego para conexões de categoria UBR não tem garantia de qualidade de serviço e é tratado com a menor prioridade entre todas as categorias na utilização dos recursos da rede. Por essa razão são conhecidos por serviços de melhor esforço (*best-effort*). Nenhum tipo de alocação de recursos é considerada para garantir qualquer que seja o parâmetro de QoS. Nesse caso, as aplicações que utilizam essas conexões devem

estar preparadas para tratar atrasos e perda de células. A falta de garantia de QoS permite um alto grau de multiplexação entre as fontes, visto que não há nenhum tipo de limitação de ingresso de conexões desse tipo. Aplicações adequadas para o serviço UBR incluem aquelas tolerantes ao atraso ou que não requerem respostas em tempo real. O UBR é próprio para a transmissão de dados caracterizados por uma fonte enviando rajadas não contínuas de células. Embora não tenham restrições quanto ao atraso, o desempenho das aplicações é sensível à perda de células. O uso de *buffers* de grande capacidade minimiza a razão de perda de células (CLR) às custas do aumento do atraso. No caso dessas aplicações, conter o atraso não é requisito almejado.

A utilização de *buffers* para UBR é feita dinamicamente sobre os recursos disponíveis após a alocação de *buffers* para as outras categorias de serviço (CBR, VBR-rt, VBR-nrt e ABR).

3.5.4 Alocação de Recursos para Tráfego VBR

Redes ATM de alta velocidade são dimensionadas para suportar uma grande quantidade de conexões com as mais diversas características. O controle de admissão de chamadas tenta impedir o congestionamento da rede e, por conseqüência, a perda de QoS, limitando o número de conexões que serão admitidas na rede. Muitos esquemas de admissão de chamadas são baseados no conceito de Banda Efetiva ou Capacidade Equivalente. A Banda Efetiva de uma fonte de tráfego é definida como a largura de banda requerida para garantir determinada qualidade de serviço (QoS). Em tais esquemas, a chamada é admitida pelo comutador se a banda efetiva da fonte é menor ou igual à banda disponível no enlace de saída da conexão. A banda efetiva de uma conexão está entre seus valores de taxa média e de taxa de pico de transmissão da fonte, sendo que a banda efetiva de "n" conexões agregadas é geralmente menor que "n" vezes a banda efetiva de uma conexão simples. Esse fenômeno é conhecido como ganho da multiplexação estatística.

Muitas pesquisas sobre banda efetiva têm focado a razão de perda de células (CLR), como sendo a única métrica de QoS. O CLR para "n" conexões pode ser determinado como função do tamanho do *buffer* (BU), da capacidade do enlace (TE) e de outros parâmetros do sistema. Dessa forma poderíamos determinar a banda efetiva de "n" conexões através da inversão da fórmula ou através de interações. Porém, na maioria das vezes, a função não está disponível explicitamente. [ANI82] apresenta um modelo chamado fluxo de fluído para o caso de "n" fontes *on/off*¹³ idênticas e *buffer* com tamanho infinito. Devido à premissa de *buffer* infinito, a razão para perda de células para um *buffer* de tamanho BU é aproximado pela probabilidade $P(x > BU)$ na qual a ocupação do *buffer* exceda BU . Diversas abordagens para estimar a banda equivalente surgiram a partir do modelo de fluxo de fluído. Uma simplificação computacional desse modelo inicial foi proposta por [GUE91]. A abordagem binomial simples é descrita em [MUR91] onde o autor propõe aproximar a razão de perda de células (CLR) à probabilidade de que a taxa de transmissão de células de todas as "n" conexões não excedam a capacidade do enlace, assumindo, dessa forma, *buffers* de tamanho zero. Comparações desses e outros métodos de obter a banda equivalente estão em [REG94] e [SYK93]. As limitações desses métodos são:

¹³ Fonte de tráfego com transmissões baseadas em rajadas, ou seja, períodos de atividade (*on*) de transmissão intercalados com períodos de silêncio (*off*) na transmissão.

- ❑ Consideram somente a razão de perda de células como parâmetro de QoS. Entretanto, para restringir o atraso da célula, assim como a perda, é necessário basear o cálculo de banda equivalente não somente em estimativas de células perdidas, mas também em estimativas do atraso de célula.
- ❑ Os resultados alcançados são relativamente precisos somente para probabilidades de perda muito baixas ($< 10^{-6}$). Em alguns casos pode ser impossível ou desnecessário atingir esses níveis de requisitos, por exemplo, tráfego de voz não demanda muito em termos de probabilidade de perda, aproximadamente 10^{-3} [ONV95].
- ❑ As fontes de transmissão das conexões precisam ser tratadas analiticamente. O modelo de tráfego utilizado com frequência é o *On/Off* de rajada. A abordagem desses métodos requer que várias conexões sejam modeladas através da combinação de fontes *On/Off*, o que muitas vezes não é fácil de fazer analiticamente.
- ❑ Tratam todas as conexões envolvidas como sendo de fluxos homogêneos¹⁴.

É importante que o método de avaliação da banda equivalente capture o relacionamento não linear entre a banda efetiva total de uma combinação de conexões e a banda efetiva individual de cada conexão, especialmente quando o ganho estatístico é alto. Tanto em [REG94] como em [GUE91] foram feitas proposições de funções analíticas não lineares da banda efetiva individual para determinar a banda efetiva agregada de tráfego. O método proposto em [REG94] elimina a primeira limitação citada anteriormente, porém ainda sofre com a terceira limitação. Esse método também adiciona complexidade, pois cada conexão é convertida empiricamente em uma fonte *on/off* recebida por um *buffer* de tamanho zero. Já a fórmula desenvolvida em [GUE91] é simples, porém superestima consideravelmente a banda efetiva das fontes causando o uso ineficiente dos enlaces. O trabalho desenvolvido em [RAM98] propõe uma alternativa aos métodos analíticos para a caracterização da banda equivalente. Nesse trabalho os autores utilizam o conceito de redes neurais ALN (*Adaptive Logic Network*) treinadas a partir de estimativas de perda e atraso feitas com simulações. Essas estimativas podem então ser usadas para calcular a banda efetiva satisfatória. O objetivo desse método foi desvincular a complexidade do problema da quantidade de conexões envolvidas.

Utilizando os argumentos feitos em [RAM98], o método proposto nesta dissertação para determinar os recursos (banda equivalente e *buffer*) alocados necessários por conexões VBR é feito por meio da previsão de desempenho através da simulação da multiplexação de fluxos. A simulação provê estimativas de atraso e de perda de células para diferentes valores de configuração de banda equivalente e *buffers*. Conforme os requisitos de QoS das conexões, é possível determinar a configuração adequada de recursos que atende à qualidade esperada.

O próximo capítulo descreve a implementação da simulação para estimar a alocação de recursos para tráfego VBR. Devido à sua característica de alocação estatística, o dimensionamento dos recursos das conexões de categoria de serviço VBR será realizado por meio de previsão de desempenho baseado em simulação.

¹⁴ Fluxos homogêneos são conexões independentes com mesmos parâmetros de tráfego e requisitos de QoS.

4 Simulação para Dimensionamento de Capacidade

4.1 Simulação: Vantagens e Limitações

O projeto, a análise e a implementação de redes ATM são tarefas complexas e trabalhosas. Em geral, três soluções podem ser utilizadas para auxiliar na execução dessas tarefas: a experimentação com redes ATM reais, a utilização de métodos analíticos e a simulação computacional. Cada uma dessas soluções possui vantagens e limitações, e embora não sejam mutuamente exclusivas, cada uma delas é melhor aplicável a uma determinada situação particular.

A experimentação com redes ATM reais é a única solução que pode prover uma "visão real" do estado da rede. Ela oferece uma visão razoável a respeito da qualidade de serviço provida pela rede e permite o estudo de como expansões podem ser feitas. Entretanto, o planejamento de expansões a partir de medidas observadas em redes reais possui uma série de desvantagens. Primeiro, o tráfego de redes reais é extremamente variável e imprevisível, e portanto sua utilidade para fins de dimensionamento é bastante limitada e até mesmo inviável de ser utilizado para o planejamento de novas redes e serviços. Segundo, essa solução pode ser muito dispendiosa. Dependendo do sistema e do nível de detalhe desejado, medições diretas podem requerer um conhecimento completo do ambiente, equipamentos de custo elevado e pessoas especializadas.

A utilização de métodos analíticos puros e a simulação computacional, segundo [TOW98], encontram-se na extremidade de um espectro de possíveis técnicas para a avaliação de desempenho e dimensionamento de redes de comunicações. A utilização de métodos analíticos consiste em substituir uma rede de comunicações por um modelo matemático que possa ser resolvido analiticamente. Dessa forma, o problema de avaliação de desempenho e dimensionamento de redes é transformado em um problema matemático, que pode ser resolvido através de técnicas matemáticas. O método analítico, comparado com a simulação, é muito rápido de ser resolvido computacionalmente. A limitação é que esse método tem sérias restrições no contexto de redes de alta velocidade. A fim de manter os modelos analiticamente tratáveis, várias simplificações são feitas, produzindo assim conclusões bastante limitadas em termos de precisão.

A simulação computacional é a mais flexível das soluções mencionadas, e encontra-se na outra extremidade do espectro de possíveis técnicas para avaliação de desempenho e dimensionamento de redes de comunicações. Entre as vantagens da simulação de redes está o modelamento realista de fontes de tráfego, a facilidade de realização de experimentos, a comparação de mecanismos de controle de tráfego, o modelamento com diferentes níveis de detalhes e escalas de tempo de observação, a estimação precisa de probabilidade de perda e atrasos de células, etc. Porém, todas essas vantagens têm seu preço: a simulação computacional demanda um alto desempenho de processamento. Simulação é adequada para resolver problemas envolvendo questões de tempo, comportamentos de fluxos heterogêneos e estados da rede que podem acontecer e são importantes para entender. A quantidade de elementos simulados aumenta a quantidade de eventos a serem tratados pelo simulador. Portanto, para viabilizar a

execução da simulação em tempos razoáveis é necessário limitar a topologia de rede a ser simulada.

Alguns produtos de simulação utilizam a técnica orientada a eventos estocásticos discretos. Estocástico significa randômico. Através de métodos matemáticos para gerar seqüência de números pseudo-randômicos, é possível imitar variações aleatórias que ocorrem em sistemas reais. A habilidade permitida pela randomização é uma das grandes vantagens da simulação. Eventos discretos significam que nós visualizamos o sistema com o tempo progredindo de um evento importante para outro, em vez das mudanças de tempo ocorrerem de forma contínua. Isso permite a representação computacional eficiente do tempo, sendo a grande vantagem da simulação orientada a eventos. Outras técnicas de simulação são: orientada pelo tempo, orientada por dados e simulação mista, que é uma mistura das duas técnicas [SHA94].

4.2 Simulador ATM-HFC

O produto escolhido para implementar o modelo foi a ferramenta de simulação orientada a eventos ATM/HFC [GOL98] desenvolvida no *National Institute of Standards and Technology* (NIST) dos Estados Unidos da América, desde 1995. A ferramenta é composta por diferentes modelos de elementos, cada qual com seus atributos, que podem ser combinados para caracterizar determinadas configurações de rede que se deseja avaliar. Os elementos podem ser desde representações de tipos de comutadores ATM ou redes HFC (*Hybrid Fiber Coax*) até diferentes técnicas de controle de tráfego a serem utilizadas na simulação. A carga de serviço na simulação é provida por elementos modeladores que caracterizam diferentes tipos de aplicações geradoras de tráfego, permitindo arranjos de serviços CBR, VBR, ABR e UBR utilizando seus respectivos parâmetros descritores.

A escolha da ferramenta de simulação foi baseada nos estudos feitos nos trabalhos [JIR00] e [ALB98], onde foram comparadas as características de diferentes produtos disponíveis. Os destaques na escolha do NIST ATM/HFC são:

- Código fonte dos programas, feitos na linguagem C, é aberto para alterações e inclusões de novos módulos;
- Possibilidade de execuções interativas através de interface gráfica desenvolvida em X Window System;
- Executável em diferentes plataformas UNIX, incluindo LINUX que é conhecido por sua estabilidade e disponibilidade de produtos de licença livre e código aberto;
- A experiência de 14 anos do NIST americano em desenvolver ferramentas de simulação, depositando confiança e validade na implementação do ATM/HFC;
- A experiência adquirida pelo grupo de redes do Programa de Pós-Graduação em Ciência da Computação da UFRGS através dos estudos e utilização da ferramenta em outros trabalhos de dissertação.

Com o objetivo de ser utilizado como ferramenta de apoio no planejamento de capacidade de redes ATM, esse produto é capaz de simular diferentes configurações de topologia e carga de tráfego obtendo estatísticas sobre estimativas de utilização de enlaces, taxa de perda de células e outras medidas de desempenho relevantes. Também

é possível, através da inclusão de componentes, utilizar o simulador NIST ATM/HFC para avaliar o desempenho de novos mecanismos de controle de tráfego.

4.2.1 Conceitos do Simulador

O simulador é implementado por um conjunto de componentes que trocam eventos entre si para caracterizar a transmissão de células pela rede. O produto possui um gerenciador de eventos que providencia as funções gerais para o escalonamento dos eventos. Os eventos são organizados em uma fila pela ordem crescente de tempo de seu acontecimento (tempo previsto para o "disparo"). Portanto, a execução das ações de determinado evento acontece do primeiro para o último na fila. O gerenciador de eventos mantém um relógio interno que é alterado constantemente conforme o tempo do primeiro evento da fila. Dessa forma, o tempo desse relógio, após a execução de um evento, salta para o tempo do próximo evento na fila. Esse método caracteriza um simulador orientado a eventos.

O relógio interno da simulação utiliza uma unidade de tempo denominada *ticks* que podem representar, através da configuração do produto, qualquer grandeza de tempo em segundos. A conversão de *ticks* para segundos utilizada foi a padrão encontrada no produto, um *tick* é igual a 10 nanossegundos. O relógio interno é mantido em uma variável sem sinal com 8 *bytes*, o que possibilitaria, na teoria, a execução de simulações com tempo simulado de enorme grandeza¹⁵. O tempo de execução computacional da simulação é geralmente maior que o tempo simulado. O tempo gasto para executar uma simulação aumenta conforme a quantidade de componentes e eventos que devem ser tratados na simulação. Para uma noção dos tempos envolvidos, a tabela 4.1 mostra resultados obtidos na simulação do cenário Sim_01 (fig. 4.1) com o produto ATM/HFC versão 4.1 do NIST.

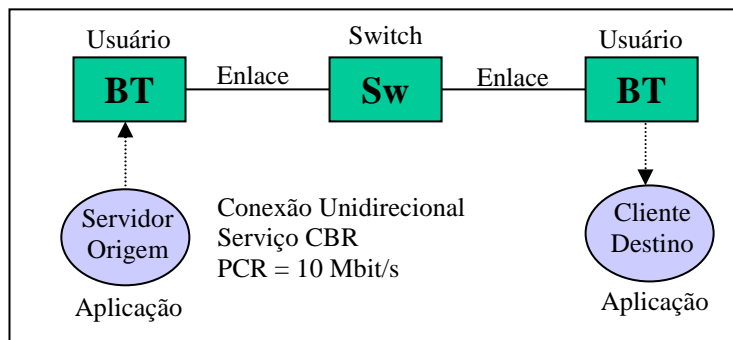


FIGURA 4.1 - Cenário de Simulação SIM_01

TABELA 4.1 - Relação do tempo simulado e tempo de execução

Cenário	Tempo Simulado	Tempo Real (Execução Batch)	Células Geradas	Tam. do Arquivo de Resultados (MB)
Sim_01	3s	26s - 0:00:26	70.754	1,17
Sim_01	30s	265s - 0:04:25	707.546	13,13
Sim_01	42s	370s - 0:06:10	990.565	18,53

Plataforma: PC Pentium II 230 Mhz, 32 MB de memória

¹⁵ Utilizando a unidade de ticks de 10 nanossegundos e sabendo que a quantidade máxima de ticks que podem ser representados por uma variável sem sinal de 8 bytes é dada por $2^{64} - 1$, o tempo simulado máximo é aproximadamente de 5.849,42 anos.

4.2.2 Componentes do Simulador

Componentes são os blocos básicos de construção do simulador. O processo de simulação consiste em vários componentes que são combinados entre si para formarem o ambiente a ser simulado. Os tipos de componentes do NIST ATM/HFC estão agrupados em 4 classes, conforme segue:

- Comutadores ATM;
- Equipamentos Terminais dos usuários;
- Aplicações ATM;
- Enlaces de transmissão.

Algumas classes possuem diferentes tipos de componentes para implementar diferentes configurações de rede. Por exemplo, uma aplicação ATM pode gerar tráfego à taxa constante de transmissão de células ou à taxa variável que é definida por determinada função de distribuição. Cada componente consiste de um programa em C onde estão definidas as ações que devem ser executadas para os eventos previstos para o componente em questão. Cada instância¹⁶ de um componente tem sua própria estrutura de dados que é utilizada para armazenar informações que caracterizam o componente.

A classe de comutadores ATM tem dois tipos de componentes definidos: RBSWITCH e SWITCH. O primeiro implementa algoritmos de controle de tráfego para serviços ABR e diversos esquemas de gerenciamento de *buffers* para serviços UBR. O segundo é um comutador genérico com as funções básicas de multiplexação de células. Esses dois tipos de comutadores não estão disponíveis simultaneamente para configuração de cenários de simulação. A escolha por um ou outro é feita no momento da compilação do NIST ATM/HFC.

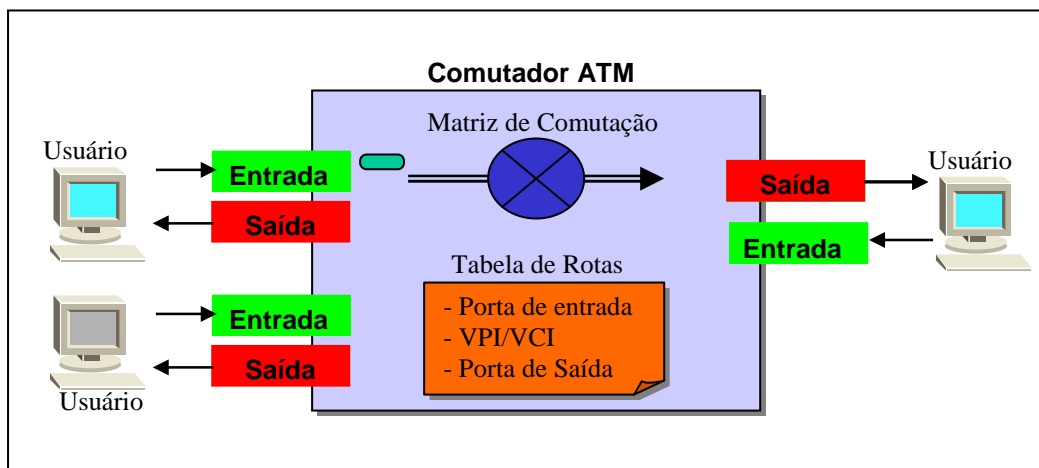


FIGURA 4.2 - Ações simuladas de um comutador ATM

O funcionamento do comutador ATM é baseado no encaminhamento das células que chegam para as portas de saída. Cada comutador possui uma tabela local de roteamento contendo basicamente: uma identificação de conexão e o número da porta

¹⁶ Terminologia de OOP (Programação Orientada a Objetos) que significa criar um objeto a partir de uma classe.

de saída. Para exemplificar o funcionamento da simulação de um comutador ATM, consideremos uma célula chegando a partir de um enlace de transmissão (fig. 4.2). No próximo período de tempo de comutação, após algum atraso¹⁷ configurado no simulador, a célula é encaminhada para a porta de saída. O encaminhamento é baseado na identificação de conexão no cabeçalho da célula e na tabela de rotas do comutador. Se a porta de saída não possui um intervalo de tempo (*slot time*) para a transmissão da célula, ela é colocada em um fila de espera (buffer) para ser transmitida. O comutador do NIST ATM/HFC implementa uma estrutura de filas por grupo baseada em prioridade de atendimento conforme a categoria de serviço (fig. 4.3). A fila que recebe a maior prioridade no atendimento de suas células é a CBR/VBR, seguida pelas filas ABR e UBR, nessa ordem, significando que somente quando a fila CBR/VBR estiver vazia, a fila ABR será atendida. Portanto, a fila UBR que possui a menor prioridade só será atendida para transmitir suas células quando as outras duas filas estiverem vazias. Se o número de células na fila excede determinado valor configurado para controle de tráfego, um indicador de congestionamento é acionado para a porta. O indicador de congestionamento é desligado quando as três filas estiverem abaixo do valor predeterminado. O tamanho de cada fila (capacidade de armazenar células) é configurado para cada comutador. Ultrapassando seu limite de armazenamento, as células excedentes são descartadas e contabilizadas para cada fila na porta.

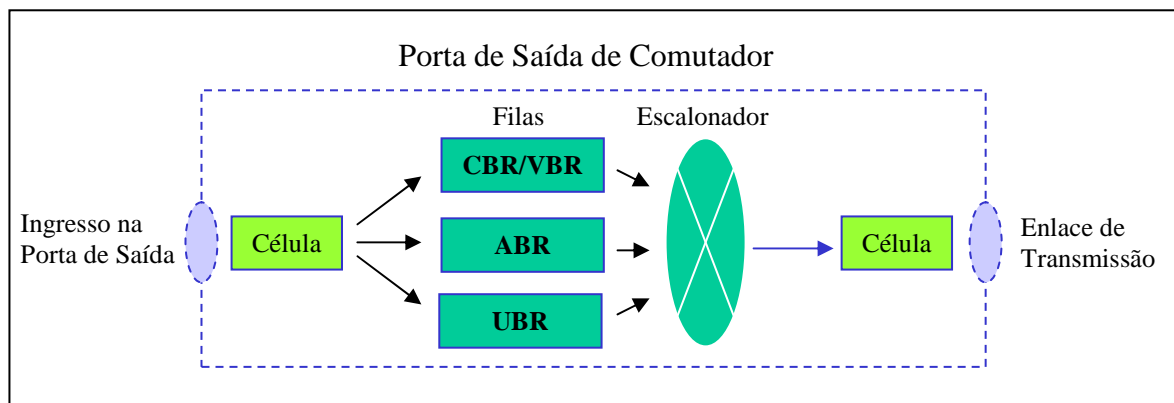


FIGURA 4.3 - Estrutura de filas e escalonador por prioridade de categoria de serviço

A classe de equipamentos terminais de usuários simula dispositivos físicos que requisitam conexões da rede como computadores, centrais telefônicas, codificadores de sinais de vídeo, e outros. Sobre cada equipamento são executadas uma ou mais aplicações. São três os tipos de componentes definidos nessa classe: RBBTE, BTE e HFC. O primeiro implementa algoritmos de controle de fluxo baseado em taxa para serviços ABR. O segundo é um terminal genérico com as funções básicas de acesso à rede. Esses dois tipos de equipamentos terminais não estão disponíveis simultaneamente para configuração de cenários de simulação. A escolha por um ou outro é feita no momento da compilação do NIST ATM/HFC. O terceiro componente dessa classe, o HFC, é utilizado para simular redes híbridas de fibra e cabo coaxial para serviços do tipo de TV a cabo.

As ações simuladas dos equipamentos terminais são: receber as células das aplicações e enviá-las pelo enlace de transmissão (fig. 4.4). Se a porta de saída não

¹⁷ Atraso interno simulado para caracterizar o atraso fixo de comutação de células que todo comutador possui intrinsecamente.

possui um intervalo de tempo (*slot time*) para a transmissão da célula, essa é colocada em uma das três filas de prioridade (CBR/VBR, ABR ou UBR). O tamanho máximo de cada fila de saída é configurável e, da mesma forma que nos comutadores, as células que excederem o limite são descartadas. Os parâmetros que podem ser monitorados para essa classe são: número de células em cada fila de saída, o número de células descartadas e o número de células recebidas pela rede.

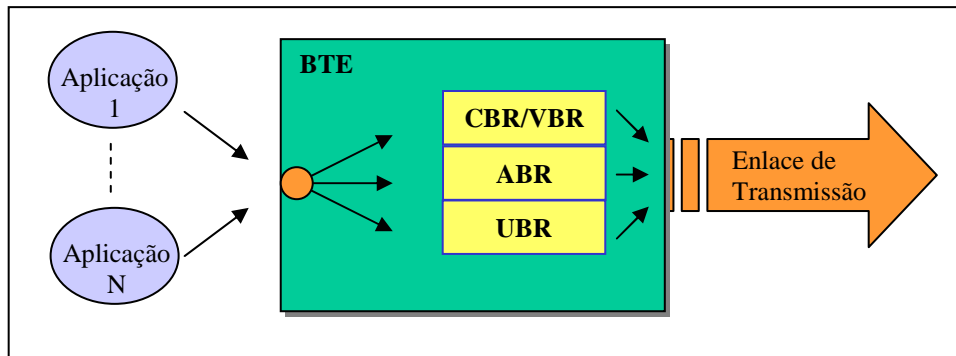


FIGURA 4.4 - Simulação do acesso dos equipamentos terminais à rede

A classe de aplicações ATM tem a função de gerar o tráfego de células sobre os equipamentos terminais. A fonte de tráfego gerada por esses componentes pode ser basicamente constante ou variável. Porém, com a utilização de diferentes métodos de geração de tráfego, essa classe totaliza 12 diferentes tipos de componentes (tab. 4.2). Os métodos de geração de tráfego podem ser agrupados em: Constante, Poisson, Batch, Auto-similar, TCP/IP e MPEG. No método de geração Constante, as células são geradas pela aplicação durante toda a simulação na taxa especificada na configuração do componente. Para o método utilizando Poisson [JAI91], o tráfego gerado tem o comportamento de rajada (*ON/OFF*) seguindo uma distribuição exponencial tanto para os períodos de atividade como os de silêncio. A configuração do método de Poisson é especificada pelo tamanho médio da rajada e pelo intervalo médio entre as rajadas. No método Batch, o gerador de tráfego é configurado especificando o número médio de células geradas durante uma rajada e o intervalo médio entre as rajadas. Os métodos Auto-similar, TCP/IP e MPEG são específicos na forma de caracterização do gerador e detalhes sobre suas utilizações podem ser encontrados em [GOL98].

TABELA 4.2 - Tipos de Aplicações ATM simuladas

Categoria de Serviço	Gerador de Tráfego
CBR	Constante
VBR	Poisson
VBR	Batch
VBR	Constante Batch
ABR	Constante
ABR	Poisson
ABR	Batch
ABR	Auto-similar
UBR	TCP/IP
ABR	TCP/IP
VBR	MPEG
VBR	MPEG GGBAR

Fonte: [GOL98]

A última classe de componentes que resta para completar o ambiente simulado é a de enlaces de transmissão. Essa classe possui apenas um tipo de componente. Esse componente simula o meio físico por onde as células são transmitidas de um equipamento para outro. A caracterização desses componentes permite considerar a taxa de transmissão do enlace e o comprimento do enlace para cálculo dos atrasos de transmissão e propagação, respectivamente. O parâmetro monitorado na simulação é a taxa de utilização do enlace.

4.3 Método da Simulação

Esse trabalho apresenta uma proposta de método para o dimensionamento de capacidade de redes ATM que pode ser utilizado como apoio ao projeto de rede. A proposta é baseada na avaliação de desempenho obtida pela simulação, orientada a eventos, do comportamento de tráfego sobre comutadores ATM. O modelo utilizado na simulação é caracterizado por elementos fundamentais que atuam no desempenho dos serviços, tais como: tamanho do *buffer*, modelo de tráfego, quantidade de conexões, largura de banda do enlace, entre outros.

Para determinada carga de tráfego são executadas diferentes simulações variando os recursos disponíveis dentro de parâmetros prováveis. As medidas de desempenho obtidas nas simulações são resultados estatísticos médios da razão de perda de células (CLR), do atraso de transferência de células (CTD) e da variação do atraso de células (CDV). O resultado das simulações é apresentado em um conjunto de curvas de desempenho em função da alocação de recursos. Conhecendo o desempenho desejado (QoS) pela carga de serviço, pode-se utilizar o gráfico do conjunto de curvas de desempenho para determinar a quantidade necessária de recursos que satisfazem os requisitos de QoS. Diferentes configurações de redes e cargas de serviços podem ser exercitadas para revelar limites necessários de alocação de recursos que evitem dificuldades de desempenho. A figura 4.5 mostra o exemplo de um cenário de simulação para três determinadas conexões no qual desejamos descobrir qual a quantidade de banda que devemos alocar na porta de multiplexação de forma que o QoS das conexões seja atingido.

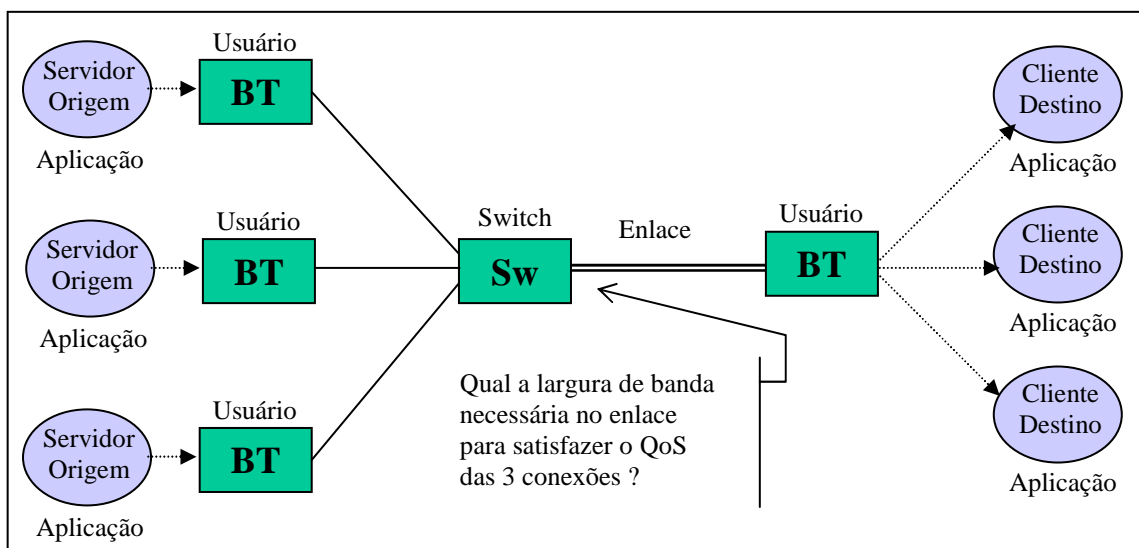


FIGURA 4.5 - Cenário de Simulação

Para descobrir o valor da largura de banda que satisfaz o fluxo agregado, o simulador experimenta inúmeras execuções variando a banda alocada na porta do multiplexador. Sabendo que a banda efetiva que satisfaz o QoS para conexões VBR está entre a taxa média e a taxa de pico [DZI97], fica estipulado o intervalo de valores para largura de banda a ser experimentado. Tomemos como exemplo determinada conexão com taxa média de transmissão de células igual a 30% da sua taxa de pico. Nessas condições para um grupo homogêneo desse tipo de conexão, o simulador executará "n" experimentos variando a largura de banda de 30% do PCR a 100% do PCR. O número "n" de simulações é determinado conforme a seguinte expressão:

$$N = \left(\frac{100 - Tx_Media}{Incr} + 1 \right) * Cf_Buffer * Qt_Grupos$$

Onde,

N é a quantidade de simulações a serem executadas;

Tx_Media é a porcentagem do PCR que a taxa média de transmissão corresponde. No exemplo acima, o valor desse parâmetro é 30;

$Incr$ é o incremento adotado na variação da largura de banda alocada. As execuções das simulações se diferenciam por esse incremento. Um exemplo para esse parâmetro é adotar o incremento da banda de 2% em 2% da taxa de PCR;

Cf_Buffer é a quantidade de valores desejáveis para configurar o tamanho do *buffer* da porta de multiplexação. Por exemplo, podemos adotar 4 valores: 10, 100, 1.000 e 10.000 células de capacidade. O simulador executará para cada um dos valores;

Qt_Grupos é a quantidade de grupos de conexões que desejamos experimentar. Por exemplo, podemos executar a simulação para 3 grupos sendo que o primeiro é formado pela combinação de 2 conexões, o segundo por 5 conexões e o terceiro por 25 conexões. Dessa forma temos o comportamento de desempenho para fluxos agregados de 2, 5 e 25 conexões homogêneas.

Aplicando os valores exemplificados, temos:

$$N = \left(\frac{100 - 30}{2} + 1 \right) * 4 * 3$$

$$N = 432$$

De cada uma das 432 execuções teremos os parâmetros de desempenho médios: CTD, CDV e CLR. Com esses resultados são montadas curvas de desempenho em função da banda alocada, do tamanho do *buffer* e da quantidade de conexões agregadas. A figura 4.6 mostra o comportamento do CTD em função da banda alocada (Banda Efetiva de 40% a 100% do PCR do fluxo agregado) e da quantidade de conexões multiplexadas (10 grupos: 1 conexão, 2 conexões, 3, 4, 5, 6, 8, 10, 15 e 25 conexões).

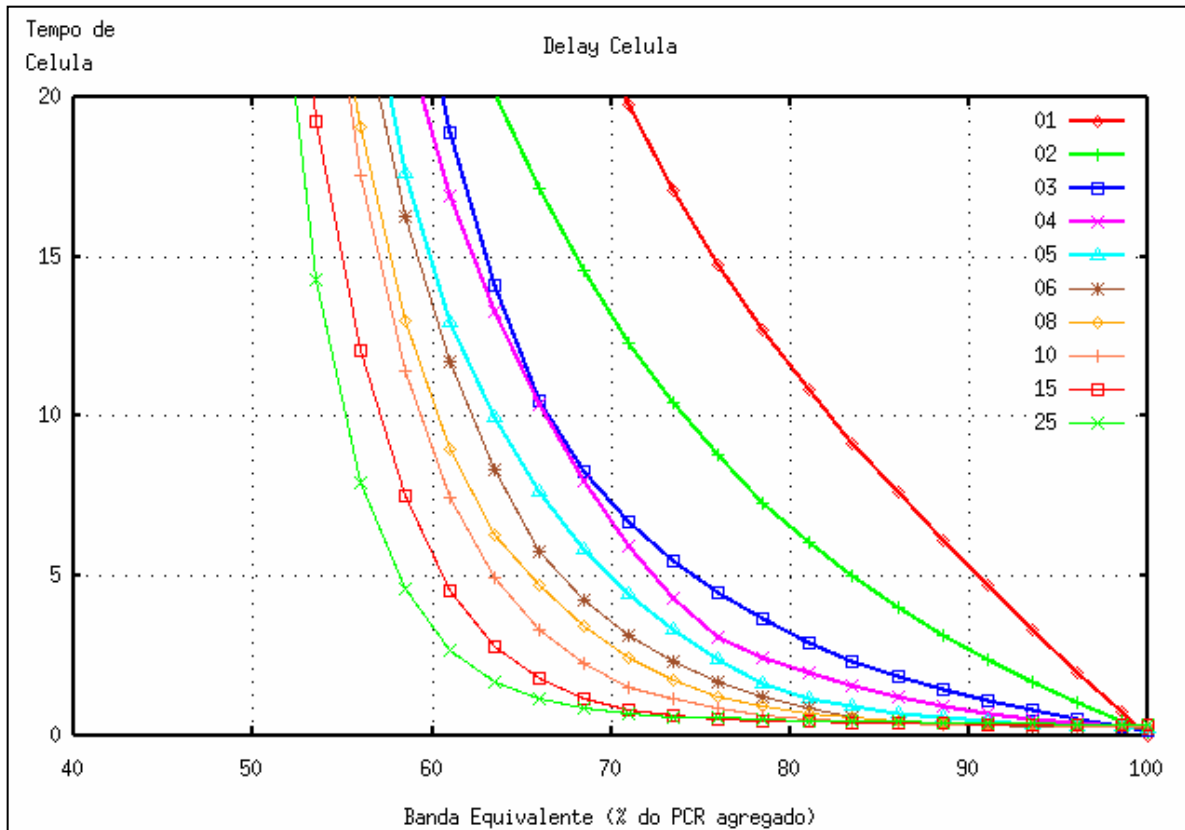


FIGURA 4.6 - Exemplo de Curva de Desempenho obtida na simulação

Observando as curvas de desempenho (fig. 4.6) podemos concluir que para atingir o QoS de atraso máximo de 5 tempos de célula¹⁸ devemos alocar banda conforme a tabela 4.3.

TABELA 4.3 - Banda Efetiva Aproximada

Quantidade de Conexões que formam o fluxo agregado	Banda Efetiva Alocada em relação à Taxa de Pico total do fluxo agregado
1	91%
2	84%
3	75%
4	73%
5	70%
6	68%
8	66%
10	64%
15	61%
25	58%

¹⁸ Tempo de célula é o tempo necessário para transmitir 424 bits (1 célula) na taxa de transmissão do enlace. É referenciado como *time slot*.

4.3.1 Estimação do Intervalo de Confiança: Método de Subintervalos

As inferências ou predições referentes ao comportamento do sistema devem ser baseadas nos resultados experimentais obtidos na simulação. Devido ao fato da simulação possuir elementos aleatórios, as saídas de uma simulação são amostras observadas de variáveis aleatórias. Em conseqüência, qualquer conclusão que fizermos quanto à operação do sistema real deve levar em consideração o intervalo de confiança¹⁹ dos resultados obtidos pela simulação. Uma variável aleatória de grande importância na simulação é a média estatística das amostras. A importância desse resultado é que a média do processo real pode ser estimada a partir da média das amostras obtidas nas simulações.

A saída da execução de uma simulação é uma seqüência de medidas que dependem da seqüência de números aleatórios utilizada. Repetindo a execução da simulação com outra seqüência de números aleatórios obteremos um resultado diferente do primeiro. Portanto, para chegarmos a um valor estimado do comportamento do sistema real, devemos avaliar estatisticamente os resultados das "n" execuções da simulação. No desenvolvimento clássico do intervalo de confiança para a média, assume-se que as amostras são independentes e identicamente distribuídas. Pelo teorema do limite central [SOA90], a distribuição da soma de "n" amostras independentes de "x" se aproxima da distribuição normal para valores de "n" maiores que 30. A maioria dos procedimentos utilizados na estimativa dos parâmetros ou testes de hipóteses se baseia na média amostral. Se uma variável aleatória provém de uma amostra normalmente distribuída, então a média amostral também é normalmente distribuída. Se a variável aleatória provém de uma amostra que não é normalmente distribuída, mas de tamanho suficientemente grande ($n \geq 30$), então o teorema do limite central mostra que a média amostral tende à curva normal.

Se assumirmos que a média de "n" amostras é normalmente distribuída ($n > 30$), então podemos afirmar, com o nível de confiança $1 - \alpha$, que o parâmetro da população que nos dá a média de uma variável aleatória X, encontra-se dentro do intervalo de confiança a seguir:

$$\left(\bar{X}_n - Z_{\alpha/2} * \frac{s}{\sqrt{n}}, \bar{X}_n + Z_{\alpha/2} * \frac{s}{\sqrt{n}} \right)$$

Onde,

\bar{X}_n é média das amostras. Cada amostra corresponde ao resultado de uma execução de simulação;

n é o número de amostras. Corresponde ao número de replicações (execuções) da simulação;

$Z_{\alpha/2}$ é o valor dado pela curva da distribuição normal quando a área é igual $\alpha/2$;

¹⁹ Intervalo de confiança, ou intervalo de estimação, é um intervalo onde o parâmetro da população deve ser encontrado com certa probabilidade. A probabilidade dada é chamada nível de confiança [SOA90].

α é o nível de significância utilizado para calcular o nível de confiança. O nível de confiança é igual a $100 * (1 - \alpha) \%$ ou, em outras palavras, um α de 0,05 indica um nível de confiança de 95%; s é o desvio padrão da média das amostras.

O desvio padrão das amostras é calculado com a seguinte expressão:

$$s = \sqrt{\frac{n \sum_{i=1}^n x_i^2 - \left(\sum_{i=1}^n x_i \right)^2}{n(n-1)}}$$

Onde,

x_i é o valor da amostra "i";
n é o número de amostras.

Para a avaliação do intervalo de confiança para a média de uma variável aleatória é importante que tenhamos várias amostras independentes resultantes da execução do simulador. Há vários métodos para estimar o intervalo de confiança [SOA90], entre eles citamos dois métodos clássicos:

- Replicação - que emprega execuções separadas do simulador, com cada execução sendo considerada uma rajada para o cálculo de uma única amostra da variável aleatória " X_i ", definida como a média na execução "i" (fig. 4.7). A vantagem desse método é a independência entre cada amostra. Como desvantagens desse método podemos citar: o fato de que cada replicação contém um segmento inicial de instabilidade significativo, e o fato de que apenas uma amostra é obtida em cada replicação, o que pode significar um grande tempo gasto em execuções pelo simulador.

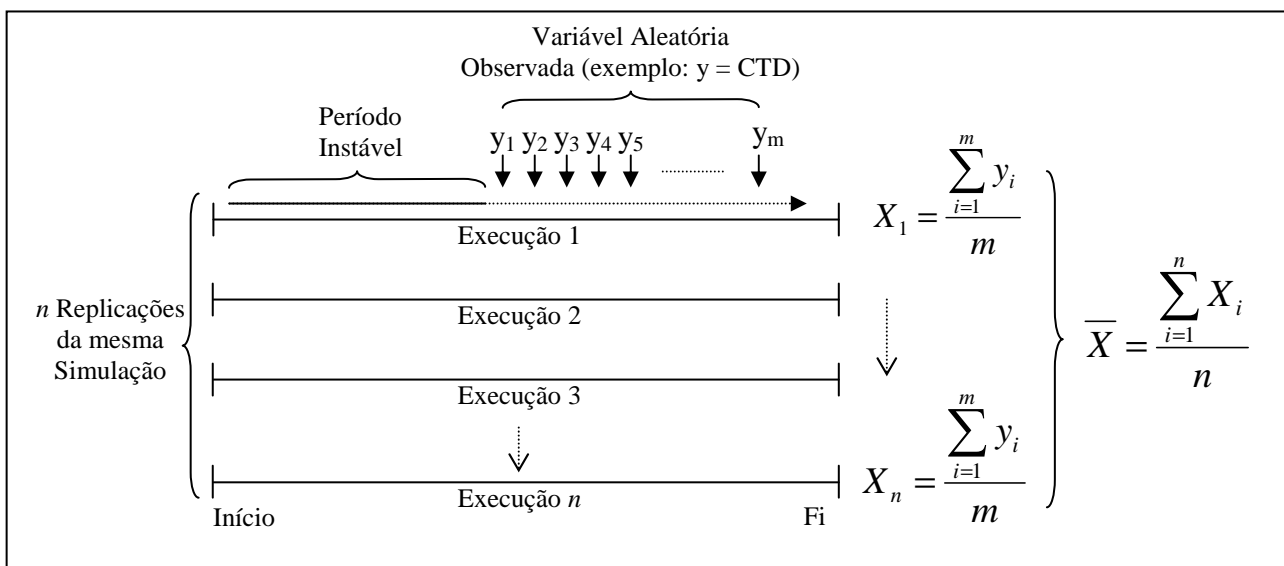


FIGURA 4.7 - Intervalo de Confiança: Método de Replicação da Simulação

- Subintervalos - que divide uma execução em iguais subintervalos (rajadas), e calcula um valor de " X_i " como uma média em cada subintervalo de " i " (fig. 4.8). Assume-se que " X_i " sejam independentes. A vantagem do uso de subintervalos na estimativa dos intervalos de confiança é que uma única execução pode ser feita na obtenção da estimativa, além disso, apenas um período transiente é excluído dos resultados. A desvantagem do procedimento é a necessidade do estabelecimento da largura do subintervalo, de forma a torná-los independentes.

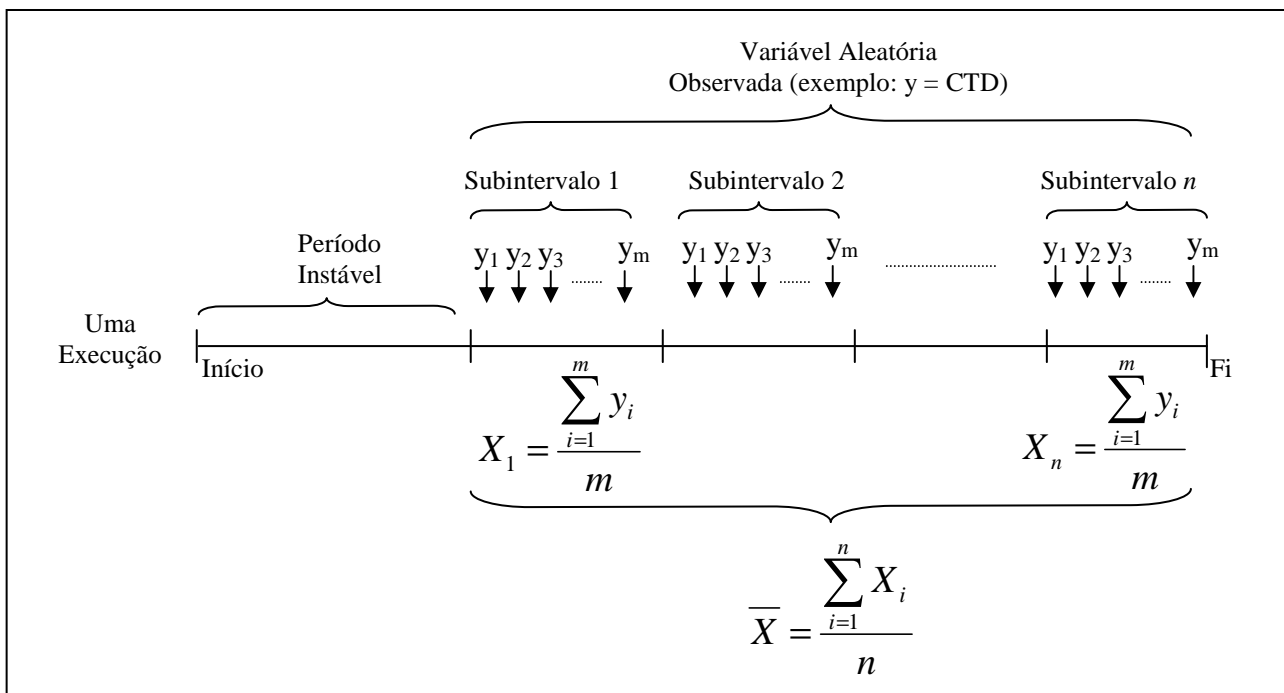


FIGURA 4.8 - Intervalo de Confiança: Método de Subintervalos

Considerando a vantagem de eliminar a redundância de reiniciar o processo de simulação, evitando assim a reincidência de períodos de instabilidade, o método escolhido neste trabalho para a estimação do intervalo de confiança é o método de divisão da execução da simulação em subintervalos. Colaborando na decisão da escolha do método, foi considerada também a facilidade de adaptação do produto NIST ATM/HFC incorporando funções estatísticas que permitissem o trabalho interno dos resultados pelo simulador. Se optássemos por replicação, a computação dos resultados obtida em cada simulação deveria ser feita por um programa externo ao simulador, perdendo a possibilidade de aproveitamento da execução gráfica e interativa do NIST ATM/HFC.

Como foi visto, a desvantagem do procedimento é a necessidade do estabelecimento da largura do subintervalo, de forma a tornar cada subintervalo independente. A suposição de independência é sempre feita na análise por simulação, embora sempre exista uma autocovariância entre os valores no fim de um subintervalo, e aqueles no começo do próximo subintervalo. Esse fato pode causar uma covariância entre as medidas em cada subintervalo, que deve decrescer à medida que a largura do subintervalo cresce, existindo vários procedimentos que podem ser utilizados para a

determinação de uma largura aceitável para o subintervalo [FIS78]. A seguir é demonstrado como chegamos ao valor do tamanho do intervalo utilizado em nossas simulações.

Consideramos as seguintes premissas na escolha do tamanho do subintervalo:

- ❑ A configuração do tamanho do subintervalo por unidade de tempo de simulação. Se escolhêssemos delimitar o subintervalo por unidades de células geradas, não teríamos como analisar os resultados com base no tempo decorrido porque a taxa de geração de células de uma simulação para outra varia;
- ❑ A quantidade de subintervalos deve ser superior a 30 unidades para utilizarmos o cálculo do intervalo de confiança baseado na distribuição normal;
- ❑ O NIST ATM/HFC tem um tempo máximo simulado igual a 48 segundos. Conforme o trabalho de [JIR00] com o simulador NIST ATM/HFC, o tempo mais adequado para alcançar a estabilidade da simulação genericamente é 8 segundos. Retirando os primeiros oito segundos do total de 48, sobra um tempo válido de 40 segundos de tempo simulado. Nessas condições (40 segundos de duração e mínimo de 30 subintervalos), temos o tamanho máximo do subintervalo em 1,3 segundo;
- ❑ O tamanho deve ser suficientemente grande para garantir independência entre os subintervalos subsequentes. A preferência é que ocorram dentro de um subintervalo as mudanças de estado das fontes várias vezes. Por exemplo, a transmissão de vídeo à taxa de 30 quadros por segundo delimita em média as rajadas (mudança de estado) a cada 33 ms;
- ❑ À medida que a quantidade de subintervalos aumenta (tamanho do subintervalo diminui), a variância da média das amostras aproxima-se de zero, permitindo um intervalo de confiança proporcionalmente menor em relação ao centro do intervalo (fig. 4.9).

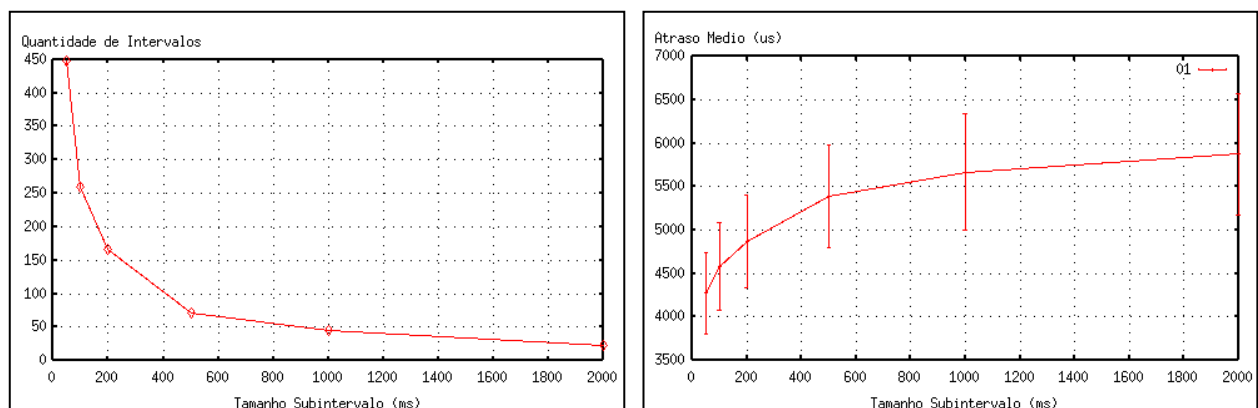


FIGURA 4.9 - Quantidade de Subintervalos (esquerda) e Intervalo de Confiança de 99% para Atraso (direita) em função do Tamanho do Subintervalo utilizado

Os gráficos comparativos (fig. 4.9 e fig. 4.10) são referentes à simulação de uma fonte VBR de vídeo, com PCR igual a 6 Mbit/s, tempo de transmissão (*on*) médio igual a 5.000 μ s e tempo de silêncio (*off*) médio igual a 29.000 μ s. A banda efetiva alocada foi de 50% do PCR. Esse mesmo cenário foi simulado com diferentes tamanhos de subintervalos: 50 ms, 100 ms, 200 ms, 500 ms, 1.000 ms e 2.000 ms.

Considerando as premissas, o valor mais adequado para tamanho do subintervalo é o de 500 ms, pelos seguintes motivos:

- Visando a independência das amostras, garante na média 15 mudanças de estado para a transmissão de vídeo à taxa de 30 quadros por segundo. Se utilizássemos o tamanho de 200 ms, teríamos uma média de 6 mudanças de subintervalos (200ms/33ms);
- É possível observar até 80 amostras de subintervalos em cada execução do simulador, mais de duas vezes e meia o valor mínimo de 30 amostras para o cálculo do intervalo de confiança. Se utilizássemos o tamanho de 1.000 ms, teríamos no máximo 40 amostras para observar;
- O tempo de convergência do intervalo de confiança para valores próximos de +10% e -10% da média das amostras é de 38 segundos (fig. 4.10). Utilizando esse parâmetro para encerrar a simulação, podemos reduzir o tempo de execução de cada simulação. Se utilizássemos o tamanho de 1.000 ms, a convergência não ocorre antes de encontrar o limite de 48 segundos do NIST ATM/HFC;
- A quantidade de células observadas em cada intervalo é mais que o dobro do que se utilizássemos o tamanho de 200 ms (fig. 4.10). Quanto maior o número de amostras dentro do subintervalo, maior é a precisão do cálculo da média observada do processo real.

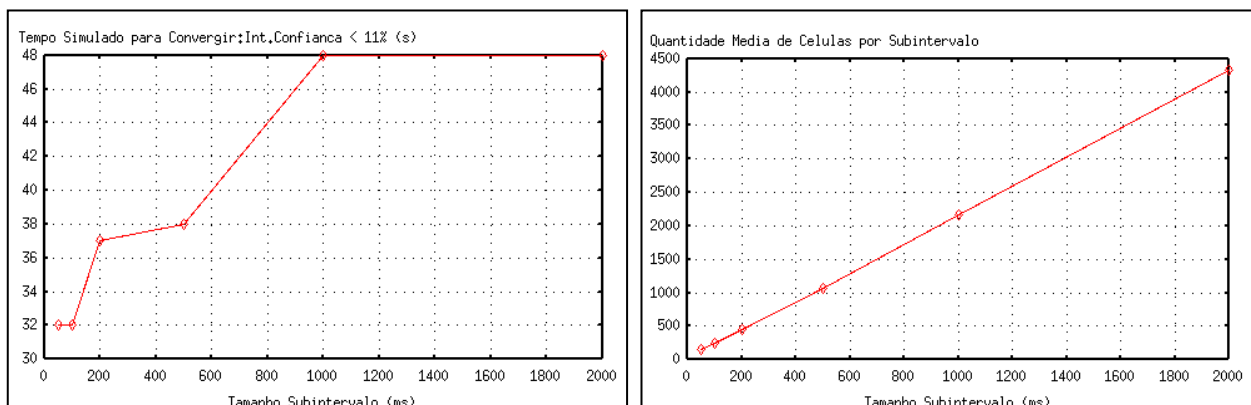


FIGURA 4.10 - Tempo Simulado (esquerda) e Quantidade de Células Analisadas (direita) em função do Tamanho do Subintervalo utilizado

Utilizando o método de divisão em subintervalos podemos estimar o intervalo de confiança para os valores médios estatísticos de CTD e CDV observados na simulação. O nível de confiança utilizado para cálculo do intervalo de confiança em todas as simulações deste trabalho é de 99%.

4.3.2 Estimativa de CTD

O atraso para a transferência de células da origem até o destino da conexão é calculado como a diferença entre o tempo de chegada e o tempo de partida da célula. Portanto, o cálculo é feito na chegada de cada célula conforme a expressão:

$$CTD_y = T_{chegada} - T_{partida}$$

Para tanto, é necessário que a célula carregue a informação do seu tempo de partida. Essa informação não está presente no formato de campos padrão da célula ATM. O simulador NIST ATM/HFC, na sua versão original, também não dispõe de processo que marque a célula na geração com o tempo de partida (*timestamp*). Devido a isso, fizemos as alterações necessárias para colocar o *timestamp* na célula. O anexo 3 detalha as alterações realizadas para adequar o simulador NIST ATM/HFC.

No final de cada subintervalo da simulação, é calculada a média de CTD entre todas as células recebidas na conexão, conforme expressão:

$$\overline{CTDsub}_x = \frac{\sum_{y=1}^m CTD_y}{m}$$

Onde,

\overline{CTDsub}_x é a média do CTD do subintervalo "x";
 CTD_y é o valor do CTD para a célula "y";
 m é a quantidade de células recebidas no subintervalo "x".

No final da simulação após "n" subintervalos, é calculado o CTD médio para a conexão em questão, conforme expressão:

$$CTDcn_z = \frac{\sum_{x=1}^n \overline{CTDsub}_x}{n}$$

Onde,

$CTDcn_z$ é o CTD médio para a conexão "z";
 n é o número de subintervalos da simulação.

Para calcular o CTD final da simulação consideramos a média entre todas as conexões homogêneas²⁰ observadas, conforme expressão:

²⁰ Conexões homogêneas são fluxos independentes que possuem em comum os mesmos valores descritores de tráfego (PCR, SCR, MBS, ...).

$$CTD_{final} = \frac{\sum_{z=1}^p CTDcn_z}{p}$$

Onde,

CTD_{final} é o CTD entre todas as conexões estabelecidas na execução do simulador;

p é o número de conexões simuladas.

Também no final da simulação, é calculado o intervalo de confiança para o CTD de cada conexão utilizando as seguintes expressões:

$$s^2 = \frac{n \sum_{i=1}^n (\overline{CTDsub}_i)^2 - \left(\sum_{i=1}^n \overline{CTDsub}_i \right)^2}{n(n-1)}$$

$$s = \sqrt{s^2}$$

Onde,

\overline{CTDsub}_i é a média do CTD do subintervalo "i";

n é o número de amostras de CTD. Cada amostra de CTD corresponde ao valor calculado em um subintervalo;

s^2 é a variância das amostras de CTD calculado em cada subintervalo;

s é o desvio padrão das amostras de CTD calculado em cada subintervalo.

O intervalo de confiança para cada conexão é dado pelas seguintes expressões:

$$Erro_z = 2,95 * \frac{s}{\sqrt{n}}$$

$$IC_z = [CTDcn_z - Erro_z, CTDcn_z + Erro_z]$$

Onde,

$CTDcn_z$ é o CTD médio para a conexão "z";

IC_z é o intervalo de confiança da conexão "z";

$Erro_z$ é o valor que determina o limite superior e inferior do intervalo de confiança para a conexão "z". A constante "2,95" da expressão refere-se à probabilidade de 99% de acerto de que o CTD estará dentro dos limites do intervalo de confiança;

s é o desvio padrão das amostras de CTD calculado em cada subintervalo;

n é o número de subintervalos em que a simulação foi dividida.

O intervalo de confiança final, considerando todas as conexões simuladas em uma execução, é dado pelas seguintes expressões:

$$Erro_{final} = \frac{\sum_{z=1}^p Erro_z}{p}$$

$$IC_{final} = [CTD_{final} - Erro_{final}, CTD_{final} + Erro_{final}]$$

Onde,

CTD_{final} é o CTD calculado entre todas as conexões estabelecidas na execução do simulador;

p é o número de conexões simuladas.

IC_{final} é o intervalo de confiança final para o CTD da simulação;

$Erro_z$ é o valor que determina o limite superior e inferior do intervalo de confiança para a conexão "z";

$Erro_{final}$ é a média de erro padrão entre todas as conexões.

O resultado final apresentado pelo simulador inclui o intervalo de confiança para o CTD (IC_{final}).

4.3.3 Estimativa de CDV

A variação do atraso para a transferência de células da origem até o destino da conexão é baseada no método de CDV de dois pontos que é dado pela seguinte expressão:

$$CDV2P_y = |CTD_y - CTD_{ref}|$$

Onde,

$CDV2P_y$ é o CDV de dois pontos da célula "y". É sempre um valor positivo;

CTD_y é o valor do CTD para a célula "y";

CTD_{ref} é o CTD de referência que é definido como sendo o CTD calculado para a primeira célula da conexão.

A definição do CDV de dois pontos ainda está em estudo pelo ITU-T. A escolha do método de CDV de dois pontos em vez do CDV de um ponto deve-se ao fato de que esse último inclui a variabilidade da fonte e nós queremos avaliar somente a variabilidade imposta pela rede.

O método de CDV de dois pontos utiliza como CTD_{ref} o atraso da primeira célula da conexão. O mais adequado seria utilizar como CTD_{ref} a média de CTD da conexão e assim medir a variação em torno desse valor padrão. Como o cálculo do CDV

é feito célula a célula, não podemos utilizar a média como referência pois não temos como calculá-la antecipadamente. Entretanto, como dividimos a simulação em subintervalos, podemos utilizar o CTD médio do subintervalo anterior como CTD_{ref} para o subintervalo corrente. Dessa forma, a expressão para o cálculo do CDV para cada célula que chega é a seguinte:

$$CDV2P_y = \left| CTD_y - \overline{CTD}_{sub_{i-1}} \right|$$

Onde,

$CDV2P_y$ é o CDV de dois pontos da célula "y". É sempre um valor positivo;
 CTD_y é o valor do CTD para a célula "y";
 $\overline{CTD}_{sub_{i-1}}$ é a média do CTD do subintervalo "i-1";
i é o número do subintervalo corrente.

Baseado nessa expressão, o simulador não calcula o CDV para as células do primeiro subintervalo. Nos subintervalos subsequentes, o CDV é calculado normalmente conforme a expressão acima.

No final de cada subintervalo da simulação, é calculada a média de CDV entre todas as células recebidas na conexão, conforme expressão:

$$\overline{CDV}_{sub_x} = \frac{\sum_{y=1}^m CDV_y}{m}$$

Onde,

\overline{CDV}_{sub_x} é a média do CDV do subintervalo "x";
 CDV_y é o valor do CDV para a célula "y";
m é a quantidade de células recebidas no subintervalo "x".

No final da simulação, após "n" subintervalos, é calculado o CDV médio para a conexão em questão, conforme expressão:

$$CDVcn_z = \frac{\sum_{x=1}^n \overline{CDV}_{sub_x}}{n}$$

Onde,

$CDVcn_z$ é o CDV médio para a conexão "z";
n é o número de subintervalos da simulação.

Para calcular o CDV final da simulação consideramos a média entre todas as conexões homogêneas observadas, conforme expressão:

$$CDV_{final} = \frac{\sum_{z=1}^p CDVcn_z}{p}$$

Onde,

CDV_{final} é o CDV entre todas as conexões estabelecidas na execução do simulador;

p é o número de conexões simuladas.

Também no final da simulação, é calculado o intervalo de confiança para o CDV de cada conexão utilizando as seguintes expressões:

$$s^2 = \frac{n \sum_{i=1}^n (\overline{CDVsub}_i)^2 - \left(\sum_{i=1}^n \overline{CDVsub}_i \right)^2}{n(n-1)}$$

$$s = \sqrt{s^2}$$

Onde,

\overline{CDVsub}_i é a média do CDV do subintervalo "i";

n é o número de amostras de CDV. Cada amostra de CDV corresponde ao valor calculado em um subintervalo;

s^2 é a variância das amostras de CDV calculado em cada subintervalo;

s é o desvio padrão das amostras de CDV calculado em cada subintervalo.

O intervalo de confiança para cada conexão é dado pelas seguintes expressões:

$$Erro_z = 2,95 * \frac{s}{\sqrt{n}}$$

$$IC_z = [CDVcn_z - Erro_z, CDVcn_z + Erro_z]$$

Onde,

$CDVcn_z$ é o CDV médio para a conexão "z";

IC_z é o intervalo de confiança da conexão "z";

$Erro_z$ é o valor que determina o limite superior e inferior do intervalo de confiança para a conexão "z". A constante "2,95" da expressão refere-se à probabilidade de 99 % de acerto de que o CDV estará dentro dos limites do intervalo de confiança;

s é o desvio padrão das amostras de CDV calculado em cada subintervalo;

n é o número de subintervalos em que a simulação foi dividida.

O intervalo de confiança final, considerando todas as conexões simuladas em uma execução, é dado pelas seguintes expressões:

$$Erro_{final} = \frac{\sum_{z=1}^p Erro_z}{p}$$

$$IC_{final} = [CDV_{final} - Erro_{final}, CDV_{final} + Erro_{final}]$$

Onde,

CDV_{final} é o CDV calculado entre todas as conexões estabelecidas na execução do simulador;

p é o número de conexões simuladas.

IC_{final} é o intervalo de confiança final para o CDV da simulação;

$Erro_z$ é o valor que determina o limite superior e inferior do intervalo de confiança para a conexão "z";

$Erro_{final}$ é a média de erro padrão entre todas as conexões.

O resultado final apresentado pelo simulador inclui o intervalo de confiança para o CDV (IC_{final}). A figura 4.11 mostra um resumo da seqüência de cálculos utilizados pelo simulador para estimar tanto o CTD como o CDV.

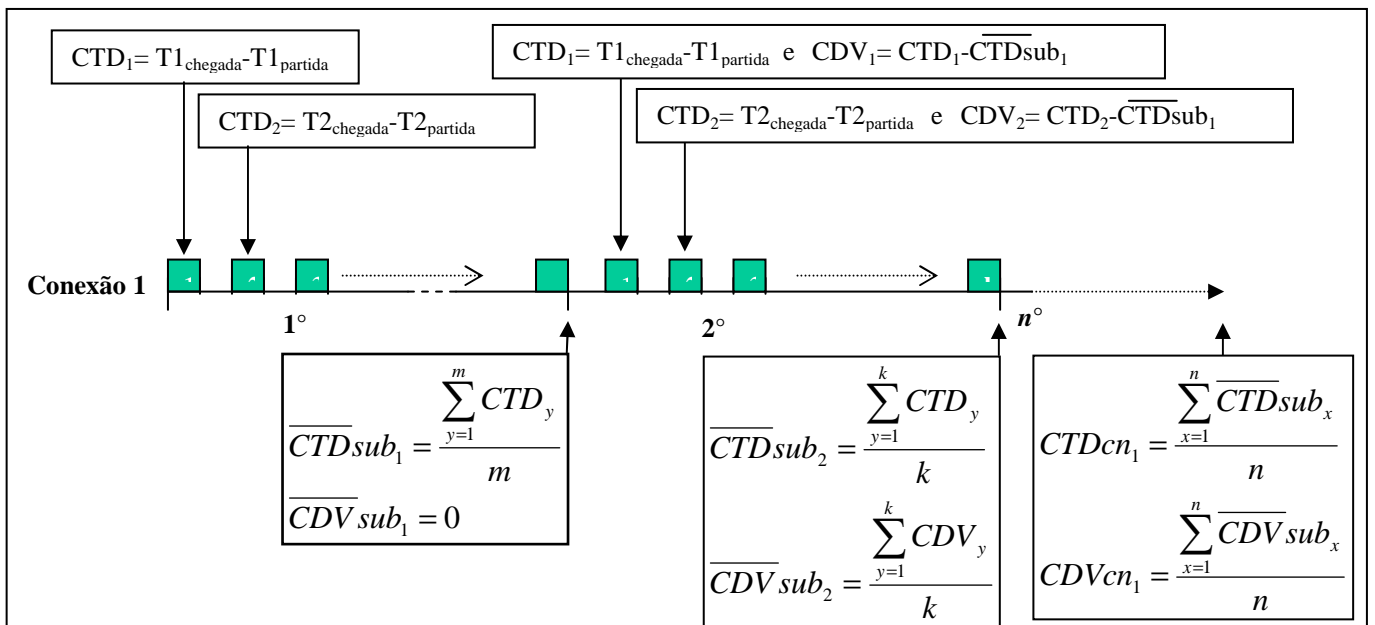


FIGURA 4.11 - Estimativa do CTD e CDV para uma conexão

4.3.4 Estimativa de CLR

A razão de células perdida de uma conexão é dada por:

$$CLR = \frac{Cel_Desc}{Cel_Env}$$

Onde,

Cel_Desc é a quantidade de células que foram descartadas por transbordo de buffer no comutador ATM;

Cel_Env é a quantidade de células enviadas pela fonte geradora do fluxo.

O intervalo de tempo de medição não é padronizado, mas geralmente é representando como o período de duração da conexão. Seguindo o método de divisão da simulação em subintervalos, utilizamos o tempo de duração do subintervalo como período de contabilização das células descartadas e das células enviadas. Portanto, no final de cada subintervalo da simulação é feito o cálculo da razão de perda de células. Segue expressão utilizada:

$$CLR_{sub_x} = \frac{Cel_Desc}{Cel_Env}$$

Onde,

CLR_{sub_x} é a razão de perda de células do subintervalo "x";

Cel_Desc é a quantidade de células descartadas no subintervalo "x";

Cel_Env é a quantidade de células enviadas no subintervalo "x".

A quantidade de células geradas depende do tamanho do subintervalo utilizado (no nosso caso 500 ms) e da taxa de transmissão da fonte que varia de um subintervalo para outro (fontes VBR). Em nossos experimentos tivemos valores médios de células por intervalo desde 1.000 a 100.000 células enviadas. Considerando 80 subintervalos por simulação, temos um total médio de 80.000 a 8.000.000 de células enviadas por conexão.

No final da simulação, é calculado o CLR de cada conexão como sendo a média entre todos os valores encontrados em cada subintervalo conforme expressão a seguir:

$$CLR_{cn_z} = \frac{\sum_{x=1}^n CLR_{sub_x}}{n}$$

Onde,

CLR_{cn_z} é o CLR médio para a conexão "z";

CLR_{sub_x} é a razão de perda de células do subintervalo "x";

n é o número de subintervalos da simulação.

Para calcular o CLR final da simulação consideramos a média entre todas as conexões homogêneas observadas, conforme expressão:

$$CLR_{final} = \frac{\sum_{z=1}^p CLRcn_z}{p}$$

Onde,

CLR_{final} é o CLR entre todas as conexões estabelecidas na execução do simulador;

p é o número de conexões simuladas.

Para o parâmetro de CLR não utilizamos o cálculo do intervalo de confiança, porque os métodos apresentados são baseados no teorema do limite central onde o intervalo de confiança é calculado para valores médios. Diferente dos procedimentos para CTD e CDV, a cada subintervalo não calculamos um valor médio de perda e sim um valor absoluto daquele subintervalo. O primeiro valor médio de CLR obtido é somente realizado no final da simulação que é o CLR da conexão em questão (CLRcn).

O resultado final apresentado pelo simulador inclui o CLR médio de todas as conexões (CLR_{final}). A figura 4.12 mostra um resumo da seqüência de cálculos utilizados pelo simulador para estimar o valor de CLR.

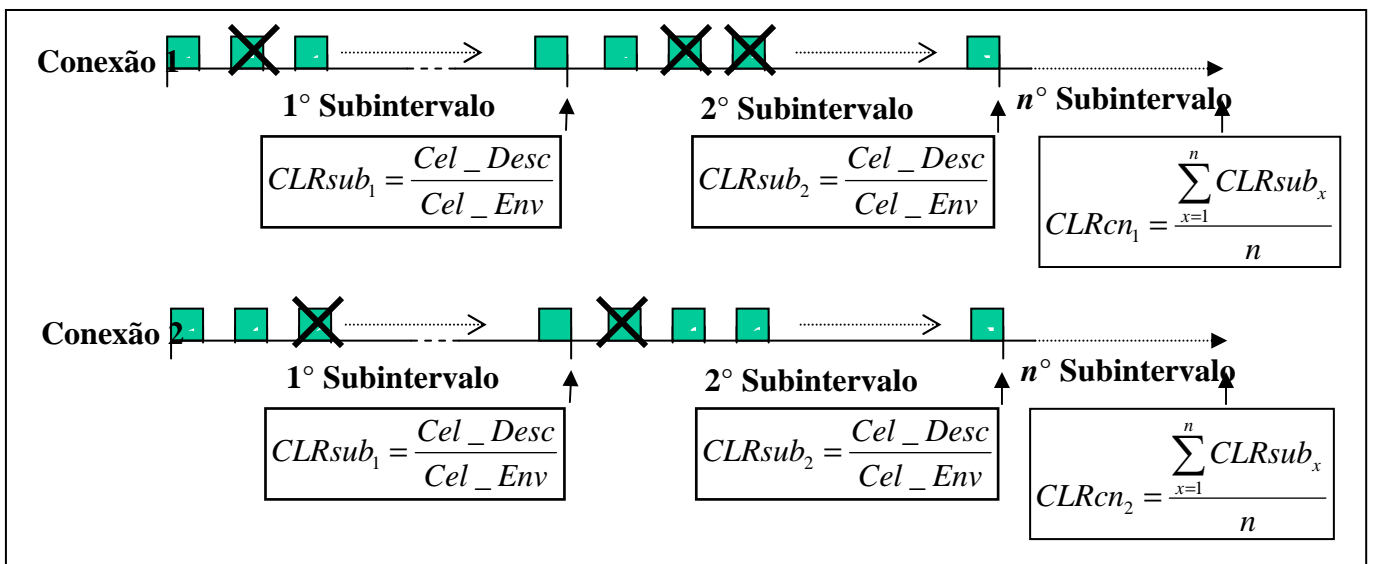


FIGURA 4.12 - Estimativa do CLR para duas conexões simuladas

4.4 Simulações para Determinar a Banda Equivalente de Tráfego VBR

O simulador NIST ATM/HFC foi modificado²¹ para apresentar os resultados de CTD, CDV e CLR utilizando o método de divisão em subintervalos. As alterações atuaram no cálculo e na forma de apresentação dos resultados. A parte central e principal do simulador responsável pela geração de tráfego e tratamento de eventos não sofreu modificações, permanecendo conforme a versão original.

O gerador de fontes utilizado para modelar o tráfego foi o VBR Poisson. Os períodos de atividade/silêncio (*on/off*) do tráfego são exponencialmente distribuídos através dos valores médios informados. A escolha do VBR Poisson nos experimentos é devida à simplicidade de caracterização das fontes, no entanto outros geradores de tráfego disponíveis no NIST ATM HFC (tab. 4.2) também podem ser utilizados. Para demonstrar o método de determinar a banda equivalente, utilizamos exemplos de aplicações de transmissão de áudio e vídeo (tab. 4.4). Os exemplos são baseados em medições realizadas pelo Instituto de Ciência da Computação de Würzburg (Alemanha) usando um codificador MPEG-I com as seguintes características: 25 quadros por segundo, quadro de 384 por 288 *pixels* com 12 bits para informação de cor e 40.000 quadros observados por evento (aproximadamente 30 minutos de vídeo).

TABELA 4.4 - Serviços Simulados

Serviço	Evento	Tamanho Médio do Quadro (bit)	Relação Pico/Média
Transmissão de TV	Corrida de Carro	30.749	6,6
Vídeo Conferência	Desktop	6.031	7.7
Vídeo Sob Demanda	Silêncio dos Inocentes	7.312	18,4

Fonte:[ROB96]

Para cada serviço são gerados diversos experimentos (simulações) variando a quantidade de conexões multiplexadas e a alocação de recursos. A execução dos experimentos é controlada a partir de arquivos de comandos (*scripts*). Detalhes sobre como executar as simulações, e os tipos de arquivos utilizados nas configurações, estão no anexo 2.

4.4.1 Caracterização do tráfego para o Simulador

O gerador de tráfego VBR Poisson do NIST ATM/HFC necessita três parâmetros para caracterização do tráfego:

- PCR, dado em Mbit/s;
- Tamanho médio da rajada, dado em microssegundos (μ s);
- Tamanho médio do intervalo entre rajadas, dado em μ s.

Para converter as informações da tabela 4.4 para os parâmetros mencionados anteriormente, devemos utilizar os cálculos a seguir para cada um dos serviços.

²¹ As modificações feitas nos programas que compõem o NIST ATM/HFC estão relacionadas no anexo 3.

Informação Inicial

TQM - Tamanho Quadro Médio, dado em bits;

TQ - Taxa de Quadros, dado em quadros/s

PM - Razão Pico/Média.

Calculamos então,

$$TTM = TQM \times TQ$$

$$PCR = TTM \times PM$$

$$Ton = \frac{TQM}{PCR}$$

$$Toff = \frac{1}{TQ} - Ton$$

Onde,

TTM é a taxa de transmissão média, dada em bit/s;

TQM é o tamanho médio do quadro;

TQ é a taxa de transmissão de quadros por segundo;

PCR é a taxa de transmissão de pico, dada em bit/s;

PM é a razão entre a taxa máxima de transmissão e a taxa média;

Ton é o tempo médio de duração da rajada (transmissão na taxa de pico);

$Toff$ é o tempo médio entre as rajadas (inatividade da transmissão).

Dessa forma, temos as informações necessárias para gerar o tráfego com o modelo VBR Poisson. Seguem os valores calculados para os três serviços utilizados como exemplo.

TABELA 4.5 - Parâmetros Descritores VBR Poisson

Serviço	Evento	PCR (bit/s)	Ton (μ s)	Toff (μ s)
Transmissão de TV	Corrida de Carro	5.073.585	6.060	33.940
Vídeo Conferência	Desktop	1.160.967,5	5.190	34.810
Vídeo Sob Demanda	Silêncio dos Inocentes	3.365.520	2.000	38.000

Nas próximas seções são apresentados os resultados das simulações para dimensionamento da capacidade dos recursos para os serviços referenciados (tab. 4.5).

4.4.2 Experimentos com diferentes configurações de *buffer*

Para cada serviço da tabela 4.5, geramos cenários de simulação com diferentes configurações de capacidade de *buffer*: 100, 1.000 e 10.000 células. Observando os resultados obtidos, notamos que para valores de atraso baixos ($< 500 \mu\text{s}$) temos gráficos de alocação de banda equivalente semelhante entre os três casos de configuração de *buffer*. As figuras 4.13 e 4.14 comparam o atraso médio (CTD) na fila de capacidade de 100 células com a fila de capacidade de 10.000 células para o serviço de transmissão de TV da tabela 4.5. O mesmo acontece com CDV. Essa semelhança é explicada para esses serviços pelo fato de que nas taxas de bandas equivalentes necessárias para garantir CTD menor que $500 \mu\text{s}$ há baixíssima perda de células para filas de 100 células e muito menor ainda para filas de 10.000 células (fig. 4.15). Levando os dois cenários para a mesma situação: sem acúmulo de células na fila.

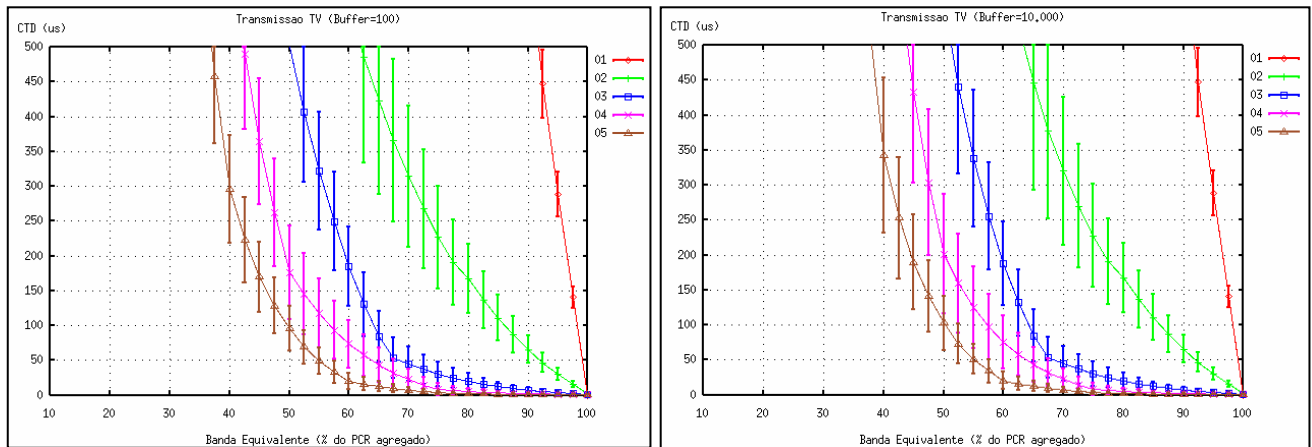


FIGURA 4.13 - CTD para Multiplexação de 1 a 5 conexões com *buffer* de 100 células (esquerda) e com *buffer* de 10.000 células (direita)

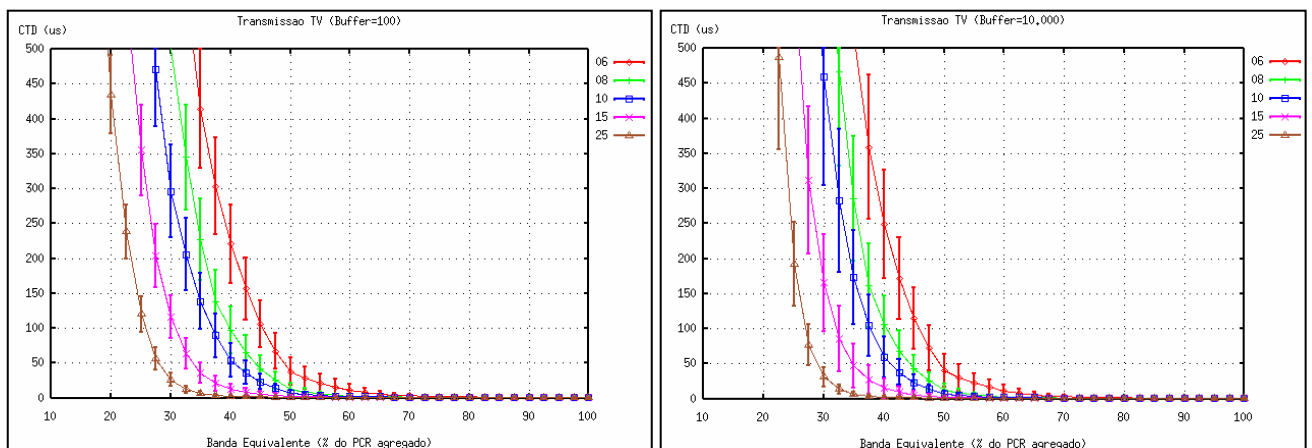


FIGURA 4.14 - CTD para Multiplexação de 6, 8, 10, 15 e 25 conexões com *buffer* de 100 células (esquerda) e com *buffer* de 10.000 células (direita)

Mantendo nosso interesse em valores de desempenho de atraso (CTD e CDV) menores que $500 \mu\text{s}$, conforme valores sugeridos no anexo 1, e considerando assim a semelhança para configurações da fila de 100, 1.000 e 10.000 células, apresentaremos

somente os resultados detalhados para a configuração de 100 células de *buffer* por ser o pior caso de configuração.

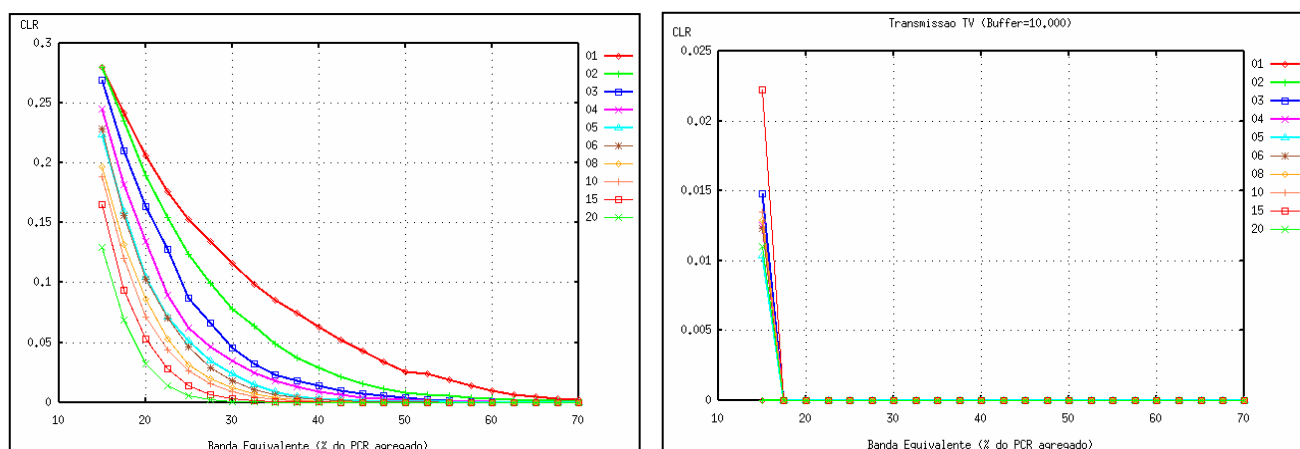


FIGURA 4.15 - CLR para Multiplexação de "*n*" conexões com *buffer* de 100 células (esquerda) e com *buffer* de 10.000 células (direita)

4.4.3 Resultados Obtidos

A tabela 4.6 mostra o resumo de execução para cada um dos conjuntos de serviços simulados. Lembramos que cada execução foi realizada três vezes: para 100, 1.000 e 10.000 células de *buffer*, tendo todas as execuções os mesmos valores apresentados na tabela 4.6. O computador utilizado nas execuções tem a seguinte configuração: processador K6-II 400 Mhz, 64 MB, sistema operacional *RedHat Linux 6.2 (kernel 2.2.14-5.0)*.

TABELA 4.6 - Cenários Simulados com 100 células de *buffer*

Informação		Transmissão TV	Vídeo Conferência	Vídeo Sob Demanda
Descrição de Tráfego	PCR (Mbit/s)	5,073585	1,1609675	3,363520
	Ton (μ s)	6.060	5.190	2.174
	Toff (μ s)	33.940	34.810	37.826
Configuração de Cenários	Quantidade Conexões	1-5, 6, 8, 10, 15 e 25	1-5, 6, 8, 10, 15 e 25	1-5, 6, 8, 10, 15 e 25
	Banda Equivalente (Incremento de 2,5 %)	15 a 100 % PCR (35 ocorrências)	12 a 100 % PCR (37 ocorrências)	5 a 100 % PCR (39 ocorrências)
Total da Execução	Num. Simulações	350	370	390
	Células Geradas	194.525.901	44.382.451	56.198.895
	Duração (h:m:s)	04:10:54	00:49:59	01:03:34

Transmissão TV

As figuras 4.16 e 4.17 mostram o desempenho de CTD e CDV para o serviço de transmissão de TV (tab. 4.6) com valores inferiores a 500 μ s. A figura 4.18 mostra a razão de perda de células (CLR) para esse serviço com *buffer* de 100 células no comutador. As figuras 4.19, 4.20 e 4.21 mostram o intervalo completo de CTD obtido

com 100, 1.000 e 10.000 células de *buffer*. O gráfico de CTD por unidade de tempo de célula corresponde ao número médio de células presentes no *buffer*.

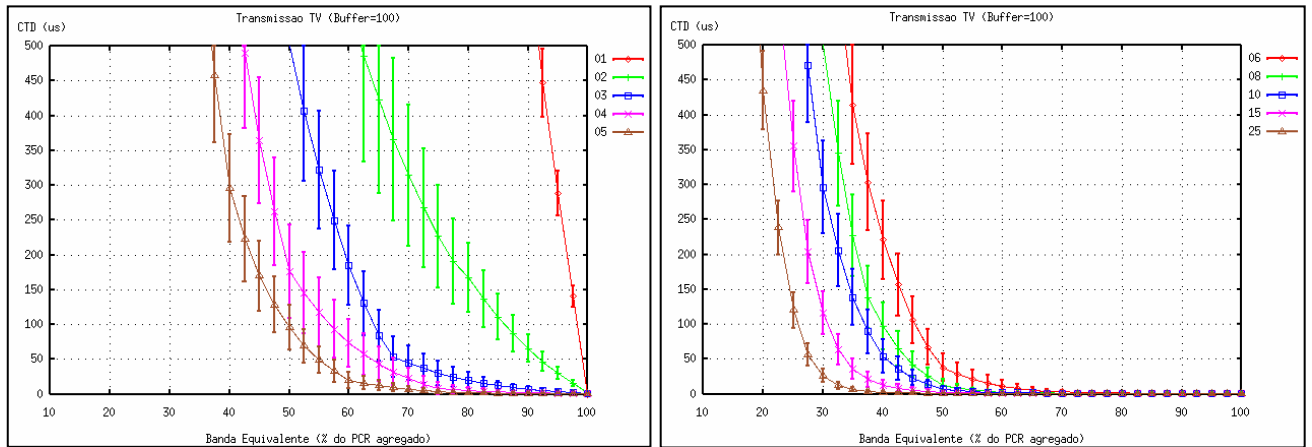


FIGURA 4.16 - CTD para Multiplexação de "*n*" conexões para transmissão de TV com *buffer* de 100 células.

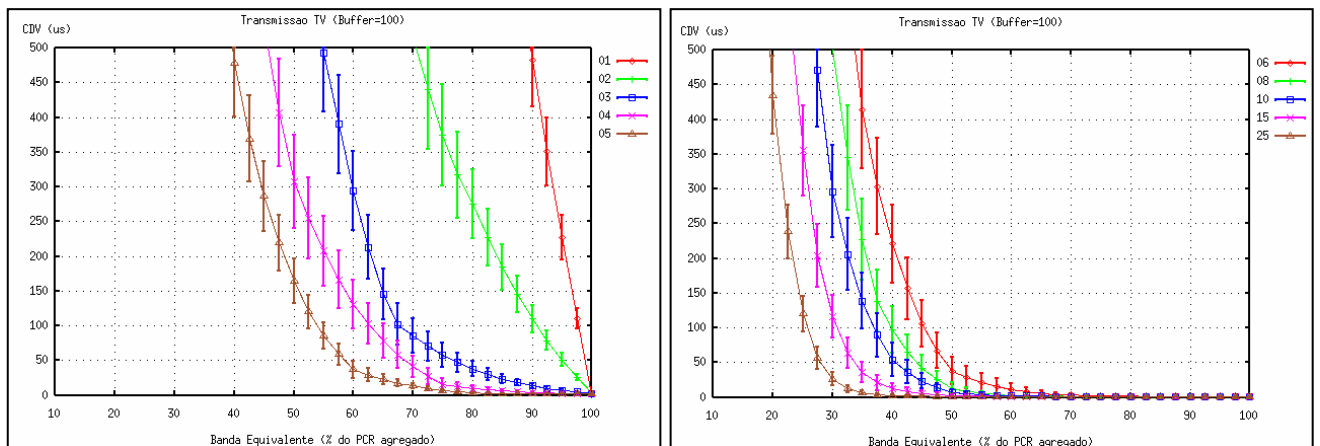


FIGURA 4.17 - CDV para Multiplexação de "*n*" conexões para transmissão de TV com *buffer* de 100 células

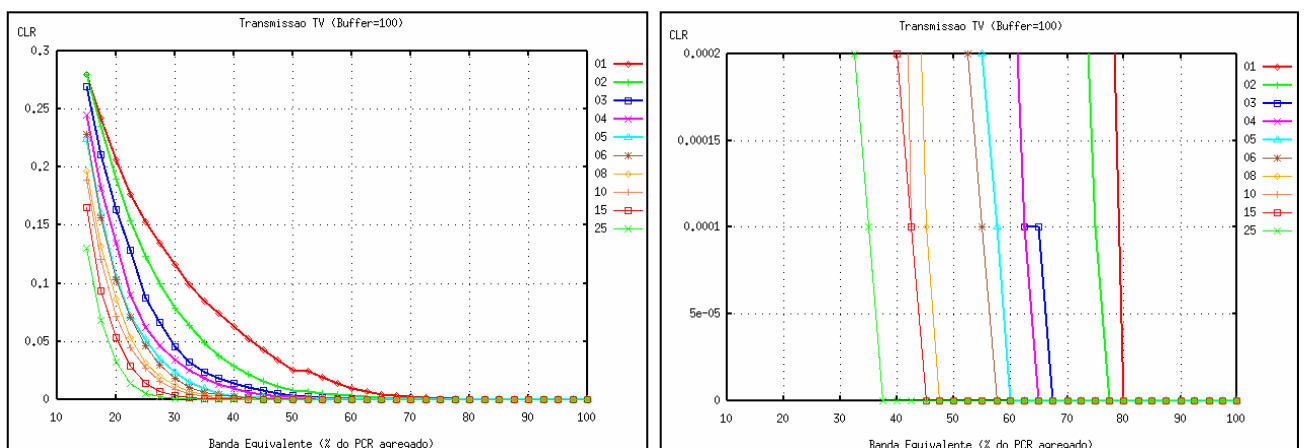
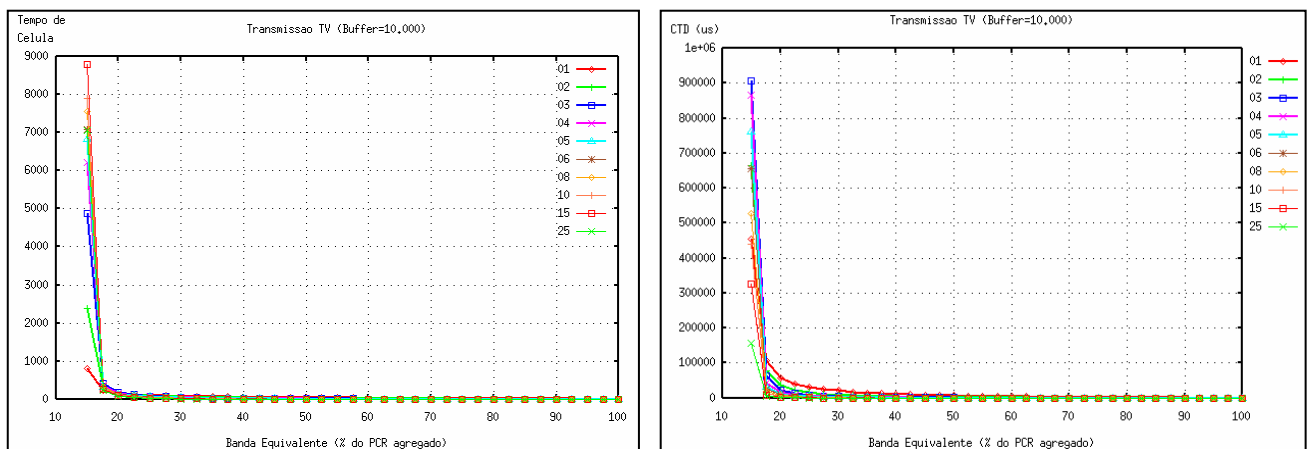
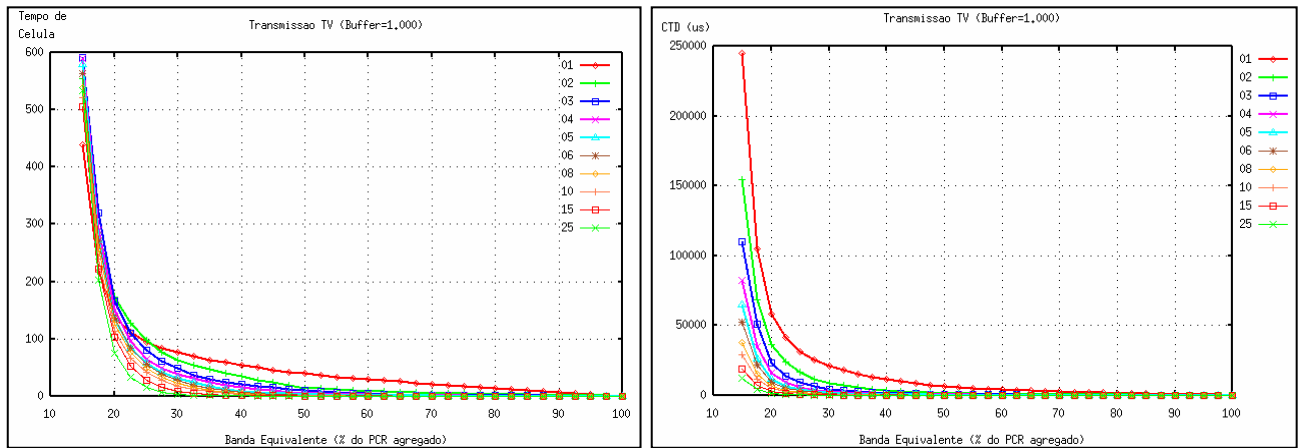
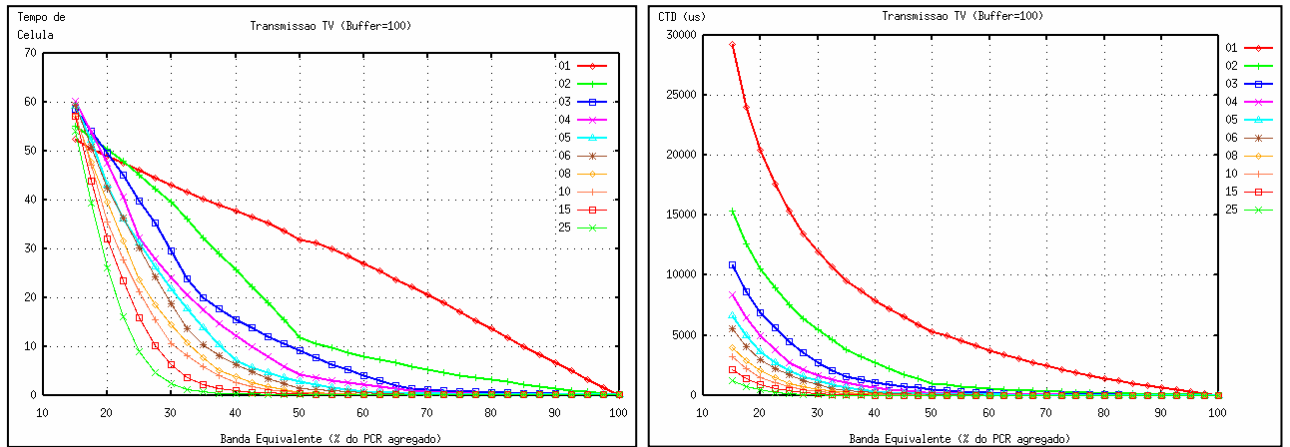


FIGURA 4.18 - CLR para Multiplexação de "*n*" conexões para transmissão de TV com *buffer* de 100 células. À direita, gráfico ampliado mostrando CLR próximo a 10^{-4} .



Vídeo Conferência

As figuras 4.22 e 4.23 mostram o desempenho de CTD e CDV para o serviço de vídeo conferência (tab. 4.6) com valores inferiores a 500 μ s. A figura 4.24 mostra a razão de perda de células (CLR) para esse serviço com *buffer* de 100 células no comutador.

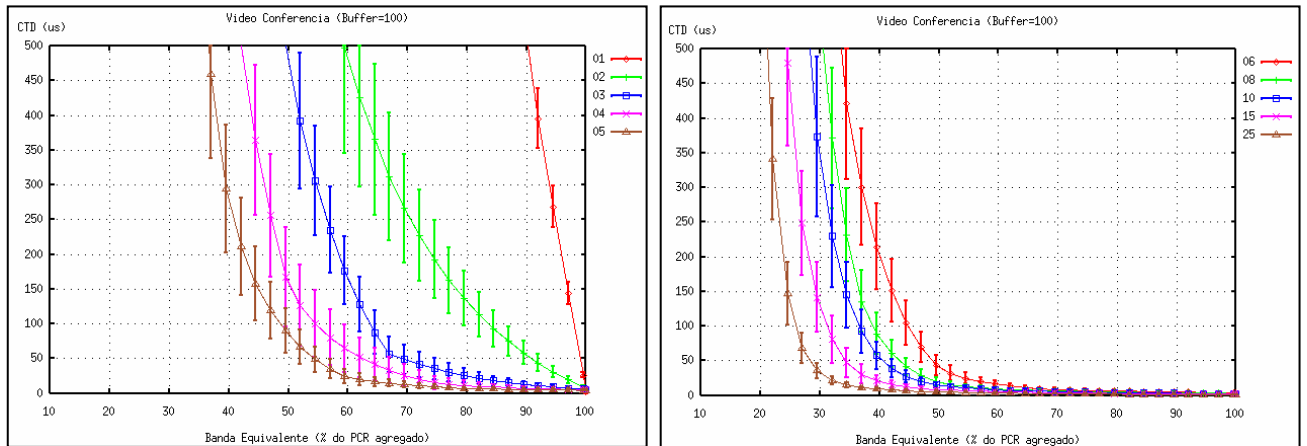


FIGURA 4.22 - CTD para Multiplexação de "*n*" conexões para vídeo conferência com *buffer* de 100 células.

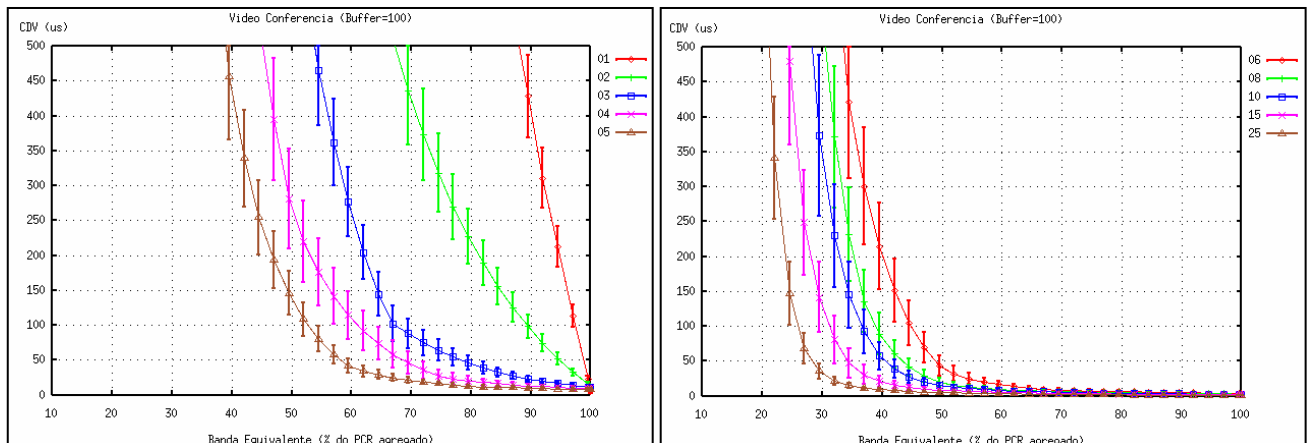


FIGURA 4.23 - CDV para Multiplexação de "*n*" conexões para vídeo conferência com *buffer* de 100 células

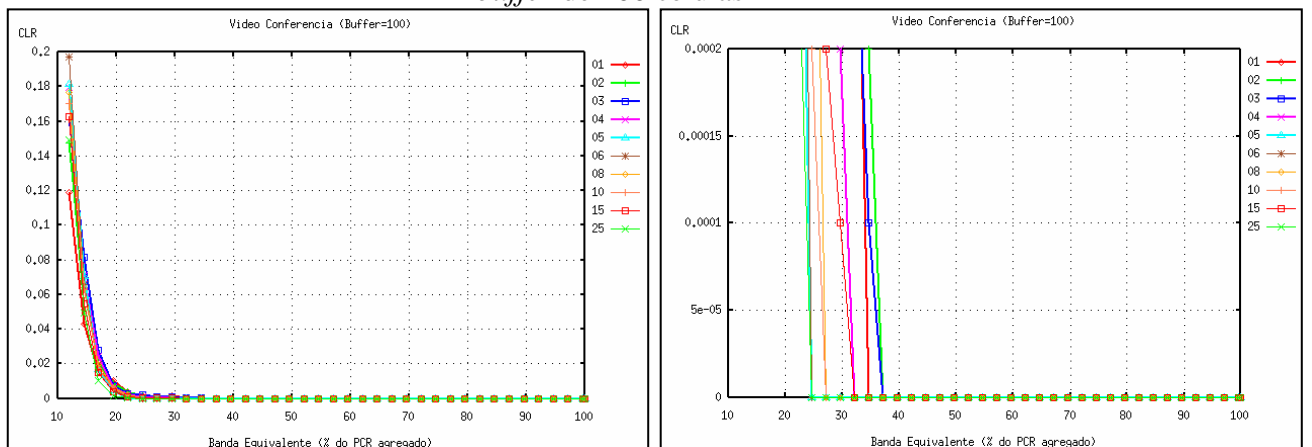


FIGURA 4.24 - CLR para Multiplexação de "*n*" conexões para vídeo conferência com *buffer* de 100 células. À direita, gráfico ampliado mostrando CLR próximo a 10^{-4} .

As figuras 4.25, 4.26 e 4.27 mostram o intervalo completo de CTD obtido com 100, 1.000 e 10.000 células de *buffer*.

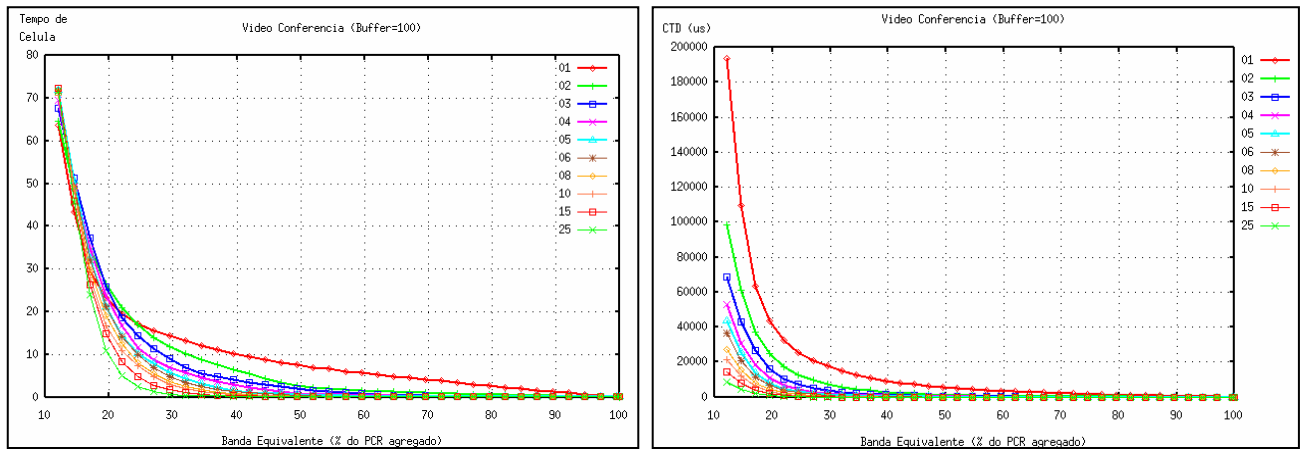


FIGURA 4.25 - CTD para Multiplexação de " n " conexões para vídeo conferência com *buffer* de 100 células. Por Tempo de Célula (esquerda) e por unidade de tempo (direita).

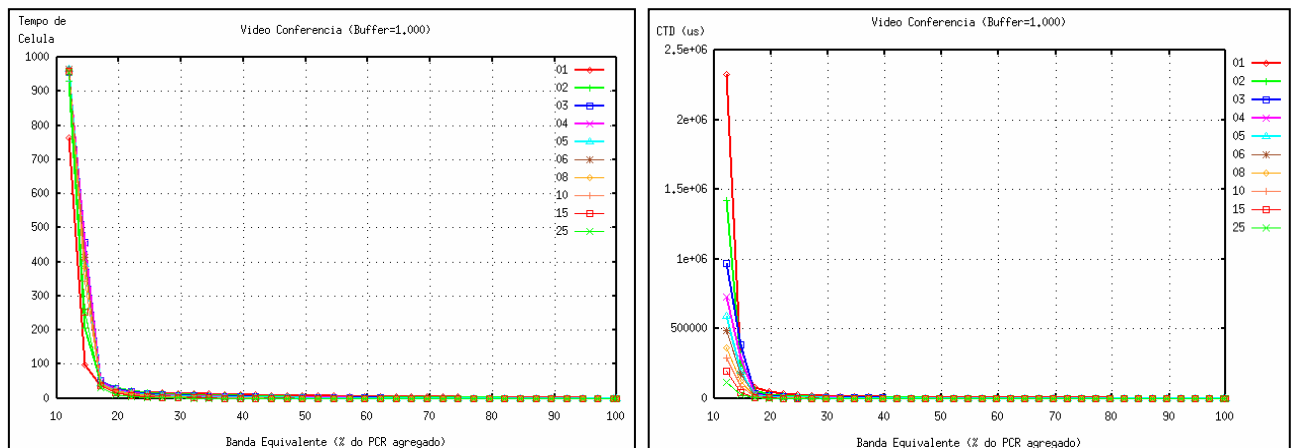


FIGURA 4.26 - CTD para Multiplexação de " n " conexões para vídeo conferência com *buffer* de 1.000 células. Por Tempo de Célula (esquerda) e por unidade de tempo (direita).

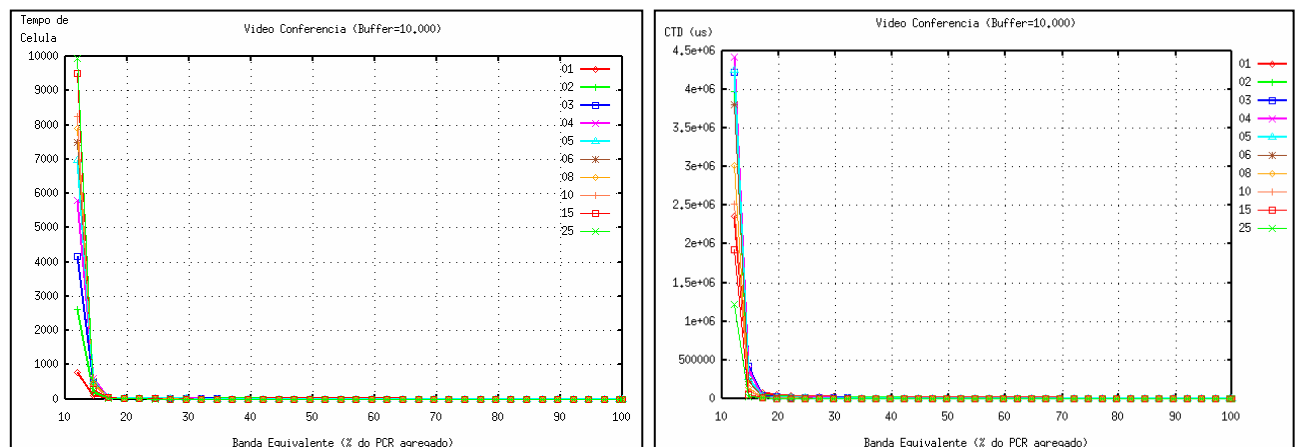


FIGURA 4.27 - CTD para Multiplexação de " n " conexões para vídeo conferência com *buffer* de 10.000 células. Por Tempo de Célula (esquerda) e por unidade de tempo (direita).

Vídeo Sob Demanda

As figuras 4.28 e 4.29 mostram o desempenho de CTD e CDV para o serviço de vídeo sob demanda (tab. 4.6) com valores inferiores a $500 \mu\text{s}$. A figura 4.30 mostra a razão de perda de células (CLR) para esse serviço com *buffer* de 100 células no comutador.

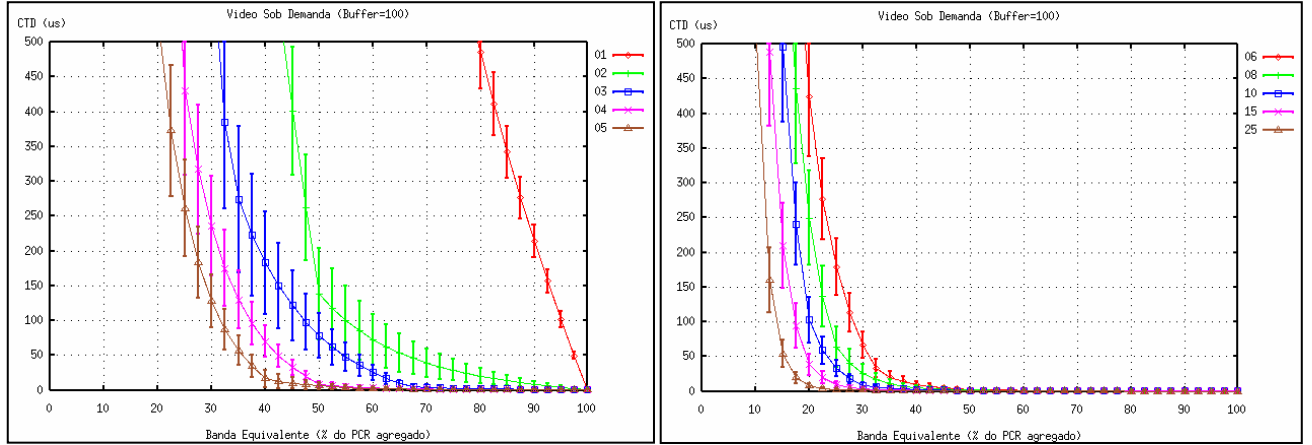


FIGURA 4.28 - CTD para Multiplexação de "*n*" conexões para vídeo sob demanda com *buffer* de 100 células.

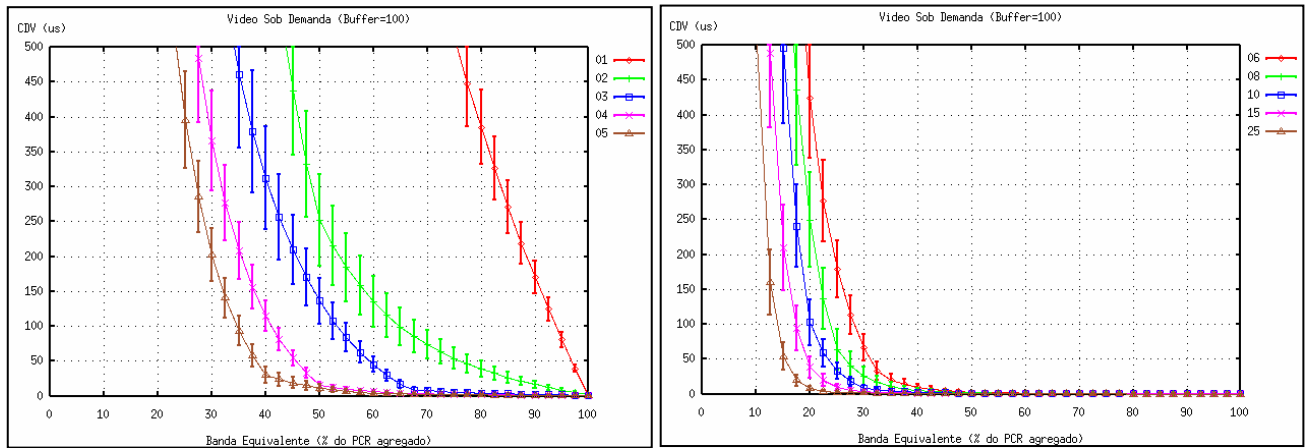


FIGURA 4.29 - CDV para Multiplexação de "*n*" conexões para vídeo sob demanda com *buffer* de 100 células

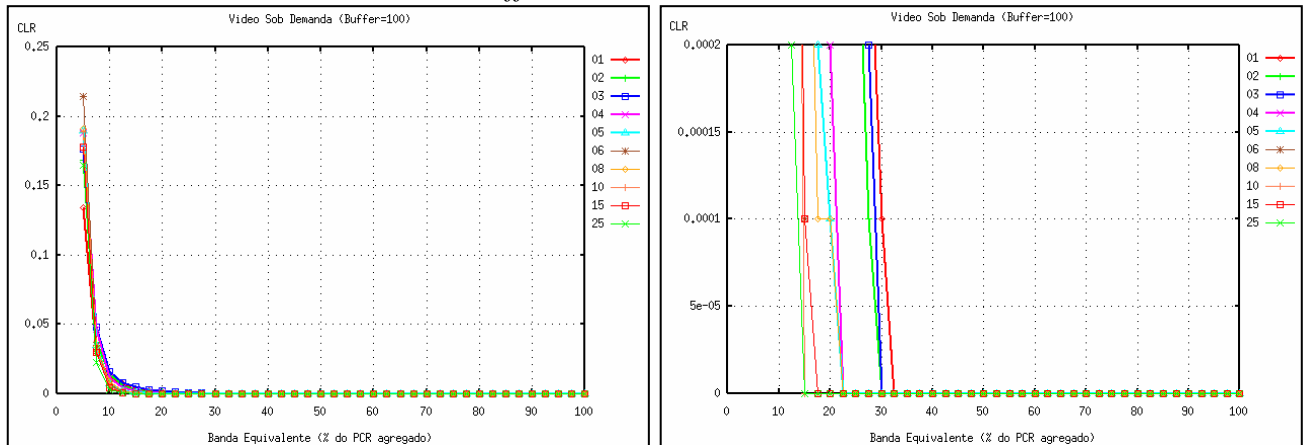


FIGURA 4.30 - CLR para Multiplexação de "*n*" conexões para vídeo sob demanda com *buffer* de 100 células. À direita, gráfico ampliado mostrando CLR próximo a 10^{-4} .

As figuras 4.31, 4.32 e 4.33 mostram o intervalo completo de CTD obtido com 100, 1.000 e 10.000 células de *buffer*.

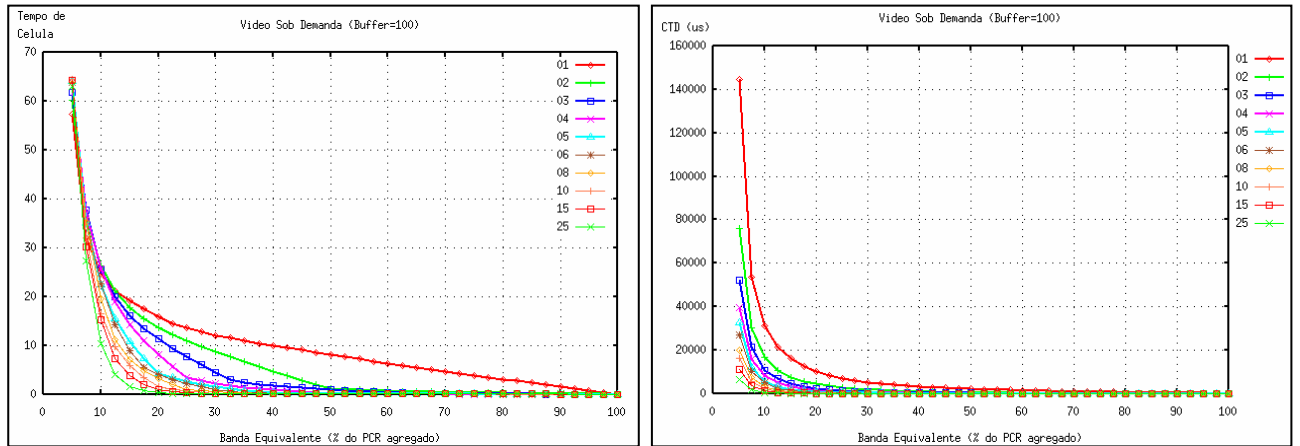


FIGURA 4.31 - CTD para Multiplexação de " n " conexões para vídeo sob demanda com *buffer* de 100 células. Por Tempo de Célula (esquerda) e por unidade de tempo (direita).

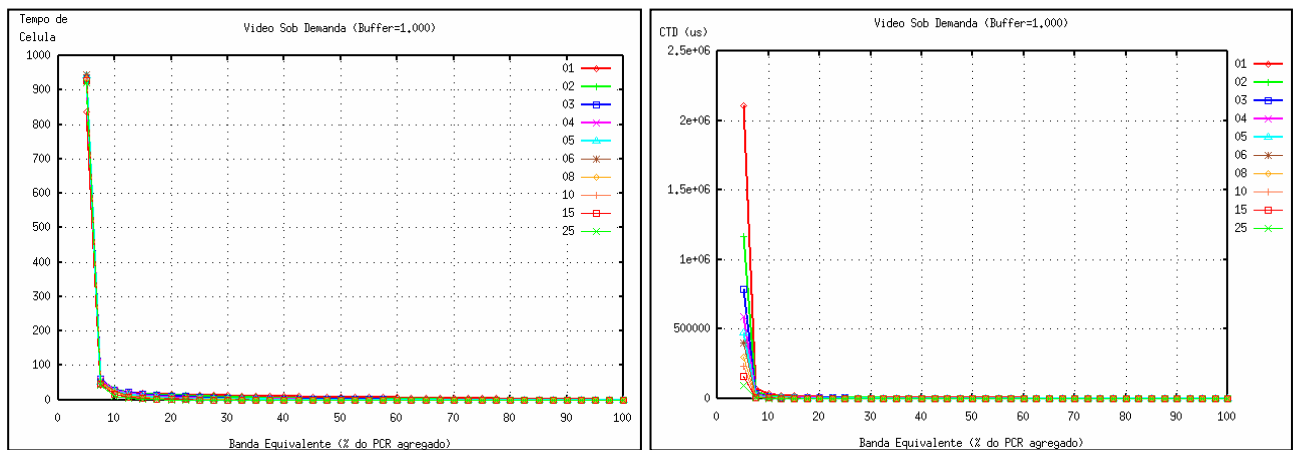


FIGURA 4.32 - CTD para Multiplexação de " n " conexões para vídeo sob demanda com *buffer* de 1.000 células. Por Tempo de Célula (esquerda) e por unidade de tempo (direita).

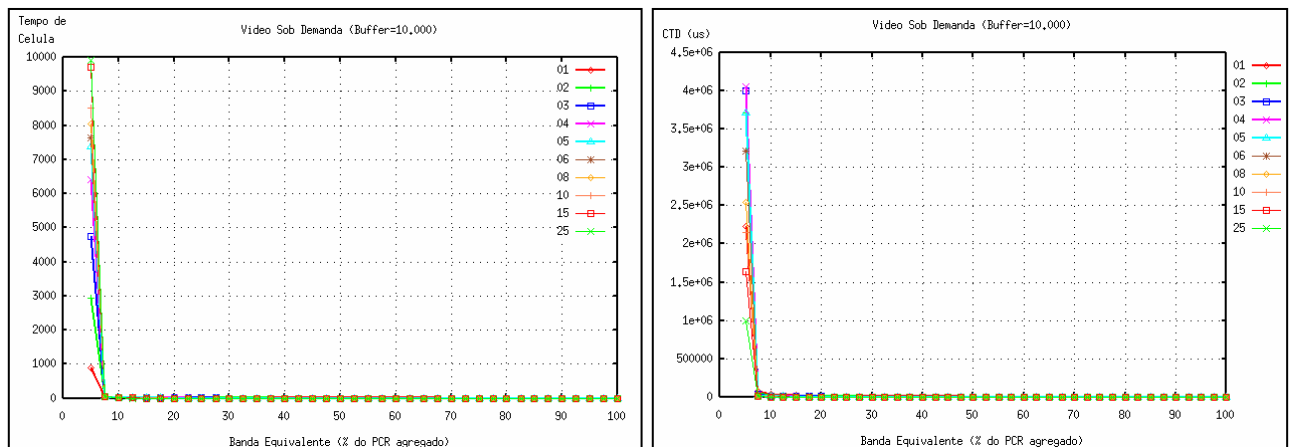


FIGURA 4.33 - CTD para Multiplexação de " n " conexões para vídeo sob demanda com *buffer* de 10.000 células. Por Tempo de Célula (esquerda) e por unidade de tempo (direita).

4.5 Dimensionamento de Capacidade conforme Resultados Obtidos

Os valores de CTD e de CDV requeridos variam de uma aplicação para outra (tab. 2.7). Por exemplo, uma vídeo conferência de 64 Kbit/s pode tolerar 300 ms de CTD, já uma vídeo conferência de 1,5 Mbit/s irá tolerar somente 5 ms de CTD de ponta a ponta da conexão. De forma geral, são utilizados os seguintes valores para transmissão de vídeo [GIN99]: CTD igual a 4 ms de ponta a ponta, limitados a 150 μ s por comutador ATM, CDV de 500 μ s de ponta a ponta e CLR de $1,7 \times 10^{-9}$. Conforme informações específicas do serviço a ser implementado sobre ATM, outros valores mais apropriados de requerimento de QoS podem ser utilizados para o dimensionamento dos recursos.

A tabela 4.7 mostra os valores de banda alocada necessários para garantir um valor de CTD médio de 150 μ s por porta de comutador para os serviços simulados. O valor de CDV acompanha o valor do CTD visto que o CDV de ponta a ponta é a diferença entre o menor CTD e o maior CTD da conexão. Por essa razão utilizaremos o parâmetro de CTD para determinar a alocação de recursos para a porta do comutador ATM. O valor de CLR utilizado será zero, ou seja, sem perda. Apesar de considerarmos o CLR, o CTD é o parâmetro mais restrigente²² para os serviços simulados, conforme mostra a tabela de alocação de recursos (tab. 4.7). A tabela de alocação de recursos foi montada a partir dos gráficos: de CTD (como a fig. 4.16) considerando valores de Intervalo de Confiança abaixo do limite de 150 μ s; e de CLR (como a fig. 4.18) considerando valores iguais ou inferiores a 10^{-4} . Lembramos que os gráficos utilizam incremento de 2,5 no eixo de coordenada horizontal (Banda Equivalente) para estabelecer cada ponto.

TABELA 4.7 - Alocação de Recursos para Serviço de **Transmissão TV** com *Buffer=100*

Conexões Agregadas	PCR Agregado (bit/s)	Banda Equivalente (% PCR Agregado)		Banda Alocada (bit/s)	Ganho de Multiplexação Estatística
		CTD<150 μ s	CLR $\leq 10^{-4}$		
1	5.073.585	97,5	80,0	4.946.745,38	1,03
2	10.147.170	85,0	77,5	8.625.094,50	1,18
3	15.220.755	65,0	65,0	9.893.490,75	1,54
4	20.294.340	57,5	62,5	12.683.962,50	1,60
5	25.367.925	50,0	57,5	14.586.556,88	1,74
6	30.441.510	45,0	55,0	16.742.830,50	1,82
8	40.588.680	40,0	45,0	18.264.906,00	2,22
10	50.735.850	37,5	42,5	21.562.736,25	2,35
15	76.103.775	30,0	42,5	32.344.104,38	2,35
25	126.839.625	25,0	35,0	44.393.868,75	2,86

²² Requisito restritivo pois exige maior alocação de recursos para ser atendido plenamente. Os requisitos menos restritivos (que exigem menos recursos) acabam sendo atendidos por consequência das exigências dos requisitos mais restritivos.

TABELA 4.8 - Alocação de Recursos para Serviço de **Transmissão TV** com *Buffer=1.000 e 10.000*

Conexões Agregadas	PCR Agregado (bit/s)	Banda Equivalente (% PCR Agregado)			Banda Alocada (bit/s)	Ganho de Multiplexação Estatística
		CTD<150µs	CLR≤10 ⁻⁴			
			Bf=1.000	Bf=10.000		
1	5.073.585	100,0	20,0	15,0	5.073.585,00	1,00
2	10.147.170	85,0	20,0	15,0	8.625.094,50	1,18
3	15.220.755	65,0	20,0	17,5	9.893.490,75	1,54
4	20.294.340	57,5	20,0	17,5	11.669.245,50	1,74
5	25.367.925	50,0	20,0	17,5	12.683.962,50	2,00
6	30.441.510	47,5	22,5	17,5	14.459.717,25	2,11
8	40.588.680	40,0	22,5	17,5	16.235.472,00	2,50
10	50.735.850	37,5	22,5	17,5	19.025.943,75	2,67
15	76.103.775	32,5	22,5	17,5	24.733.726,88	3,08
25	126.839.625	27,5	22,5	17,5	34.880.896,88	3,64

Observando os resultados mostrados na tabela 4.7 e 4.8, para a multiplexação de 1 a 3 conexões de transmissão de TV, obtemos o melhor ganho de multiplexação estatística utilizando o *buffer* de tamanho igual a 100 células. Para a multiplexação de 4 a 25 conexões, caso haja disponibilidade de *buffer*, obtemos o maior ganho de multiplexação estatística utilizando o *buffer* de 1.000 células. A utilização do *buffer* de 10.000 células se mostrou desnecessária para esse tipo de serviço.

TABELA 4.9 - Alocação de Recursos para Serviço de **Vídeo Conferência** com *Buffer=100, 1.000 e 10.000*

Conexões Agregadas	PCR Agregado (bit/s)	Banda Equivalente (% PCR Agregado)				Banda Alocada (bit/s)	Ganho de Multiplexação Estatística
		CTD<150µs	CLR≤10 ⁻⁴				
			Bf=100	Bf=1.000	Bf=10.000		
1	1.160.967,50	100,0	35,0	15,0	12,5	1.160.967,50	1,00
2	2.321.935,00	82,5	37,5	15,0	12,5	1.915.596,38	1,21
3	3.482.902,50	65,0	35,0	17,5	15,0	2.263.886,63	1,54
4	4.643.870,00	55,0	32,0	17,5	15,0	2.554.128,50	1,82
5	5.804.837,50	50,0	25,0	17,5	15,0	2.902.418,75	2,00
6	6.965.805,00	45,0	25,0	17,5	15,0	3.134.612,25	2,22
8	9.287.740,00	40,0	27,5	17,5	15,0	3.715.096,00	2,50
10	11.609.675,00	37,5	27,5	17,5	15,0	4.353.628,13	2,67
15	17.414.512,50	32,5	30,0	17,5	15,0	5.659.716,56	3,08
25	29.024.187,50	27,5	25,0	17,5	15,0	7.981.651,56	3,64

Observando os resultados mostrados na tabela 4.9, para a multiplexação de 1 a 25 conexões de vídeo conferência, obtemos o melhor ganho de multiplexação estatística utilizando o *buffer* de tamanho igual a 100 células. A utilização do *buffer* de 1.000 e 10.000 células se mostrou desnecessária para esse tipo de serviço.

TABELA 4.10 - Alocação de Recursos para Serviço de **Vídeo Sob Demanda**
com *Buffer*=100, 1.000 e 10.000

Conexões Agregadas	PCR Agregado (bit/s)	Banda Equivalente (% PCR Agregado)				Banda Alocada (bit/s)	Ganho de Multiplexação Estatística
		CTD<150 μ s	CLR $\leq 10^{-4}$				
			Bf=100	Bf=1.000	Bf=10.000		
1	3.365.520,00	95,0	30,0	7,5	7,5	3.197.244,00	1,05
2	6.731.040,00	55,0	30,0	7,5	7,5	3.702.072,00	1,82
3	10.096.560,00	47,5	30,0	7,5	7,5	4.795.866,00	2,11
4	13.462.080,00	37,5	22,5	7,5	7,5	5.048.280,00	2,67
5	16.827.600,00	32,5	20,0	7,5	7,5	5.468.970,00	3,08
6	20.193.120,00	27,5	20,0	7,5	7,5	5.553.108,00	3,64
8	26.924.160,00	25,0	20,0	7,5	7,5	6.731.040,00	4,00
10	33.655.200,00	20,0	15,0	7,5	7,5	6.731.040,00	5,00
15	50.482.800,00	17,5	15,0	7,5	7,5	8.834.490,00	5,71
25	84.138.000,00	15,0	15,0	7,5	7,5	12.620.700,00	6,67

Observando os resultados mostrados na tabela 4.10, para a multiplexação de 1 a 25 conexões de vídeo sob demanda, obtemos o melhor ganho de multiplexação estatística utilizando o *buffer* de tamanho igual a 100 células. A utilização do *buffer* de 1.000 e 10.000 células se mostrou desnecessária para esse tipo de serviço.

4.6 Validação dos Resultados

Para validar os resultados obtidos nas simulações utilizaremos a comparação com o trabalho desenvolvido em [RAM98] onde foram realizadas simulações com o produto SMURPH, ferramenta desenvolvida na Universidade de Alberta (Canadá).

Escolhemos a fonte de tráfego simulado no trabalho de [RAM98] que possui as características de transmissão de vídeo com tempo médio de transmissão da rajada (*On*) igual a 25.000 μ s, tempo médio entre rajadas (*Off*) igual a 35.000, taxa de transmissão máxima (PCR) igual a 6 Mbit/s e *buffer* para 1.000 células. Os resultados obtidos para CTD e CLR em [RAM98] são mostrados na figura 4.34. Os resultados obtidos utilizando o NIST ATM/HFC para mesma configuração de tráfego são mostrados na figura 4.35.

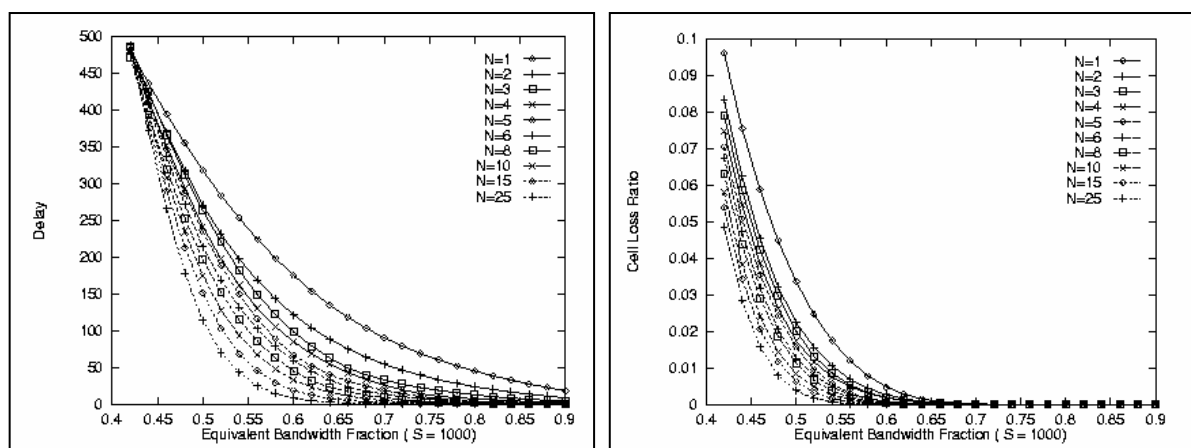


FIGURA 4.34 - CTD e CLR obtidos com o simulador SMURPH. Fonte:[RAM98]

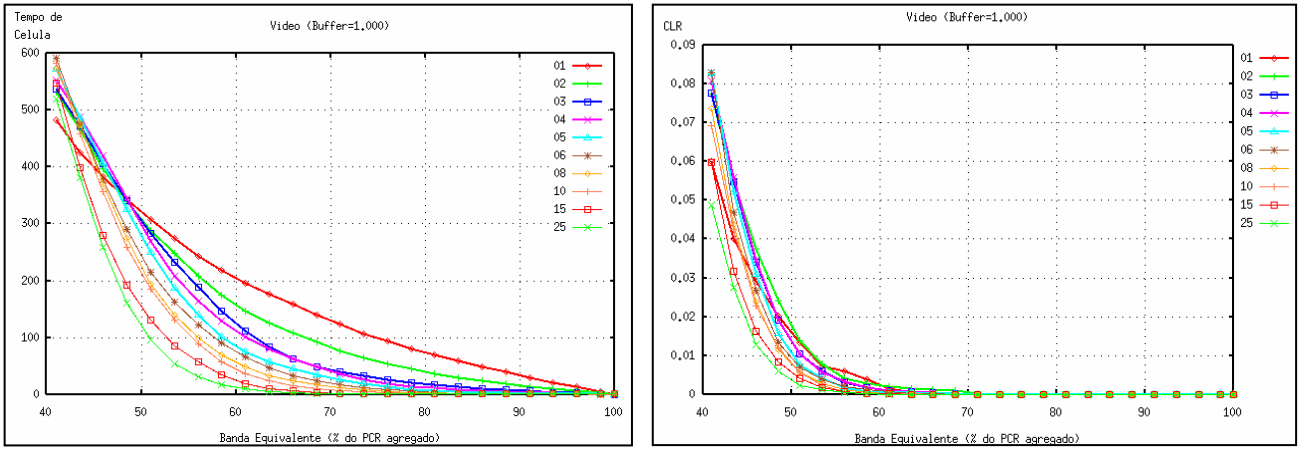


FIGURA 4.35 - CTD e CLR obtidos com o simulador NIST ATM/HFC.

Os resultados obtidos em nosso trabalho estão muito próximos em escala dos realizados em [RAM98].

5 Conclusões e Trabalhos Futuros

5.1 Conclusões

É uma tendência mundial crescente a utilização da tecnologia ATM na implantação de redes metropolitanas de alta velocidade para o suporte integrado de serviços como telefonia, transmissão de vídeo e acesso à base de dados. Porém, atualmente, são poucos os estudos sobre o dimensionamento dessas redes. Nosso trabalho vem ocupar esse espaço propondo um método para o planejamento de capacidade de redes ATM.

A utilização do nosso método para modelagem e simulação do comportamento de determinada configuração de rede sobre os efeitos de cargas de diferentes tráfegos é um grande facilitador tanto no processo de projeto quanto no de manutenção de redes. Por meio das simulações de inúmeras possibilidades de características da rede ou de tráfego podemos avaliar e comparar o desempenho das alternativas. Os resultados obtidos com as simulações podem, por exemplo, fornecer argumentos para ajustar a configuração da rede, prever alterações de desempenho, identificar pontos de saturação e congestionamento, relacionar dispositivos críticos a falhas e avaliar o impacto de ingresso de novos serviços. Recomendamos o uso do método para projetistas e administradores de redes, de forma que abandonem o processo de avaliação baseado em projeções empíricas e passem a utilizar uma ferramenta analítica baseada em probabilidades e estatísticas que, apesar de limitado em relação ao mundo real, pois abstrai características e as simplifica, modela com propriedade uma tendência de comportamento da rede. Diferentes configurações de redes e cargas de serviços podem ser exercitadas para revelar dificuldades de desempenho. O dimensionamento apropriado beneficia tanto o usuário quanto o provedor da rede. Entre os benefícios podemos citar:

- utilizar a infra-estrutura de rede de forma eficiente;
- dividir seus recursos por grande número de usuários;
- evitar congestionamento e degradação de serviços;
- atender às diferentes necessidades do usuário com custo atrativo.

O método de dimensionamento proposto neste trabalho é uma combinação de técnicas analíticas e simulação para determinar os recursos de uma rede ATM conforme a carga de serviço. A técnica empregada depende da categoria de serviço da conexão a ser estabelecida. Os recursos utilizados pelos serviços CBR, ABR e UBR foram estimados neste trabalho por equações matemáticas. Os serviços VBR, devido ao fluxo de rajadas e requisitos de QoS, utilizaram a simulação para determinar a banda equivalente. Tanto numa técnica como noutra a premissa é estimar os recursos tendo base a situação pessimista de tráfego, ou seja, um dimensionamento para o pior caso de condição de tráfego. Vejamos exemplos do porquê:

- no cálculo do *buffer* utilizado pelas conexões CBR, o MBS é o tamanho da rajada para o pior caso de CDVT. Parte-se do pressuposto que as conexões terão continuamente esse valor de CDVT, o que na realidade não acontece;

- a alocação de banda equivalente para as conexões VBR parte do mínimo necessário no enlace de transmissão. Consideramos aqui que o resto da banda do enlace já estaria comprometido para outras conexões. Na verdade o enlace utiliza para cada conexão a sua taxa disponível de transmissão;
- consideramos como se todas as conexões estivessem estabelecidas simultaneamente. O que na realidade ocorre é uma variação na quantidade de conexões estabelecidas;
- o agrupamento das conexões para o dimensionamento é feito considerando as conexões que são homogêneas. Estamos assumindo como se não houvesse o efeito de multiplexação entre fontes heterogêneas. Se considerássemos também a multiplexação entre os diferentes grupos de conexões, poderíamos ter um ganho estatístico maior propiciando a redução de alocação de recursos.

Essas premissas são necessárias visto que podem ocorrer e devem ser atendidas pela rede, em termos de recursos disponíveis, para garantir o QoS das conexões. Por outro lado, podem ser vistas como causadoras de estimacões exageradas de recursos utilizados. Considerando esses fatos, concluímos que o planejamento de capacidade garante com razoável precisão a quantidade mínima de serviço a ser suportada pela rede, porém reconhecemos que esse valor é pessimista em relação à possibilidade real de atendimento da rede.

Dentre as dificuldades que encontramos para empregar o método de dimensionamento, a maior, sem dúvida, foi a de caracterização dos serviços através dos descritores de tráfego. Muitos fornecedores de aplicações multimídia não informam as características dos fluxos gerados por desconhecimento ou desinteresse comercial. Acreditamos que com a disseminação da utilização de redes de banda larga com garantia de QoS, principalmente para o acesso de usuários residenciais, o interesse dos fornecedores em dispor tais informações aumente. De outra forma, continuaremos a contar com observações e medições de tráfego para a caracterização dos serviços em termos de taxa de pico de transmissão, taxa média, tamanho médio de rajada e outros.

Como produto resultante de nosso trabalho, a nova versão do NIST ATM/HFC pode ser usada como ferramenta básica para o planejamento de capacidade de redes ATM. Apesar de termos utilizado o simulador somente para a multiplexação de conexões VBR, o NIST ATM/HFC pode ser utilizado também para simular o comportamento da multiplexação de conexões CBR, ABR e UBR. Da mesma forma, independentemente de termos optado por trabalhar com fluxos de conexões homogêneas por motivos de simplificação do modelo, o NIST ATM/HFC suporta a multiplexação de conexões heterogêneas (diferentes descritores de tráfego e/ou diferentes categorias de serviço) passando por um ou mais de um comutador ATM. Lembramos que a complexidade do ambiente simulado aumenta o tempo necessário para a execução da simulação, porém esse é um inconveniente que está sendo minimizado com a disponibilização de computadores cada vez mais poderosos.

5.2 Trabalhos Futuros

Vários métodos têm sido propostos para a diminuição do tempo de processamento requerido em simulações. Dentre essas técnicas, segundo [TOW98], amostragem por importância tem se mostrado muito atrativa para a simulação rápida de vários sistemas, incluindo canais de comunicação, sistemas de fila e sistemas de confiabilidade. Dependendo do sistema, obtém-se uma redução de várias ordens no número de amostras necessárias por simulação para se obter uma estimativa confiável. A simulação com amostragem por importância aparece como um método alternativo para o dimensionamento de multiplexadores estatísticos ATM, procurando contornar os problemas apresentados pela simulação convencional e pelos métodos analíticos. Como trabalho futuro sugerimos a implementação no NIST ATM/HFC de técnicas de redução de variância do tipo amostragem por importância.

Em nosso trabalho fizemos adequações no gerador de tráfego VBR Poisson do simulador para a obtenção do CTD e CDV de cada célula transferida. Através de pequenas alterações (detalhes no anexo 3) outros geradores de tráfego disponíveis no NIST ATM/HFC podem também ser utilizados para a caracterização de tráfego no planejamento de capacidade. Da mesma forma, previsões de desempenho de tráfego ABR e UBR podem ser realizados utilizando o gerador de tráfego TCP/IP já existente no NIST ATM/HFC. Somente considerando o serviço de dados, há muito trabalho a ser aprofundado nas questões de oferecer qualidade de serviço.

Estudos de novos modelos de tráfegos, mais adequados para a simulação de serviços multimídia, podem ser implementados e avaliados no NIST ATM/HFC. Devido a sua construção em módulos (componentes) é relativamente fácil inserir novos componentes.

Como havíamos mencionado, o *script* "Banda_Equivalente" utilizado para escalonar as simulações de cenários de tráfegos homogêneos pode ser aprimorado para possibilitar o arranjo de fluxos heterogêneos, possibilitando a combinação de diferentes serviços.

Outra sugestão de trabalho futuro é desenvolver um programa que automatize, a partir dos dados obtidos nas simulações, o dimensionamento dos recursos da rede ATM. O armazenamento contínuo dos resultados obtidos nas simulações criaria uma base de dados que permitiria através de interpolação o dimensionamento de recursos para serviços com características similares aos encontrados na base, economizando, dessa forma, tempo de execução do simulador.

Anexo 1 Parâmetros de QoS Genéricos

Este anexo provê os objetivos de QoS para BSS (cross-broadband switching systems) baseados nos requisitos genéricos do B-ISDN do Bellcore [BEL95]. Essas informações foram reunidas de [GUI98].

Os parâmetros de desempenho CLR, CTD e CDV do Bellcore são requisitos mínimos esperados para cada porta de transmissão dos comutadores ATM. Portanto não são valores que totalizam o QoS da conexão de ponta a ponta.

Os valores são definidos conforme a velocidade de transmissão do enlace (porta) e a classe de QoS da conexão.

TABELA Anexo 1.1 - Parâmetro de QoS para conexões em uma porta OC-3c e OC-12

Parâmetros de Desempenho	CLP	QoS da Conexão		
		Classe 1	Classe 3	Classe 4
CLR	0	$\leq 10^{-10}$	$\leq 10^{-7}$	$\leq 10^{-7}$
CLR	1	n/s ²³	N/s	n/s
CTD (99 percentile)	0/1	150 μ s	150 μ s	150 μ s
CDV (10^{-10} quantile)	0/1	250 μ s	N/s	n/s
CDV (10^{-7} quantile)	0/1	n/s	250 μ s	250 μ s

TABELA Anexo 1.2 - Parâmetro de QoS para conexões em uma porta DS-3²⁴

Parâmetros de Desempenho	CLP	QoS da Conexão		
		Classe 1	Classe 3	Classe 4
CLR	0	$\leq 10^{-10}$	$\leq 10^{-7}$	$\leq 10^{-7}$
CLR	1	n/s	N/s	n/s
CTD (99 percentile)	0/1	500 μ s	500 μ s	500 μ s
CDV (10^{-10} quantile)	0/1	500 μ s	N/s	n/s
CDV (10^{-7} quantile)	0/1	n/s	500 μ s	500 μ s

TABELA Anexo 1.3 - Parâmetro de QoS para conexões em uma porta DS-1²⁵

Parâmetros de Desempenho	CLP	QoS da Conexão		
		Classe 1	Classe 3	Classe 4
CLR	0	$\leq 10^{-10}$	$\leq 10^{-7}$	$\leq 10^{-7}$
CLR	1	n/s	N/s	n/s
CTD (99 percentile)	0/1	2000 μ s	2000 μ s	2000 μ s
CDV (10^{-10} quantile)	0/1	4000 μ s	N/s	n/s
CDV (10^{-7} quantile)	0/1	n/s	4000 μ s	4000 μ s

²³ Não especificado.

²⁴ DS3 é uma taxa de transmissão de 44,736 Mbit/s.

²⁵ DS1 é uma taxa de transmissão de 1,544 Mbit/s.

Anexo 2 Execução do Simulador

O simulador NIST ATM/HFC possui dois modos de execução: interface gráfica ou linha de comando (*background*). A execução da simulação via interface gráfica permite a configuração do cenário de rede a ser simulado, bem como acompanhar interativamente a situação da ocupação dos *buffers*, da taxa de transmissão dos enlaces, do número de células descartadas e de outros valores disponíveis. Para os dois modos podemos utilizar um arquivo texto de entrada para configuração do cenário de rede a ser simulado. A vantagem da execução via linha de comando (sem interface gráfica) é a rapidez na execução da simulação, chegando a ser vinte vezes mais rápido do que no modo gráfico. Os resultados da simulação para os dois modos são apresentados em um arquivo texto de saída, chamado geralmente de "SIM_LOG". No final da execução, também é gerado no formato texto um arquivo com a configuração do cenário de rede. Esse arquivo é chamado "SIM_SNAP" e tem como objetivo servir como entrada no simulador para configurar uma nova execução. Exemplo dos arquivos "SIM_LOG" e "SIM_SNAP" estão neste anexo.

O comando para execução do simulador e seus parâmetros são apresentados no exemplo a seguir:

```
sim -x -s 963360534 -o sim_log -int 500000 -w 8000000 conf_temp 48000000
```

Onde,

sim é o nome do programa executável do NIST ATM/HFC;
-x é o parâmetro utilizado para especificar o modo *background* de execução;
-s é o parâmetro para informar a semente para o gerador de números aleatórios;
-o é o parâmetro que informa o nome do arquivo de saída dos resultados (SIM_LOG);
-int é o parâmetro para informar o tamanho do subintervalo em microssegundos. Esse parâmetro não é original da versão do NIST e foi implementado em nosso trabalho;
-w é o parâmetro utilizado para informar o tamanho em microssegundos do tempo de instabilidade (*warmuptime*) que deverá ser aguardado após o início da simulação para começar a contabilização das variáveis observadas;
conf_temp é o nome do arquivo de configuração do cenário de rede a ser simulado. Contém todas as informações das características dos elementos que compõem a rede. Por exemplo, para o comutador temos: quantidade de portas, quantidade de *buffer* por porta, velocidade dos enlaces, *delay* interno de comutação, e outros;
48000000 é o valor em microssegundos para a duração do tempo simulado. Quando o simulador atinge esse valor de tempo simulado, ele encerra a execução gravando os arquivos de saída SIM_LOG e SIM_SNAP.

A versão original do NIST encerra a execução do simulador somente quando o tempo simulado informado na linha de comando é alcançado. Nós adicionamos no simulador a condição de também encerrar a simulação após a execução de no mínimo 30 subintervalos quando o intervalo de confiança do CTD tiver uma largura de borda²⁶ menor que 10%.

No final da execução da simulação, dois tipos de detalhes de informações (linha do arquivo de saída) foram implementados no arquivo SIM_LOG. Um tipo para mostrar os valores obtidos em cada subintervalo e outro, no final do arquivo, com o consolidado de todos os subintervalos. O formato das linhas de saída é apresentado a seguir.

Linha detalhe de Final de Subintervalo (SIM_LOG):

```
"# 1 'FTP_C"
"1 Tick      Num_Int
  CLP_Int    CTD_Int    CDV_Int    Cel_Rec    CLP_Med
  CTD_Med    CDV_Med    CTD_IC99(99%)  CDV_IC99(99%)"

1 853073 1 9.40 61.08 0.01 1000 5.30 61.08 0.01 0.00 0.00
A  B  C  D  E  F  G  H  I  J  K  L
```

Onde,

A = Identificador da linha detalhe. É igual para todas as linhas detalhes de final de subintervalo.

B = Instante em *ticks* (ns) da impressão da linha de detalhe do intervalo

C = Número do subintervalo

D = Razão de perda de células do subintervalo corrente (porcentagem)

E = Média do *delay* (CTD) do intervalo corrente

F = Média do *jitter* (CDV) do intervalo corrente

G = Número de células do intervalo

H = Média da razão de perda de células entre todos os subintervalos

I = Média do *delay* entre todos os intervalos

J = Média do *jitter* entre todos os intervalos

K = Valor para cálculo do intervalo de confiança do *delay*. [I-K,I+K] 99%

L = Valor para cálculo do intervalo de confiança do *jitter*. [J-L,J+L] 99%

Linha detalhe de Final de Simulação (SIM_LOG):

```
"999 Num_Cnx    Cel_Env    Cel_Rec    Num_Int
      Tam_Int(us)  Warmup(us)  CLR_Med    CTD_Med(us)
      CDV_Med(us)  CTD_IC99(us)  CDV_IC99(us)  T_Sim(s)  T_Exec(s)"

999 4 42454 38208 9 200000 8000000 10.00 94263.23 10.47 0.00 2.98 3 4
A  B  C  D  E  F  G  H  I  J  K  L  M  N
```

²⁶ Largura de borda é a distância do centro até as bordas do intervalo de confiança. 10 % de largura corresponde a um intervalo de (0,9 x CTDmédio) a (1,1 x CTDmédio).

Onde,

A = Identificação do parâmetro a ser gravado. É 999 para a linha detalhe de final de simulação;

B = Número de conexões;

C = O total de células enviadas por todas as conexões, as células do período de *warmuptime* não são contabilizadas;

D = O total de células recebidas por todas as conexões, as células do período de *warmuptime* não são contabilizadas;

E = Quantidade de subintervalos que a simulação foi subdividida;

F = Tamanho do subintervalo em microssegundos;

G = Tamanho do período inicial (*warmuptime*) da simulação aguardado para começar a contabilizar os subintervalos estatísticos;

H = Média da razão de perda de células da simulação considerando todas as conexões;

I = Média do CTD da simulação considerando todas as conexões;

J = Média do CDV da simulação considerando todas as conexões;

K = Intervalo de confiança de 99% para a média do CTD [I-K , I+K]

L = Intervalo de confiança de 99% para a média do CDV [J-L , J+L]

M = Tempo simulado em segundos;

N = Tempo de execução da simulação em segundos;

Exemplo de arquivo de saída SIM_LOG:

```
# 1 'REC1'
# '1 Tick Num_Int CLP_Int CTD_Int CDV_Int Cel_Rec CLP_Med CTD_Med CDV_Med CTD_
IConf(99%) CDV_IConf(99%)'
1 850002362 1 0.0000 40.66 0.00 1594 0.0000 40.66 0.00 0.00 0.00
1 900018657 2 0.0000 36.30 53.86 1553 0.0000 38.48 53.86 5.58 68.94
1 950025004 3 0.0000 40.41 54.42 1590 0.0000 39.13 54.14 3.62 46.20
1 1000029477 4 0.0000 37.44 54.90 1571 0.0000 38.71 54.39 2.78 34.82
1 1050039759 5 0.0000 39.62 54.26 1590 0.0000 38.89 54.36 2.20 27.84
1 1100058330 6 0.0000 40.66 56.39 1587 0.0000 39.19 54.77 1.95 23.39
1 1150076445 7 0.0000 40.35 56.85 1608 0.0000 39.35 55.11 1.70 20.18
1 1200085072 8 0.0000 39.16 55.14 1603 0.0000 39.33 55.12 1.48 17.66
1 1250090271 9 0.0000 42.19 56.58 1629 0.0000 39.65 55.30 1.53 15.76
1 1300092593 10 0.0000 39.61 55.84 1601 0.0000 39.64 55.36 1.37 14.20
1 1350112109 11 0.0000 39.27 54.74 1592 0.0000 39.61 55.30 1.24 12.89
1 1400118930 12 0.0000 39.37 55.10 1592 0.0000 39.59 55.28 1.14 11.81
1 1450134906 13 0.0000 39.79 55.49 1598 0.0000 39.60 55.30 1.05 10.91
1 1500154794 14 0.0000 41.50 56.53 1600 0.0000 39.74 55.39 1.03 10.15
1 1550155607 15 0.0000 38.55 56.19 1580 0.0000 39.66 55.45 0.98 9.48
1 1600168711 16 0.0000 39.61 55.06 1569 0.0000 39.66 55.42 0.92 8.89
1 1650169643 17 0.0000 38.49 54.90 1583 0.0000 39.59 55.39 0.88 8.36
1 1700189259 18 0.0000 39.71 54.88 1593 0.0000 39.60 55.36 0.83 7.89
1 1750203560 19 0.0000 40.63 56.01 1595 0.0000 39.65 55.40 0.80 7.48
1 1800208206 20 0.0000 38.95 55.71 1580 0.0000 39.61 55.41 0.76 7.11
1 1850214991 21 0.0000 38.20 54.28 1565 0.0000 39.55 55.36 0.74 6.76
1 1900234712 22 0.0000 41.04 55.63 1610 0.0000 39.62 55.37 0.73 6.46
1 1950253801 23 0.0000 42.46 58.02 1621 0.0000 39.74 55.49 0.77 6.20
1 2000274098 24 0.0000 40.57 57.60 1594 0.0000 39.77 55.58 0.74 5.95
1 2050293125 25 0.0000 38.72 54.89 1591 0.0000 39.73 55.55 0.72 5.71
1 2100300510 26 0.0000 40.78 54.89 1614 0.0000 39.77 55.53 0.70 5.49
1 2150305042 27 0.0000 38.45 55.14 1587 0.0000 39.72 55.51 0.68 5.29
1 2200362388 28 0.0000 40.55 54.66 1616 0.0000 39.75 55.48 0.66 5.10
1 2250370240 29 0.0000 41.09 56.71 1606 0.0000 39.80 55.52 0.65 4.92
1 2300370752 30 0.0000 40.28 56.49 1597 0.0000 39.81 55.56 0.63 4.76
1 2350382475 31 0.0000 38.17 54.77 1571 0.0000 39.76 55.53 0.62 4.61
# 999 'RESULTADO FINAL'
# '999 Num_Cnx | Cel_Env | Cel_Rec | Num_Int | Tam_Int(us) | Warmup(us) |
# ' CLR_Med(%) | CTD_Med(us) | CDV_Med(us) | CTD_IC99(us) | CDV_IC99(us)
| T_Sim(s) | T_Exec(s)'
999 1 49380 49380 31 500000 8000000 0.0000 39.76 55.53 0.62 4.61 24 3
```

Conforme descrito na seção **4.3 Método da Simulação**, para cada serviço são feitas simulações da multiplexação de fluxos homogêneos com diferentes alocações de recursos. As variações nas execuções das simulações são determinadas pelos fatores: número de conexões, taxa de transmissão²⁷ e tamanho do *buffer*. A tabela a seguir (tab. Anexo 2.1) mostra os valores adotados para esses fatores. Todos esses valores podem ser modificados através de parâmetros dos arquivos de execução (*scripts*²⁸) das simulações. O anexo 4 traz a relação dos *scripts* utilizados para execução e apresentação dos resultados. Diversas execuções do simulador são necessárias para montar os gráficos com os resultados de desempenho da multiplexação das conexões. Um *script* denominado *Banda_Equivalente* foi desenvolvido para gerar "n" arquivos de configuração, um para cada situação de alocação de recurso.

TABELA Anexo 2.1 - Fatores para combinação

Fator	Valor
Número de Conexões	1, 2, 3, 4, 5, 6, 8, 10, 15, 25
Taxa de Transmissão	De SCR/PCR até 1,0 com incremento de 0,025
Tamanho Buffer	100, 1.000 e 10.000

O script *Banda_Equivalente* tem duas finalidades:

- gerar os arquivos de configurações para multiplexação de "n" conexões;
- executar o simulador NIST ATM/HFC em modo de linha de comando, variando: o arquivo de configuração para "n" conexões e a taxa de transmissão da porta de saída do comutador

A arquivo de saída do *script* *Banda_Equivalente* é chamado de SAIDA_SIM. O conteúdo desse arquivo é a linha detalhe de cada final de execução do simulador (SIM_LOG) acrescida das seguintes informações identificadas com "*":

```
999 4 42454 38208 9 200000 1200000 10.00 94263.23 10.47 0.00 2.98 3 4 0.9 36 11.77 1000
A B C D E F G H I J K L M N O* P* Q* R*
```

onde,

A a N = Detalhe da linha final de cada simulação presente no arquivo SIM_LOG.

O = Porcentagem de Taxa Efetiva utilizada

P = Taxa Efetiva utilizada = B x O x PCR

Q = Tempo de Célula (us) = 424 / P

R = Tamanho do *buffer* do comutador em unidades de células

Exemplo do arquivo SAIDA_SIM é mostrado a seguir.

²⁷ Taxa de serviço de atendimento da fila onde há a multiplexação dos fluxos.

²⁸ Scripts são arquivos utilizados pelo interpretador de comando para execução de seqüência de comandos. O interpretador de comando utilizado foi o bshell do sistema operacional Linux.

SAIDA_SIM

```

<início>
999 1 223311 207533 79 500000 8000000 2.9980 83316.64 48067.03 9433.29 5324.49 4 8 9
0.41 2.46 172.358 1000
999 1 223311 212336 79 500000 8000000 2.0082 69149.65 46108.14 8640.28 5059.21 4 8 9
0.435 2.61 162.452 1000
...
...
999 25 5848642 5848642 79 500000 8000000 0.00 0.92 1.20 0.14 0.12 48 641 1 147 2.86971 1000
999 25 5848642 5848642 79 500000 8000000 0.00 0.89 1.16 0.14 0.11 48 664 1 150 2.82667 1000
#####
#                               Servico: Video Sob Demanda
#####
#                               Categoria: VBR
#                               PCR = 6 Mbit/s
#                               MBS = 25000 us
#                               MBSO= 35000 us
#####
#                               Cenario: 250 simulacoes
#                               Buffer:      [1000 celulas ]
#                               Conexoes:   [1,2,3,4,5,6,8,10,15,25] (10 ocorrencias)
#                               B. Efetiva: [41;100] (2.5%)          (25 ocorrencias)
#####
#                               Execucao: 462726932 celulas geradas
#                               Data e Hora Inicial: 02/11 11:27:51
#                               Data e Hora Final:   02/11 22:02:50
#                               Duracao:            10:34:59
#####
<fim>

```

Anexo 3 Relação dos Programas Alterados no NIST ATM/HFC

O simulador NIST ATM/HFC foi desenvolvido na linguagem C e está dividido em 99 módulos (programas fontes). Desse total, 16 módulos foram alterados para adequar o simulador a este trabalho. A tabela Anexo 3.1 lista as alterações feitas nos arquivos fontes.

TABELA Anexo 3.1 - Notas das Alterações Efetuadas

Nome do Arquivo Fonte	Nome da Função	Modificação e comentário
CLOCK.c	draw_digits()	- ARG_inc: Incl. de comentário na tela da interface gráfica : "Milliseconds arg"
XINIT.c	xinit()	- ARG_alt: Proporção da tela ambiente gráfico (8/10 - > 10/10)
EDIT.c	Define	- ARG_alt: Posição das janelas de escolha de componente para criar: - ARG_alt:EDIT_X de 500->450, EDIT_Y de 100->20 - ARG_alt:EDIT_X_C de 700->600, EDIT_Y_C de 190->40
LOG.c		
LOG.h		- ARG_com: Para compilar alterações nesse modulo, modificar data de fontes que o utilizem (ex. CBBTE.c)
	log_param_estatistica()	- ARG_inc: Adicionada função para imprimir DELAY
	log_resultado_final(num_conexoes, delay, jitter)	- ARG_inc: Adicionada função para imprimir resultados finais no arquivo log
CBRCONNECTI ON.h e VBRCONNECTI ON.h	Define	- ARG_inc: Inclui na estrutura CBR/VBR Connectiонт (tipo de componente) na posição que coincida com a definição Component (Component.h): Estatistica *cn_estatistica; A estrutura "Estatistica" está definida no component.h e é utilizada também na estrutura Component.
CBRCONNECTI ON.c e VBRCONNECTI ON.c	cn_send() CBR e cn_my_send_cell() VBR	- ARG_inc: timestamp de saída da célula na geração da célula: cell->stamp1=ev_now(); - Arg_inc: Marca na célula qual o componente que vai recebe-la (aplicação) no outro extremo da conexão. - ARG_inc: incrementa nas estatísticas o número de células enviadas.
	cn_reset()	- ARG_inc: reseta campos de estatísticas: Estat_reset(cn->cn_estatistica);
	cn_create()	- ARG_inc: inicialização do parâmetro Células Enviadas para este destino - ARG_inc: inicialização do parâmetro Células Recebidas - ARG_inc: inicialização do parâmetro Tamanho de Subintervalo - ARG_inc: inicialização do parâmetro Numero do Subintervalo - ARG_inc: inicialização do parâmetro Perda Media - ARG_inc: inicialização do parâmetro Delay Médio - ARG_inc: inicialização do parâmetro Jitter Médio - ARG_inc: inicialização do parâmetro Intervalo de Confiança do Jitter - ARG_inc: inicialização da área de estatística

Nome do Arquivo Fonte	Nome da Função	Modificação e comentário
	cn_receive()	- ARG_inc: Chama rotina para calcular estatísticas e depois imprime. ARG_inc: Atualiza variáveis da tela e imprime estatísticas.
	cn_route()	- ARG_alt: correção do nome do bte do outro extremo. Troquei l_tail por l_head
	cn_send()	- ARG_inc: incrementa nas estatísticas o número de células enviadas
	cn_hops()	- ARG_alt: adicionei o teste de SWITCH_CLASS, para permitir conexão da aplicação direto no switch. Para diminuir quantidade de eventos gerados
	cn_neighbor()	- ARG_alt: adicionei no teste SWITCH_CLASS, para permitir conexão da aplicação direto no switch. Para diminuir quantidade de eventos gerados
SUBR.c	calcula_estatísticas()	- ARG_inc: nova rotina para calcular estatísticas a partir da célula recebida
	calcula_estatísticas_interv()	- ARG_inc: nova rotina para calcular estatísticas por intervalo de tempo
	estat_init()	- ARG_inc: Rotina para inicializar variáveis de Estatísticas. Aloca espaço para as variáveis e retorna o endereço da estrutura.
	estat_reset()	- ARG_inc: Rotina para reinicializar variáveis de Estatísticas
	contabiliza_descarte()	- ARG_inc: Rotina para contabilizar na aplicação recebedora o número de células descartadas pelo switch (transbordo de buffer).
	contabiliza_envio()	- ARG_inc: Rotina para contabilizar envio de células a partir da aplicação geradora de tráfego. Se o período de warmuptime não terminou, não contabiliza.
COMPONENT.h	Define	- ARG_inc: Estrutura para as estatísticas de CTD e CDV: typedef struct _Estatistica
		- ARG_inc: estatísticas CTD, CDV e CLR no campo Component: Estatistica *co_estatistica;
SWITCH.c	switch_action()	- ARG_alt: o parâmetro sw_speed (vel. comutação) para aceitar double (de u.i para u.d). ARG_alt: o parâmetro sw_speed para aceitar double (de int para double). - ARG_inc: Faz a comutação da célula sem esperar pelo próximo slot time do switch - dessa forma elimina-se o delay da comutação. Permanece o delay interno que e' informado pelo usuário na configuração do switch.
	contabiliza_descarte()	- ARG_inc: Rotina para calculo de perda, contabilizando o descarte de células por conexão.
SIM.h	Define	- ARG_inc: define a quantidade de Subintervalos máxima (subdivisões) para cada simulação: = 10000. - ARG_inc: define o numero da versão do simulador para ser impresso no arquivo SIM_SNAP.
FILE.c	save_whatever()	- ARG_inc: imprime versão do NIST ATM. - ARG_alt: passa a impressão do tempo do snapshot de %u para %1.0f (double). - ARG_inc: impressão do tempo real decorrido.
	save_world()	- ARG_inc: calculo do tempo decorrido
	save_snapshot()	- ARG_inc: calculo do tempo decorrido
	hhmmss()	- ARG_inc: rotina para transformar de segundos para HH:MM:SS

Nome do Arquivo Fonte	Nome da Função	Modificação e comentário
MAIN.c	Define Main()	<ul style="list-style-type: none"> - ARG_inc: variáveis para cálculo do tempo decorrido: <pre>time_t tempo_de_inicio, tempo_final; double tempo_real;</pre> - Antes de começar a simulação, guarda o tempo corrente. Utiliza a definição <time.h> - ARG_inc: Novo argumento de entrada na chamada do simulador para valor do subintervalo em unidades de microssegundos; - ARG_inc: Inicialização do valor do subintervalo caso o campo de entrada não seja utilizado <pre>tick_t tam_interv = USECS_TO_TICKS(500);</pre> - ARG_inc: início do primeiro intervalo após o término do período de warmuptime - ARG_inc: Durante a simulação, acompanha o tempo simulado dividindo em subintervalos para executar as estatísticas. - ARG_inc: calculo do tempo decorrido na chamada da impressão do arquivo SIM_SNAP (final da simulação = stoptime). - ARG_inc: imprime resultados finais como última linha do arquivo SIM_LOG. No final da simulação, imprime os resultados das medias entre todas as conexões. - ARG_inc: Coloca o tempo simulado e o tempo real decorrido no arquivo SIM_LOG. - ARG_inc: Condicionar final da simulação pelo intervalo de confiança do Delay.
	sim_resultado()	- ARG_inc: rotina para imprimir resultado final da Simulação
	coloca_valor_intervalo()	- ARG_inc: coloca no display da tela de todos as aplicações o valor do intervalo para a execução da simulação. Só para constar no display.
	verifica_intervalo_confianca()	- ARG_inc: rotina para verificar se o intervalo de confiança atingiu final da Simulação.
CONTROLS.c	control_button_press_handler()	- ARG_inc: cálculo do tempo real decorrido.
Makefile	-x-	- ARG_alt: rbswitch.(cho) --> switch.(cho) e rbbte.(cho) --> bte.(cho)

Anexo 4 *Scripts* e Arquivos utilizados na Simulação

Arquivos de comandos conhecidos por *scripts* foram desenvolvidos para as seguintes finalidades:

- gerar dinamicamente os arquivos de configuração das simulações;
- escalonar a chamada da execução de cada simulação;
- extrair os dados dos arquivos de saída da simulação para mostrá-los em gráficos de duas dimensões.

O interpretador de comandos empregado para implementar os *scripts* foi o *bash* (GNU Bourne-Again Shell versão 1.14.7(1)) do sistema operacional Linux.

TABELA Anexo 4.1 - Relação dos *Scripts* e suas Funções sobre os Arquivos

Nome do <i>Script</i>	Função	Arquivos
Cenários	- Escalonar as chamadas do script Banda_Equivalente.	Entrada: -x- Saída: -x-
Banda_Equivalente	- Gerar arquivos textos para configurar as simulações no NIST ATM/HFC; - Escalonar a chamada do simulador variando a configuração do ambiente simulado; - Gravar no arquivo de saída um totalizador para a execução do <i>script</i> ;	Entrada: arquivos de configuração. Ex.: vbrserv_model, switch_model, bte_model e outros modelos. Saída: saida_sim
CLR_x_Conexoes	Montar gráfico da Razão de Perda de Células (CLR) em função: - Número de conexões multiplexadas; - Banda Equivalente alocada;	Entrada: saida_sim e def_graf_CLR Saída: -x-
CDV_x_Conexoes	Montar gráfico da Variação do Atraso (CDV) em função: - Número de conexões multiplexadas; - Banda Equivalente alocada;	Entrada: saida_sim e def_graf_CDV Saída: -x-
CTD_x_Conexoes	Montar gráfico do Atraso de Transferência (CTD) em função: - Número de conexões multiplexadas; - Banda Equivalente alocada;	Entrada: saida_sim e def_graf_CTD Saída: -x-
CDVic_x_Conexoes	Montar gráfico do Intervalo de Confiança da Variação do Atraso (CDV) em função: - Número de conexões multiplexadas; - Banda Equivalente alocada;	Entrada: saida_sim e def_graf_CDVic Saída: -x-
CTDic_x_Conexoes	Montar gráfico do Intervalo de Confiança do Atraso de Transferência (CTD) em função: - Número de conexões multiplexadas; - Banda Equivalente alocada;	Entrada: saida_sim e def_graf_CTDic Saída: -x-
CTDcel_x_Conexoes	Montar gráfico da Variação do Atraso (CDV) dado em unidades de <i>timeslot</i> do enlace em função: - Número de conexões multiplexadas; - Banda Equivalente alocada;	Entrada: saida_sim e def_graf_CDVcel Saída: -x-
PL	Executar o Gnuplot a partir da linha de comando. Script desenvolvido por Michael Sternberg <sternberg@physik.tu-chemnitz.de>	Entrada: -x- Saída: -x-

A seguir são apresentados os fontes dos *scripts* e os arquivos utilizados.

Conteúdo dos Scripts

Script: Cenários

```
cd /home/adriano/simulacao/cenarios/cen_11

/home/adriano/simulacao/banda_equivalente VBR 10 5.073585 6060 33940 100
/home/adriano/simulacao/banda_equivalente VBR 10 5.073585 6060 33940 1000
/home/adriano/simulacao/banda_equivalente VBR 10 5.073585 6060 33940 10000
```

Script: Banda_Equivalente

```
#####
##                                     ##
## Script para:                       ##
## =====                           ##
##                                     ##
## - Gerar "n" arquivos de configuracao CBR ou VBR                       ##
##   Onde "n" e' igual ao numero de conexoes simultaneas de 1 a "n".     ##
##   Exemplo: n=3, sao gerados 3 arquivos:                               ##
##     - o primeiro arquivo com 1 conexao                               ##
##     - o segundo arquivo com 2 conexoes                               ##
##     - o terceiro arquivo com 3 conexoes                               ##
##                                     ##
## - Executar "m" Simulacoes com NIST ATM/HFC                             ##
##   Variando a Banda Equivalente para cada arquivo de configuração     ##
##                                     ##
#####
## Data: 18/08/2000                                                         ##
## Autor: Adriano Ramos Goncalves                                         ##
##                                     ##
## Entrada:                                                                  ##
##   - Categoria de Servico (CBR/VBR)                                     ##
##   - Numero de Conjuntos de conexoes (maximo 10 conjuntos)             ##
##   - PCR (Mbit/s)                                                       ##
##   - MBS (tempo medio da rajada em us)                                  ##
##   - Toff (tempo medio entre rajadas em us)                             ##
##   - Tamanho Buffer                                                       ##
##                                     ##
## Exemplo de chamada:                                                     ##
##   banda_equivalente VBR 10 2 5000 30000 100                             ##
##                                     ##
#####

rm -f conf_sim*
arq_resultado="saida_sim_v9"

if [ -f $arq_resultado ]
then
  arq_resultado="$arq_resultado.$$"
fi

gawk -v categoria=$1 -v tot_con=$2 -v PCR=$3 -v MBS=$4 -v TOFF=$5 -v BUF=$6 '

BEGIN {

  dir = "/home/adriano/simulacao/modelos/"

#   Se alterar a combinacao de conexoes: alterar no script de execucao simulacao
  comb_cnx[1]=1
  comb_cnx[2]=2
  comb_cnx[3]=3
  comb_cnx[4]=4
  comb_cnx[5]=5
  comb_cnx[6]=6
  comb_cnx[7]=8
  comb_cnx[8]=10
  comb_cnx[9]=15
  comb_cnx[10]=25
```

```

if (tot_con > 10)
{
  print "=====> Maximo 10 conjuntos <====="
  exit(1)
}

if (categoria == "CBR")
{
  prep_cbrserv()
  arq_servico = "conf_sim_cbr"
}
else
  if (categoria == "VBR")
  {
    prep_vbrserv()
    arq_servico = "conf_sim_vbr"
  }
  else
  {
    print "=====> Categoria de servico invalida <====="
    exit(1)
  }

for (ind = 1; ind <= tot_con; ind += 1)
{
  num_arq = comb_cnx[ind]

  arq_saida = arq_servico "_" num_arq

  gera_switch()

  for (num_fonte = 1; num_fonte <= num_arq; num_fonte += 1)
  {
    if (categoria == "CBR")
      gera_cbrserv()
    else
      gera_vbrserv()
  }

  for (num_fonte = 1; num_fonte <= num_arq; num_fonte += 1)
  {
    if (categoria == "CBR")
      gera_cbrclie()
    else
      gera_vbrclie()
  }

  comando = "cat " dir "lk0_model >> " arq_saida
  system(comando)

  for (num_fonte = 1; num_fonte <= num_arq; num_fonte += 1)
  {
    gera_neighbor()
  }

  print " " >> arq_saida

  for (num_fonte = 1; num_fonte <= num_arq; num_fonte += 1)
  {
    gera_rota()
  }

  close (arq_saida)
}
}

function prep_cbrserv()
{
  comando = "rm -f cbrserv_model_tmp"
  system(comando)

  arq_entrada = dir "cbrserv_model"
  arq_saida = "cbrserv_model_tmp"
  troca = "<pcr>"
  por = PCR
}

```

```

        Le_Altera()
    }

function prep_vbrserv()
{
    comando = "rm -f vbrserv_model_tmp*"
    system(comando)

    arq_entrada = dir "vbrserv_model"
    arq_saida = "vbrserv_model_tmp"
    troca = "<pcr>"
    por = PCR

    Le_Altera()

    arq_entrada = "vbrserv_model_tmp"
    arq_saida = "vbrserv_model_tmp1"
    troca = "<mbs>"
    por = MBS

    Le_Altera()

    arq_entrada = "vbrserv_model_tmp1"
    arq_saida = "vbrserv_model_tmp2"
    troca = "<toff>"
    por = TOFF

    Le_Altera()
}

function gera_switch()
{
    arq_entrada = dir "switch_model"
    troca = "<buffer>"
    por = BUF

    Le_Altera()

    print " " >> arq_saida
}

function gera_bte()
{
    arq_entrada = dir "bte_model"
    troca = "<cx>"
    por = num_fonte

    Le_Altera()

    print " " >> arq_saida
}

function gera_cbrserv()
{
    arq_entrada = "cbrserv_model_tmp"
    troca = "<cx>"
    por = num_fonte

    Le_Altera()

    print " " >> arq_saida
}

function gera_vbrserv()
{
    arq_entrada = "vbrserv_model_tmp2"
    troca = "<cx>"
    por = num_fonte

    Le_Altera()

    print " " >> arq_saida
}

function gera_cbrclie()
{

```

```

    arq_entrada = dir "cbrclie_model"
    troca = "<cx>"
    por = num_fonte

    Le_Altera()

    print " " >> arq_saida
}

function gera_vbrclie()
{
    arq_entrada = dir "vbrclie_model"
    troca = "<cx>"
    por = num_fonte

    Le_Altera()

    print " " >> arq_saida
}

function gera_lk()
{
    arq_entrada = dir "lk_model"
    troca = "<cx>"
    por = num_fonte

    Le_Altera()

    print " " >> arq_saida
}

function gera_neighbor()
{
    arq_entrada = dir "neighbor_model"
    troca = "<cx>"
    por = num_fonte

    Le_Altera()
}

function gera_rota()
{
    arq_entrada = dir "rota_model"
    troca = "<cx>"
    por = num_fonte

    Le_Altera()
}

function Le_Altera()
{
    eof = getline < arq_entrada

    while (eof != 0)
    {
        alt = gensub(troca,por,"g",$0)
        print alt >> arq_saida
        eof = getline < arq_entrada
    }

    close (arq_entrada)
    close (arq_saida)
}

,

#####
#
# Executa Simulacoes
#
#####

gawk -v categoria=$1 -v tot_con=$2 -v PCR=$3 -v MBS=$4 -v TOFF=$5 -v
saida=$arq_resultado -v BUF=$6 '

```

```

BEGIN {
    cenario = "Video Conferencia"
    conta_celula = 0
    conta_simulacoes = 0
    seg_inicio = systime()

    comb_cnx[1]=1
    comb_cnx[2]=2
    comb_cnx[3]=3
    comb_cnx[4]=4
    comb_cnx[5]=5
    comb_cnx[6]=6
    comb_cnx[7]=8
    comb_cnx[8]=10
    comb_cnx[9]=15
    comb_cnx[10]=25

    if (categoria == "CBR")
    {
        arq_servico = "conf_sim_cbr"
        BE_ini = 0.5
        BE_fin = 0.9
        BE_inc = 0.2
    }
    else
    if (categoria == "VBR")
    {
        arq_servico = "conf_sim_vbr"
        BE_ini = int( 100 * (MBS / (MBS + TOFF))) / 100
        BE_fin = 1.025
        BE_inc = 0.025
    }

    rep = int((BE_fin - BE_ini) / BE_inc) + 1
    tot_simulacoes = tot_con * rep

    for (ind = 1; ind <= tot_con; ind += 1)
    {
        num_fontes = comb_cnx[ind]

        arq_conf = arq_servico "_" num_fontes

        for (tx_efetiva = BE_ini; tx_efetiva < BE_fin; tx_efetiva += BE_inc)
        {
            if (tx_efetiva > 1) tx_efetiva = 1

            vazao = tx_efetiva * PCR * num_fontes

            conta_simulacoes += 1

            Tempo_execucao()

            print "\n\n"
            print "-----X-----"
            print conta_simulacoes "/" tot_simulacoes "\t\tTaxa de Pico : "
            PCR "\t\t(" exe_tempo_agora ")"
            print "\t\tConexoes      : " num_fontes
            print "\t\tBanda Efetiva: " tx_efetiva * 100 "%"
            print "\t\tTaxa de Vazao: " vazao
            print "-----X-----"

            antigo = "<taxa>"

            Le_Altera()
            Executa_Simulacao()

            ult_taxa = tx_efetiva
        }
    }

    print "#=====
=====#" >> saida
    print "#\t\t Servico: " cenario >> saida
    print "#=====
=====#" >> saida
    print "# \t\t Categoria: " categoria >> saida
    print "# \t\t PCR = " PCR " Mbit/s" >> saida

```

```

if (categoria == "VBR")
{
    print "# \t\t MBS = " MBS " us" >> saida
    print "# \t\t MBSO= " TOFF " us" >> saida
}

cons = ""
for (ind = 2; ind <= tot_con; ind += 1)
    cons = sprintf ("%s,%s",cons, comb_cnx[ind])

print "#=====
=====#" >> saida
print "# \t Cenario: " conta_simulacoes " simulacoes" >> saida
print "# \t\t Buffer:      [" BUF " celulas ]" >> saida
print "# \t\t Conexoes:  [1" cons "]" "\t(" tot_con " ocorrencias)" >>
saida
print "# \t\t B. Efetiva: [" BE_ini * 100 ";" ult_taxa * 100 "]" (" BE_inc
* 100 "%)" "\t(" rep " ocorrencias)" >> saida
print "#=====
=====#" >> saida
print "# \t Execucao: " conta_celula " celulas geradas" >> saida

seg_final = systime()

hr_inicio = strftime("%d/%m %T",seg_inicio)
hr_final = strftime("%d/%m %T",seg_final)
# diminui 75600 seg (21 horas em relacao a 24 horas UTC)
dif_tempo = seg_final - seg_inicio - 75600
exe_tempo = strftime(" %T",dif_tempo)

print "# \t\t Data e Hora Inicial: " hr_inicio >> saida
print "# \t\t Data e Hora Final:   " hr_final >> saida

print "# \t\t Duracao:                " exe_tempo >> saida
print "#=====
=====#" >> saida

close (saida)

print "\n\n\n"
print "=====> Resultados gravados no arquivo: " saida
print "Duracao: " exe_tempo
print "\n\n\n"
}

function Le_Altera()
{
    eof = getline < arq_conf

    while (eof != 0)
    {
        alt = gensub(antigo,vazao,"g",$0)
        print alt > "conf_temp"
        eof = getline < arq_conf
    }

    close ("conf_temp")
    close (arq_conf)
}

function Executa_Simulacao()
{
    comando = "sim -i -x -s 963360534 -o sim_log -int 500000 -w 1000000
conf_temp 48000000"
    system(comando)

    comando = "tail -1 sim_log > temp_res_v8"
    system(comando)

    $0 = "vazio"
    getline < "temp_res_v8"

    conta_celula = conta_celula + $3
}

```

```

tempo_celula = 424 / vazao

linha = $0 " " tx_efetiva " " vazao " " tempo_celula " " BUF
print linha >> saida

close ("temp_res_v8")

comando = "rm -f sim_log"
system(comando)

comando = "rm -f conf_temp"
system(comando)

comando = "rm -f temp_res_v8"
system(comando)
}

function Tempo_execucao()
{
seg_agora = systime()

#
diminui 75600 seg (21 horas em relacao a 24 horas UTC)
dif_tempo = seg_agora - seg_inicio - 75600
exe_tempo_agora = strftime("%T",dif_tempo)
}

'
rm -f cbrserv_model_tmp
rm -f vbrserv_model_tmp*

###> Fim Script BANDA EQUIVALENTE

```

Scripts: CLR_x_Conexoes, CTD_x_Conexoes e CDV_x_Conexoes

```

#####
#          CLR_x_Conexoes          #
#          #                       #
# Script para chamar gnuplot      #
#          #                       #
#   Entrada: $1 = numero do arquivo #
#          #                       #
#   Exemplo Chamada: ./CLR_x_Conexoes v9.2751 #
#####
titulo="CLR X Conexoes"
arq_saida="saida_sim_$1"

p01='-u ($15):($2>1 ? 1/0 : $2>0 ? $8 : 1/0) ti "01" with linespoint lw 1.5 ps 1.5'
p02='-u ($15):($2>2 ? 1/0 : $2>1 ? $8 : 1/0) ti "02" with linespoint lw 1.5 ps 1.5'
p03='-u ($15):($2>3 ? 1/0 : $2>2 ? $8 : 1/0) ti "03" with linespoint lw 1.5 ps 1.5'
p04='-u ($15):($2>4 ? 1/0 : $2>3 ? $8 : 1/0) ti "04" with linespoint lw 1.5 ps 1.5'
p05='-u ($15):($2>5 ? 1/0 : $2>4 ? $8 : 1/0) ti "05" with linespoint '
p06='-u ($15):($2>6 ? 1/0 : $2>5 ? $8 : 1/0) ti "06" with linespoint '
p08='-u ($15):($2>8 ? 1/0 : $2>7 ? $8 : 1/0) ti "08" with linespoint '
p10='-u ($15):($2>10 ? 1/0 : $2>9 ? $8 : 1/0) ti "10" with linespoint '
p15='-u ($15):($2>15 ? 1/0 : $2>14 ? $8 : 1/0) ti "15" with linespoint '
p25='-u ($15):($2>25 ? 1/0 : $2>24 ? $8 : 1/0) ti "20" with linespoint '

pl "$p01" "$p02" "$p03" "$p04" "$p05" "$p06" "$p08" "$p10" "$p15" "$p25" -title
"$titulo" -f def_graf_CLR $arq_saida

```

Os *scripts* CTD e CDV são semelhantes com o do CLR, modificando apenas o parâmetro "\$8" acima em todas as linhas por "\$9" para CTD e por "\$10" para CDV.

Scripts: CDVic_x_Conexoes, CTDic_x_Conexoes

```
#####
#          CDVic_x_Conexoes          #
#          #                          #
# Script para chamar gnuplot        #
#          #                          #
#   Entrada: $1 = numero do arquivo #
#          #                          #
#   Exemplo Chamada: ./CDVic_x_Conexoes v9.2751 #
#####
titulo="CDV X Conexoes"
arq_saida="saida_sim_$1"

p01='-u ($15):($2>1 ? 1/0 : $2>0 ? $10 : 1/0) notitle with line lt 1'
p02='-u ($15):($2>2 ? 1/0 : $2>1 ? $10 : 1/0) notitle with line lt 2'
p03='-u ($15):($2>3 ? 1/0 : $2>2 ? $10 : 1/0) notitle with line lt 3'
p04='-u ($15):($2>4 ? 1/0 : $2>3 ? $10 : 1/0) notitle with line lt 4'
p05='-u ($15):($2>5 ? 1/0 : $2>4 ? $10 : 1/0) notitle with line lt 5'

p01i='-u ($15):($2>1 ? 1/0 : $2>0 ? $10 : 1/0):($12) ti "01" with yerrorbars lt 1 ps 1.5'
p02i='-u ($15):($2>2 ? 1/0 : $2>1 ? $10 : 1/0):($12) ti "02" with yerrorbars lt 2 ps 1.5'
p03i='-u ($15):($2>3 ? 1/0 : $2>2 ? $10 : 1/0):($12) ti "03" with yerrorbars lt 3 ps 1.5'
p04i='-u ($15):($2>4 ? 1/0 : $2>3 ? $10 : 1/0):($12) ti "04" with yerrorbars lt 4 ps 1.5'
p05i='-u ($15):($2>5 ? 1/0 : $2>4 ? $10 : 1/0):($12) ti "05" with yerrorbars lt 5 ps 1.5'

p1 "$p01" "$p01i" "$p02" "$p02i" "$p03" "$p03i" "$p04" "$p04i" "$p05" "$p05i" -title
"$titulo" -f def_graf_CDV $arq_saida

p01='-u ($15):($2>6 ? 1/0 : $2>5 ? $10 : 1/0) notitle with line lt 1'
p02='-u ($15):($2>8 ? 1/0 : $2>7 ? $10 : 1/0) notitle with line lt 2'
p03='-u ($15):($2>10 ? 1/0 : $2>9 ? $10 : 1/0) notitle with line lt 3'
p04='-u ($15):($2>15 ? 1/0 : $2>14 ? $10 : 1/0) notitle with line lt 4'
p05='-u ($15):($2>25 ? 1/0 : $2>24 ? $10 : 1/0) notitle with line lt 5'

p01i='-u ($15):($2>6 ? 1/0 : $2>5 ? $10 : 1/0):($12) ti "06" with yerrorbars lt 1 ps
1.5'
p02i='-u ($15):($2>8 ? 1/0 : $2>7 ? $10 : 1/0):($12) ti "08" with yerrorbars lt 2 ps
1.5'
p03i='-u ($15):($2>10 ? 1/0 : $2>9 ? $10 : 1/0):($12) ti "10" with yerrorbars lt 3 ps 1.5'
p04i='-u ($15):($2>15 ? 1/0 : $2>14 ? $10 : 1/0):($12) ti "15" with yerrorbars lt 4 ps
1.5'
p05i='-u ($15):($2>25 ? 1/0 : $2>24 ? $10 : 1/0):($12) ti "25" with yerrorbars lt 5 ps
1.5'

p1 "$p01" "$p01i" "$p02" "$p02i" "$p03" "$p03i" "$p04" "$p04i" "$p05" "$p05i" -title
"$titulo" -f def_graf_CDV $arq_saida
```

O *script* CTDic é semelhante ao CDVic, modificando apenas os parâmetros "\$10" para "\$9" e os parâmetros "\$12" para "\$11".

Script: CTDcel_x_Conexoes

```
#####
#          CTDcel_x_Conexoes          #
#          #                          #
# Script para chamar gnuplot        #
#          #                          #
#   Entrada: $1 = numero do arquivo #
#          #                          #
#   Exemplo Chamada: ./CTDcel_x_Conexoes v9.2751 #
#####
titulo="Atraso(celula) x Conexoes"
arq_saida="saida_sim_$1"

p01='-u ($15):($2>1 ? 1/0 : $2>0 ? $9/$17 : 1/0) ti "01" with linespoint lw 1.5 ps 1.5'
p02='-u ($15):($2>2 ? 1/0 : $2>1 ? $9/$17 : 1/0) ti "02" with linespoint lw 1.5 ps 1.5'
p03='-u ($15):($2>3 ? 1/0 : $2>2 ? $9/$17 : 1/0) ti "03" with linespoint lw 1.5 ps 1.5'
```



```

p04='-u ($15):($2>4 ? 1/0 : $2>3 ? $9/$17 : 1/0) ti "04" with linespoint lw 1.5 ps 1.5'
p05='-u ($15):($2>5 ? 1/0 : $2>4 ? $9/$17 : 1/0) ti "05" with linespoint '
p06='-u ($15):($2>6 ? 1/0 : $2>5 ? $9/$17 : 1/0) ti "06" with linespoint '
p08='-u ($15):($2>8 ? 1/0 : $2>7 ? $9/$17 : 1/0) ti "08" with linespoint '
p10='-u ($15):($2>10 ? 1/0 : $2>9 ? $9/$17 : 1/0) ti "10" with linespoint '
p15='-u ($15):($2>15 ? 1/0 : $2>14 ? $9/$17 : 1/0) ti "15" with linespoint '
p25='-u ($15):($2>25 ? 1/0 : $2>24 ? $9/$17 : 1/0) ti "25" with linespoint '

pl "$p01" "$p02" "$p03" "$p04" "$p05" "$p06" "$p08" "$p10" "$p15" "$p25" -title
"$titulo" -f def_graf_CTDcel $arq_saida

```

Arquivos de Configuração

Switch_model

```

# Simulador NIST-ATM Versao: 5.0
# Seed 965781655
# Time of snapshot (ticks) 0
# Tempo Simulado ==> 00:00:00 (0 seg)
# Tempo Execucao ==> 00:00:00 (0 seg)
component 'Switch' SWITCH 250 260
param 'Switch' 32 0 # Switch
param 0 12 0 # Delay to process a cell (uSec): 0
param 155 12 0 # Switching Slot time (Mbit/s): 155
param <buffer> 12 0 # Output q_size (cells, -1=inf): <buffer>
param 0 12 0 # High Threshold, Q Cong. Flag (cells): 0
param 0 12 0 # Low Threshold, Q Cong. Flag (cells): 0
param 0 12 0 # Logging every (ticks) (e.g. 1, 100): 1
pflags 2a 4 #Cells Received: 0
pflags 2a 4 #Cell Drop %: 0
pflags 2a 4 #Cells in VBR Q to LK1: 0
pflags 2a 4 #Cells dropped in VBR Q to LK1: 0
pflags 2a 4 #Cells in ABR Q to LK1: 0
pflags 2a 4 #Cells Dropped in ABR Q to LK1: 0
pflags 2a 1 #Congestion for Link LK1: FALSE
pflags 2a 4 #Cells in VBR Q to LK2: 0
pflags 2a 4 #Cells dropped in VBR Q to LK2: 0
pflags 2a 4 #Cells in ABR Q to LK2: 0
pflags 2a 4 #Cells Dropped in ABR Q to LK2: 0
pflags 2a 1 #Congestion for Link LK2: FALSE
pflags 2a 4 #Cells in VBR Q to LK0: 0
pflags 2a 4 #Cells dropped in VBR Q to LK0: 0
pflags 2a 4 #Cells in ABR Q to LK0: 0
pflags 2a 4 #Cells Dropped in ABR Q to LK0: 0
pflags 2a 1 #Congestion for Link LK0: FALSE

component 'TE0' BTE 250 150
param 'TE0' 32 0 # TE0
param -1 12 0 # Max Output Queue Size(-1=inf): -1
param 1 12 0 # Logging every (ticks) (e.g. 1, 100): 1
pflags 2a 4 #Cells Received: 0
pflags 2a 4 #Cells in VBR Q to LK0: 0
pflags 2a 4 #Cells dropped in VBR Q toLK0: 0
pflags 2a 4 #Cells in ABR Q to LK0: 0
pflags 2a 4 #Cells Dropped in ABR Q to LK0: 0

```

Bte_model

```

component 'TE<cx>' BTE 180 370
param 'TE<cx>' 32 0 # TE<cx>
param -1 12 0 # Max Output Queue Size(-1=inf): -1
param 1 12 0 # Logging every (ticks) (e.g. 1, 100): 1
pflags 2a 4 #Cells Received: 0
pflags 2a 4 #Cells in VBR Q to LK1: 0
pflags 2a 4 #Cells dropped in VBR Q toLK1: 0
pflags 2a 4 #Cells in ABR Q to LK1: 0
pflags 2a 4 #Cells Dropped in ABR Q to LK1: 0

```

VBRserv_model

```

component 'GER<cx>' VBRCONNECTION 180 420
param 'GER<cx>' 32 0 # GER<cx>
param <pcr> 12 0 # Bit Rate (Mbits/s): <pcr>

```

```

param <mbs> 12 0 # Mean Burst Length (usecs): <mbs>
param <toff> 12 0 # Mean Interval Between Bursts (usecs): <toff>
param 1 12 0 # Start time (usecs): 1
param 1500 12 0 # Number of MBits to be sent: 1500
pflags 2a 4 #Celulas Enviadas para ca': 0
pflags 2a 4 #Celulas Recebidas: 0
pflags 2e 4 # Tamanho Subintervalo (ms): 0
pflags 2a 4 # Subintervalo numero: 0
pflags 2a 4 # Perda Media (%): 0
pflags 2a 4 # Atraso Medio (usecs): 0
pflags 2a 4 # Interv. Conf. 99% (-/+ usecs): 0
pflags 2a 4 # Jitter Medio (usecs): 0
pflags 2a 4 # Interv. Conf. 99% (-/+ usecs): 0
pflags 22 0

```

VBRclie_model

```

component 'REC<cx>' VBRCONNECTION 186 99
param 'REC<cx>' 32 0 # REC<cx>
param 0 12 0 # Bit Rate (Mbits/s): 0
param 0 12 0 # Mean Burst Length (usecs): 0
param 0 12 0 # Mean Interval Between Bursts (usecs): 0
param 0 12 0 # Start time (usecs): 0
param 0 12 0 # Number of MBits to be sent: 0
pflags 2a 4 #Celulas Enviadas para ca': 0
pflags 2a 4 #Celulas Recebidas: 0
pflags 2a 4 # Tamanho Subintervalo (ms): 0
pflags 2a 4 # Subintervalo numero: 0
pflags 2a 4 # Perda Media (%): 0
pflags 2a 4 # Atraso Medio (usecs): 0
pflags 2a 4 # Interv. Conf. 99% (-/+ usecs): 0
pflags 2a 4 # Jitter Medio (usecs): 0
pflags 2a 4 # Interv. Conf. 99% (-/+ usecs): 0
pflags 22 0

```

Lk_model

```

component 'LK<cx>' LINK 211 315
param 'LK<cx>' 32 0 # LK<cx>
param 155 12 0 # Link Speed (Mbits/sec): 155
pflags 2 0
pflags 2 0
pflags 2 0
pflags 2 0
pflags 2 0
pflags 2 0
param 1 12 0 # Distance (Km): 1
pflags 2a 4 #Link rate (Mbit/s) to TEcx: 0
pflags 2a 4 #Link rate (Mbit/s) to Switch: 0

```

Neighbor_model

```

neighbor1 'GER<cx>' 'Switch'
neighbor1 'TE0' 'REC<cx>'
neighbor1 'REC<cx>' 'TE0'
neighbor1 'Switch' 'GER<cx>'

```

Rota_model

```

route1 'GER<cx>' 'Switch' 'LK0' 'TE0' 'REC<cx>'

```

Observações:

- Trocando o parâmetro do "pflags" de 2e para 6e, os valores desse campo são armazenados no SIM_LOG.
- No "param `Servidor` 32 0", trocando o valor 32 para 72, armazena no SIM_LOG a quantidade de células recebidas.
- Para habilitar a impressão das estatísticas por subintervalo utilizar o parâmetro 'Subintervalo numero' passando de 2a para 6a.

Bibliografia

- [ALB98] ALBERTI, A. M. **Simulação de Redes ATM**. Campinas: Faculdade de Engenharia Elétrica e Computação - Unicamp, nov. 1998. p. 27-31.
- [ANI82] ANICK, D. et al. Stochastic Theory of Data-Handling System with Multiple Sources. **The Bell Technical Journal**, New York, v.61, n.8, p.1871, 1982.
- [BEL95] BELLCORE TECHNICAL ADVISORY. **Broadband ISDN Switching System Generic Requirements**, TA-NWT-001110, Revision 2. June 1995.
- [BEL96] BELLCORE TECHNICAL ADVISORY. **Generic Requirements for the Operations of ATM Network Elements**, GR-1248-CORE, n.3. Aug.1996.
- [BEN97] BENSAOU, B. et al. Estimation of the Cell Loss Ratio in ATM Networks with a Fuzzy System and Application to Measurement-Based Call Admission Control. **IEEE/ACM Transactions on Networking**, New York, v.5, n. 4, p.572-584, 1997.
- [CER97] CEREDA, R. L. D.; CRUZ, M. A. C.; DUTRA, L. S. V. **ATM – O Futuro das Redes**. São Paulo: Makron Books e Brisa, 1997. p32-34.
- [COU95] COUCOUBETIS, C. et al. Admission Control and Routing in ATM Networks using Inferences from Measured Buffer Occupancy. **IEEE Transactions on Communications**, New York, v.43, n. 2/3/4, p.1178-1784, 1995.
- [DJM97] DZIONG, Z.; JUDA, M.; MADSON, L.G. A Framework for bandwidth Management in ATM Networks-Aggregate Equivalent Bandwidth Estimation Approach. **IEEE/ACM Transactions on Networking**, New York, v.5, n. 1, p.134-1147, 1997.
- [DZI97] DZIONG, Zbigniew. **ATM Networking Resource Management**. New York: MacGraw-Hill, 1997.
- [FIS78] FISHMAN, G. S. **Principles of Discrete Event Simulation**. New York: John Wiley & Sons, 1978.
- [FLA91] FLANAGAN, W. **Frames, Packets and Cells in Broadband Networks**. New York: Telecom Library, 1991.
- [GIN99] GINSBURG, David. **ATM - Solutions for Enterprise Internetworking**. 2nd ed. Harlow: Addison-Wesley, 1999. p.391-426.

- [GIR98] GIROUX, Natalie; GANTI, Sudhakar. **Quality of Services in ATM Networks - State-of-the-art Traffic Management**. Upper Saddle River: Prentice Hall PTR, 1997. p.1-84.
- [GOL98] GOLMIE, N. et al. **The NIST ATM/HFC Network Simulator Operation and Programming Guide Version 4.1**. Disponível em <<http://isdn.ncsl.nist.gov>>. Acesso em: jul. 1999.
- [GUE91] GUERIN, R.; AHMADI, H.; NAGHSINEH, M. Equivalent Capacity and its Application to Bandwidth Allocation in High-speed Networks. **IEEE Journal on Selected Areas in Communications**, New York, v.9, n.7, p.968, 1991.
- [GUI98] GUIZANI, Mohsen; AMMAR, Rayes. **Designing ATM Switching Networks**. New York: MacGraw-Hill, 1998. p.341-344.
- [I.356] ITU-TELECOMMUNICATION STANDARDIZATION SECTOR. **B-ISDN ATM Layer Cell Transfer Performance**, recommendation I.356. Geneve, 1996.
- [I.610] ITU-TELECOMMUNICATION STANDARDIZATION SECTOR. **Integrated Service Digital Network (ISDN) Maintenance Principle - B-ISDN Operation and Maintenance**, recommendation I.610. Geneve, 1995.
- [JAI91] JAIN, Raj. **The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling**. Chinchester: John Wiley & Sons, 1991. p.179-496.
- [JIR00] JIRON, D. M. B. **Simulação de Videoconferências Através de Técnicas de Simulação por Eventos Discretos**. Porto Alegre: PPGC/UFRGS, 2000. 86p.
- [KOW97] KWOK, Timothy. **ATM - The New Paradigm for Internet, Intranet and Residential Broadband Services and Applications**. Upper Saddle River: Prentice Hall PTR, 1997. p.41-101.
- [MCD00] McDYSAN, D. **QoS & Traffic Management in IP & ATM Networks**. New York: McGraw-Hill, 2000.
- [MEN85] MENASCÉ, Daniel A.; ALMEIDA, Virgílio A. F. **Planejamento de Capacidade de Sistemas de Computação: Análise Operacional como Ferramenta**. Rio de Janeiro: Campus, 1985. p.13-35.
- [MON94] MONTEIRO, J. A. S. **Redes Digitais de Serviços Integrados de Faixa Larga (RDSI-FL)**. Recife: UFPE-DI, 1994. 206p. Trabalho apresentado na Escola de Computação, 9., 1994, Recife.

- [MUR91] MURASE, H. et al. A Call Admission Control Scheme for ATM Networks Using a Simple Quality Estimate. **IEEE Journal on Selected Areas in Communications**, New York, v.9, n.9, p.1461, 1991.
- [ONV95] ONVURAL, R. O. **Asynchronous Transfer Mode Networks: Performance Issues**. Norwood: Artech House, 1995. p.77-182.
- [PER96] PERROS, H. G.; ELSAYED, K. M. Call Admission Control Schemes: A Review. **IEEE Communications Magazine**, New York, v.34, n.11, p.82-91, 1996.
- [PIT96] PITTS, J. M; SCHORMANS, J. A. **Introduction to ATM Design and Performance**. Chinchester: John Wiley & Sons, 1996. 190p.
- [PNN1.0] ATM FORUM TECHNICAL COMMITTEE. **Private Network-Network Interface Specification, version 1.0**, AF-PNNI-0055. Mountain View, 1996.
- [RAM98] RAMASWAMY, S. et al. Equivalent Bandwidth Characterization for Real-time CAC in ATM Networks. **Journal of High Speed Networks**, Baltimore, p.1-25, 1998.
- [REG94] REGE, K. M. Equivalent Bandwidth and Related Admission Criteria for ATM Systems - A Performance Study. **International Journal of Communication Systems**, New York, v.7, p.181, 1994.
- [RF95] ROMANOW, A.; FLOYD, S. Dynamics of TCP Traffic over ATM Networks. **IEEE Journal on Selected Areas in Communications**, New York, v.13, n. 4. p.633-641, May 1995.
- [ROB96] ROBERTS, J. et al.(Ed). **Broadband Networks Teletraffic: Performance Evaluation and Design of Broadband Multiservices networks.**, Berlin: Springer Verlag, p.6-46, 1996. (Final Report of Action COST 242).
- [RUS93] RUSSELL, J. **Multimedia Networking Performance Requirements**. New York: Plenum, 1993. p. 187-198.
- [SAI91] SAITO, H.; SHIOMOTO, K. Dynamic Call Admission Control in ATM Networks. **IEEE Journal on Selected Areas in Communications**, New York, v. 9, n. 7, p.982-989, 1991.
- [SHA94] SHANMUGAN, S. K. Simulation and Implementation Tools for Signal Processing and Communication Systems. **IEEE Communications Magazine**, New York, July 1994.
- [SOA90] SOARES, L. F. G. **Modelagem e Simulação Discreta de Sistemas**. São Paulo: IME/USP, 1990. 254p. Trabalho apresentado na Escola de Computação, 7., 1990, São Paulo.

- [STE81] STEVENS, B. Capacity Management and DP Planning: A Basic Approach. **Computerworld Magazine**, [S.L.], Nov. 1981.
- [SYK93] SYKAS, E. D.; VLAKOS, K. M.; TSOUKATOS, K. P.; PROTONOTARIOS, E. N. Congestion Control - Effective Bandwidth Allocation in ATM Networks. **IFIP Transactions C [Communication Systems]**, [S.L.], v.14, p.65, 1993.
- [TM4.0] ATM FORUM TECHNICAL COMMITTEE. **Traffic Management Specification, version 4.0**, AF-TM-0056.000. Mountain View, 1996.
- [TOW98] TOWNSEND, J. K. et al. Simulation of Rare Events in Communications Networks. **IEEE Communications Magazine**, New York, August 1998.
- [UNI4.0] ATM FORUM TECHNICAL COMMITTEE. **User-Network Interface Signaling Specification, version 4.0**, AF-SIG-0061.000. Mountain View, 1996.