UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO
DEPARTAMENTO DE COMUNICAÇÕES

ESTUDO DE MÉTODOS DE CONTROLE DE ADMISSÃO DE CHAMADAS EM REDES ATM COM ANÁLISE DE DESEMPENHO

Por

HENRIQUE ANTÔNIO MIELLI CAMARGO

Banca Examinadora:
Prof. Dr. Dalton Soares Arantes (Orientador) - FEEC/UNICAMP
Prof. Dr. Jorge Guedes Silveira - PUCRS/UFRGS
Prof. Dr. Shusaburo Motoyama - FEEC/UNICAMP
Prof. Dr. Lee Luan Ling (Suplente) - FEEC/UNICAMP

Dissertação apresentada à Faculdade de Engenharia Elétrica e de Computação da Universidade Estadual de Campinas como parte dos requisitos exigidos para a obtenção do Título de Mestre em Engenharia Elétrica.

Campinas, 08 de outubro de 1999.

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

C14e

Camargo, Henrique Antônio Mielli
Estudo de métodos de controle de admissão de chamadas em redes ATM com análise de desempenho / Henrique Antônio Mielli Camargo.--Campinas, SP: [s.n.], 1999.

Orientador: Dalton Soares Arantes
Dissertação (mestrado) - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação.

1. Computadores - Controle de acesso. 2. Comutação de pacotes (Transmissão de dados). 3. Rede digital de serviços integrados. 4. Telecomunicações - Tráfego. I. Arantes, Dalton Soares. II. Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. III. Título.

Resumo

O objetivo desta tese é o estudo comparativo das diversas técnicas de Controle de Admissão de Chamadas em Redes ATM. Para isso, apresenta-se a conceituação básica para a compreensão e análise da maioria dos diferentes métodos de Controle de Admissão de Chamadas-CAC, introduzindo-se os significados da extensa relação de siglas que fazem parte do universo ATM/CAC. Utilizando-se o MATLAB versão 5 e o Simulador de Redes SimATM para ambiente Windows NT, em fase de desenvolvimento na Faculdade de Engenharia Elétrica e de Computação da UNICAMP, faz-se uma análise comparativa de três algoritmos de CAC conhecidos da literatura.

Abstract

The objective of this work is the comparative study of several Admission Control Techniques for ATM networks. To accomplish this, a number of different methods of Connection Admission Control is presented along with the meanings of a wide list of acronyms that are part of the CAC/ATM universe. Using MATLAB and the SimNT simulator designed for the Windows NT environment, which is under development at the Faculty of Electrical and Computer Engineering, a comparative analysis of three well-known Connection Admission Control algorithms is performed.

Aos meus saudosos pais Luis Gonzaga e Júnia,
à minha esposa Marli, às minhas filhas Natália e Juliana, e
aos meus queridos irmãos,
sem os quais esta tarefa teria sido muito mais árdua
e com os quais sou feliz.

Agradecimentos

Agradeço a todas as pessoas que direta ou indiretamente contribuiram para a realização desta dissertação de Mestrado. Em especial:

- ao Professor Dr. Dalton Soares Arantes pela excelente orientação, paciência e amizade nesses anos de trabalho em conjunto;
- aos meus familiares que, além de serem pacientes, me incentivaram a concretizar este objetivo;
- à Escola Técnica Federal de Mato Grosso - ETFMT (CEFET-MT) pela viabilização do afastamento para que eu realizasse este trabalho em tempo integral;
- à CAPES-PICDT pelo apoio financeiro.
- ao Professor Dr. Leonardo de Souza Mendes e sua equipe, pela disponibilização do Projeto SimATM (Projeto FAPESP 96/12372-2, em andamento) e, em especial, o aluno de doutorado e colega Ernesto Luiz Andrade Neto, pelo apoio;

Conteúdo

Lista de Figuras

Capítulo 1

Introdução

O objetivo do presente trabalho é apresentar um estudo comparativo de algoritmos de controle de admissão de chamadas (CAC) em redes ATM visando a avaliação destes quando são submetidos a duas fontes de diferentes características de tráfego ou seja, uma não auto-similar e outra com diversos graus de auto-similaridade.

Apresenta-se inicialmente, neste capítulo 1, a conceituação básica para redes ATM e em seqüência, no capítulo2, a conceituação sobre controle de congestionamento e modelos de fontes de tráfego utilizados em redes ATM .

Esta conceituação se justifica como o "background" necessário para a compreensão e análise teórica da maioria dos diferentes métodos de CAC apresentados no capítulo 3 e, finalmente, no capítulo 4 é realizado um estudo numérico de três algoritmos de CAC associados dois a dois, o primeiro destinado a uma fonte de tráfego de voz de característica não auto-similar ou Poissoniana, os outros dois servem a fontes de tráfego auto-similar ou de natureza fractal.

No apêndice 1, faz-se uma síntese do processo de estabelecimento de conexão ATM ou fase "setup" com o objetivo de esclarecer a problemática do "peso" do gerenciamento na admissão de chamadas, bem como, situando o CAC na camada de gerenciamento de rede; no apêndice 2, apresenta-se os programas em Matlab-Versão 5 para obtenção dos resultados mostrados no capítulo 4 e por último, no apêndice 3, apresenta-se a relação de acrônimos empregados em toda a extensão deste trabalho.

A principal contribuição que se pretende com este trabalho, portanto, é a avaliação numérica e validação de algoritmos de CAC já conhecidos, quando estes são submetidos conjuntamente às diferentes fontes de tráfego já mencionadas, em um mesmo multiplex ATM.

A seguir, neste capítulo, discute-se então a evolução das redes de comunicações, desde rede telefônica comutada até a rede ATM e introduz-se vários termos utilizados neste trabalho.

1.1. Modo de transferência assíncrono - surgimento e constituição.

Desde a invenção do telefone, a *comutação de circuitos* tem sido a tecnologia dominante para a comunicação de voz, sendo constituída basicamente de nós de comutação e estações formando as redes de comutação. Estas executam a comunicação fazendo o estabelecimento de circuito, transferência de dados e a desconexão de circuitos.

Uma rede de comutação de circuitos é constituída das ligações de usuários às centrais de comutação local, e das conexões destas às centrais intermediárias (urbanas e interurbanas). As comutações podem ocorrer por divisão espacial, onde os circuitos são estabelecidos através de canais físicos nas centrais (serviço orientado a conexão), enquanto durar a chamada, ou comutações por divisão no tempo-TDM, onde cada ligação participa em um intervalo fixo de tempo. Na Figura 1.1, exemplificamos a comutação por divisão espacial em que o usuário entrante em (6) se conecta com outro conectado em (2), e em caminhos diferentes estão conectados outro par de clientes em (7) e (1).

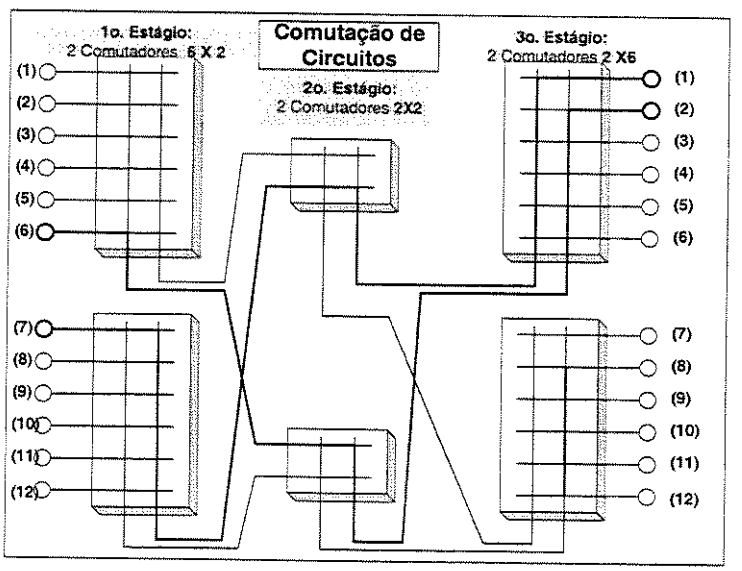

Figura 1.1: Comutação de Circuitos

Uma alternativa à comutação de circuitos é a *comutação por mensagens*. Nesta estratégia nenhum meio físico é estabelecido entre os dois terminais, para o tempo de comunicação. Em vez disso, o terminal transmissor envia o bloco de dados (mensagem de tamanho variável) para a primeira central de comutação (aquela a que está fisicamente conectado), onde o mesmo é armazenado, para depois ser encaminhado de central em central até que chegue na central a que está fisicamente conectado o terminal de destino. Cada uma destas centrais de comutação recebe o bloco de dados ou mensagem inteira, faz-se uma verificação de erros e depois o encaminha para a próxima central que executa as mesmas tarefas. Esta é uma estratégia de comutação não orientada à conexão e do tipo "store and forward".

Por outro lado, a *comutação por pacotes* baseia-se na separação dos bits de in-

formação em pacotes de tamanhos fixo e/ou variável. Uma parte dos bits destina-se ao controle dos pacotes e outra ao transporte da informação dos usuários (vide Figura 1.2). Com esta técnica é possível a utilização de **circuitos virtuais** (serviço orientado a conexão), ou seja, caminhos lógicos estabelecidos por critérios de gerenciamento de tráfego, sem comutação física, onde a parte que se destina ao controle de pacotes se encarrega de entregar a informação ao seu destino. A capacidade de processamento crescente dos equipamentos favoreceu a implantação dessas redes de comutação de pacotes.

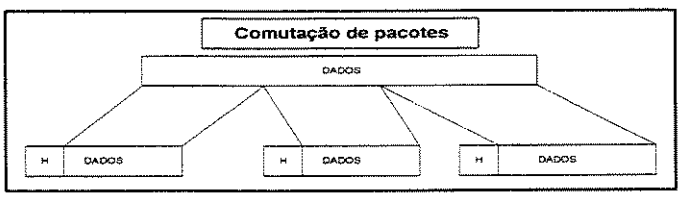

Figura 1.2: Comutação de Pacotes

Surgem então, novos mecanismos de transporte para atender à demanda, considerando a multiplicidade de serviços, além da telefonia requeridos agora também pela empresa moderna e pelo cidadão comum, tais como conexão à Internet (dados em altas taxas), FAX de alta qualidade e com transmissão rápida, videoconferência, marketing em tempo real e interativo, redes corporativas integradas e suas conexões, etc.

A tendência mundial de integração destes serviços em somente uma rede de transporte, interligando o usuário ao provedor de serviços e multiplexando seus sinais em apenas um sinal de alta velocidade, esta convergindo de forma inexorável para a *Rede Digital de Serviços Integrados-RDSI.*

No início, para fins de padronização, o ITU definiu um modelo de referência para a RDSI-FE integrando unicamente serviços de dados e voz em uma única rede, conforme Figura 1.3.

A necessidade de também integrar serviços de vídeo fez com que essa rede evoluísse para a RDSI-FL, conforme Figura 1.4, mas diversos entraves configuravam-se para a sua operacionalização, sendo o principal, a morosidade da maioria dos protocolos em uso (X.25, por exemplo), devido à necessidade de retransmissão de seqüências longas de bytes para a correção de erros.

Com isso, o ITU-T definiu o modo de transferência assíncrono-**ATM** como a tecnologia de transmissão, multiplexação e comutação para a RDSI-FL [1]. A tecnologia de modo de transferência assíncrono leva este nome devido ao fato de que a alocação de banda é determinada de acordo com a demanda dos usuários. Ao contrário do modo de transferência síncrono-SDH [2], um canal não é identificado pela posição fixa de seus "slots" em uma estrutura recorrente no tempo, mas pelo seqüenciamento assíncrono de segmentos fixos de informação chamados "células". Cada célula possui um cabeçalho e um campo de informação ("carga útil").

Assim, uma célula ATM é identificada por um rótulo, o cabeçalho H , que repre-

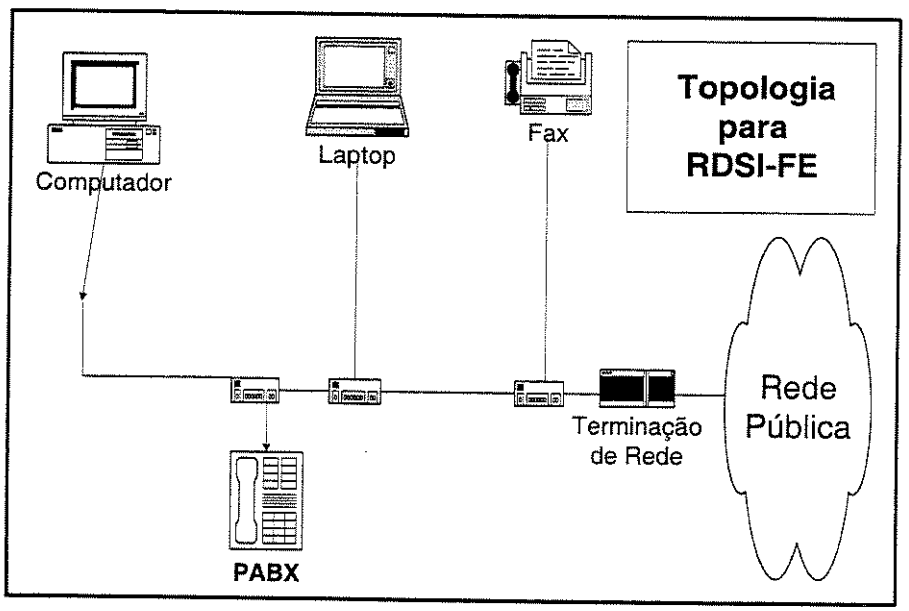

Figura 1.3: Topologia para RDSI-FE

senta a conexão com o circuito virtual estabelecido para o transporte de células (vide Figura 1.5, onde são mostrados os formatos das células para a interface usuário/rede (a) e rede/rede (b)).

Os campos de bits destas células são descritos a seguir:

Controle Genérico de Fluxo - 4 bits, é um campo que aparece somente do lado do usuário, na UNI e se destina ao controle do fluxo de células no acesso a rede de acordo com requisitos de QoS, sendo utilizado também como indicador de nível de prioridade de célula.

VPI -Identificador de Caminho Virtual - 8 bits, para UNI e 12 bits para NNI. Identifica o roteamento da célula ATM ao entrar e através da rede ATM.

VCI-Identificador de Canal Virtual - 8 bits. Identifica o canal utilizado dentro do caminho virtual. É um ponto SAP da camada ATM.

Tipo de Informação - 3 bits. Indica o tipo de informação inserida no campo de carga útil.

bit CLP-Bit de Prioridade de Perda de Célula - 1 bit, assume valor"0" para células prioritárias e valor "1" para células de menor prioridade.

HEC-Controle de Erro de Cabeçalho - 8 bits, controla erros de cabeçalho evitando erros de configuração de canal e caminho virtual, que provocam surtos de erros. Podem utilizar vários métodos, sendo que o mais comum é o polinomial com geração de Código $(X^8 + X^2 + X + 1)$. A outra função do HEC é a delineação da célula ATM, ou seja, determina o início e o fim de cada célula, auxiliando na definição da seqüência de células (detalhes em [2] [3] [4], além das normas ITU-T e ATM FORUM).

Ao se transmitir a célula ATM pela rede ATM, verificam-se peculiaridades que a

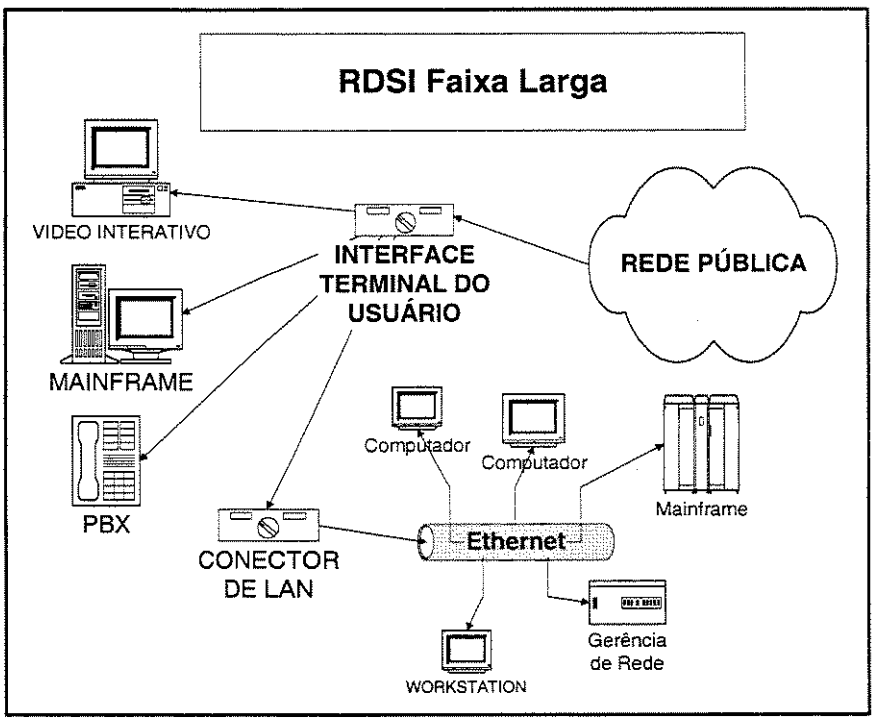

Figura 1.4: Topologia para RDSI-FL

diferenciam de outras redes, tais como QoS, velocidade/latência, etc. Isto será examinado mais detalhadamente adiante. A seguir discute-se os aspectos da camada ATM e de controle, no contexto das redes de computadores.

1.2. Redes de computadores: arquitetura IP ou OSI da ISO?

Nos anos 80, começou nos Estados Unidos o ambiente de redes interconectadas, aproveitando como espinha dorsal a já constituída rede ARPA (Projeto inicial das Forças Armadas dos EUA), surgindo assim, a *Internet*, que aproveitou as principais aplicações ARPA, como o protocolo FTP e TELNET, nessa época foi criado o IAB, que elaborou a arquitetura TCP/IP. Como o ambiente Internet não padroniza as sub-redes de acesso, tecnologias diferentes rapidamente puderam se interconectar como mostra a Figura 1.6.

Na Figura 1.6, várias tecnologias distintas possuem acesso via gateway, um conversor de protocolos. A multiplicidade de redes de acesso e a resultante "confusão" em torno dos diferentes protocolos e funções de processamento fez surgir a padronização pelo conceito de camadas OSI, que objetiva tornar os sistemas públicos e privados de redes em *sistemas abertos*, que são aqueles que podem ser interneconectados com qualquer outro sistema implementado a partir destas mesmas padronizações.

A padronização fica a cargo da ISO, órgão das Nações Unidas. O modelo da ISO é o OSI, que utiliza um modelo de 7 camadas, onde cada camada é prestadora de serviço

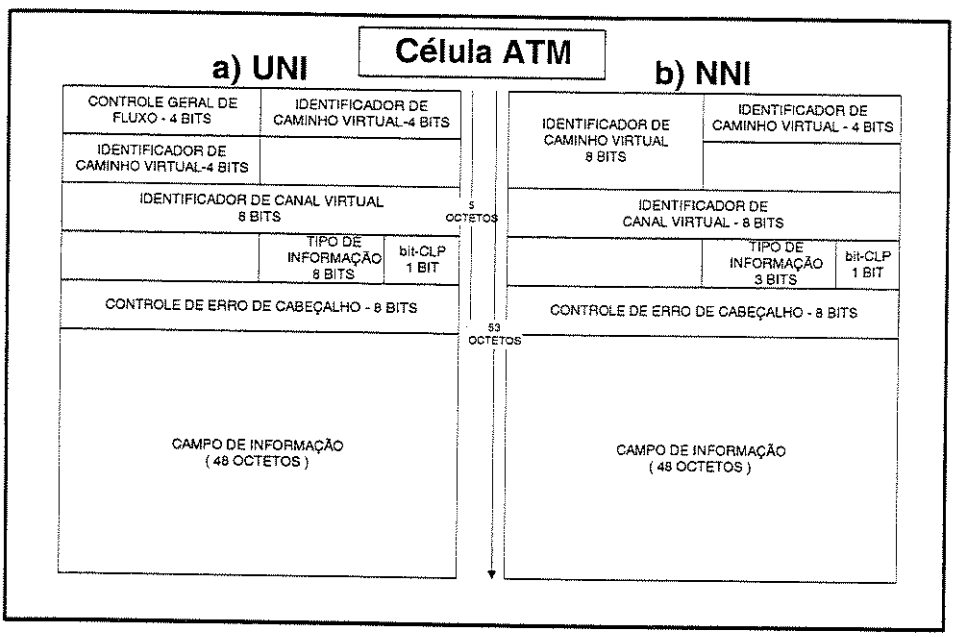

Figura 1.5: Célula ATM: a) UNI e b) NNI

à camada imediatamente superior e usuária dos serviços prestados pela camada imediatamente inferior. Hoje existe uma arquitetura elaborada de acordo com o movimento da demanda e necessidades imediatas do mercado TCP e outra, que é resultado de um projeto planejado e executado segundo etapas pré-determinadas, visando a padronização mundial e ignorando as pressões econômicas. Na Figura 1.7 mostra-se a composição da hierarquia funcional da arquitetura OSI.

Em maiores detalhes, a cada camada compete:

Camada física: É a camada de comunicação real, definida pelas características mecânicas, elétricas e funcionais (especifica interfaces físicas RS232, V24, X21, V 35, etc);

Camada de enlace: É a camada que possui a função de detectar e ou corrigir os erros que, por ventura, ocorram nos níveis físicos. Os serviços desta camada à camada física são os seguintes:

-Estabelecimento e liberação de conexão de enlace, montagem de quadros.

-Controle de seqüência e de fluxo de quadros.

-Controle de erro e de interconexão.

-Gerenciamento.

O nível de enlace pode fornecer ao nível superior de rede três tipos de serviços:

-Serviços sem conexão e sem reconhecimento

-Serviços sem conexão mas com reconhecimento e

-Serviços orientados à conexão

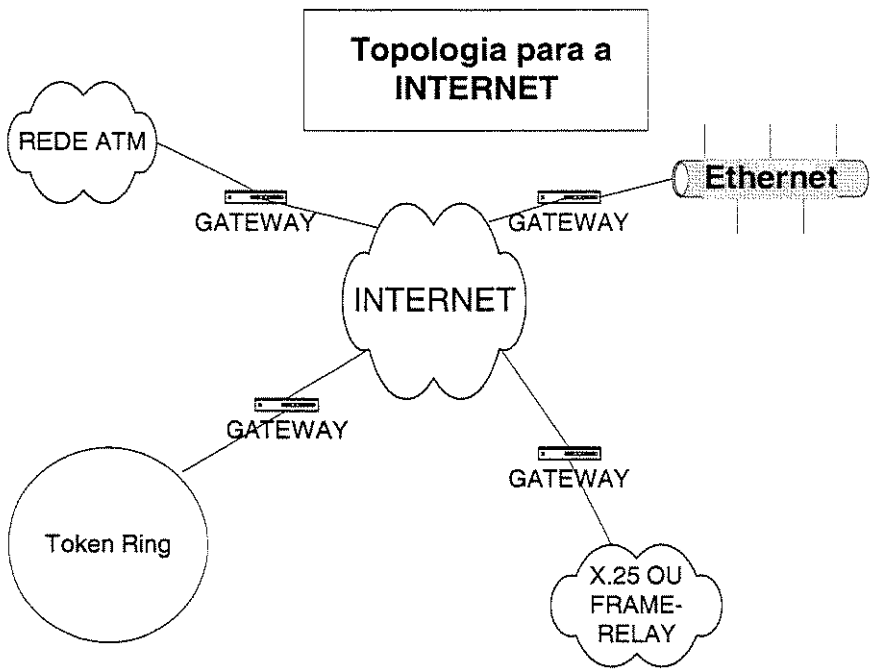

Figura 1.6: Topologia para a INTERNET

Os métodos de acesso podem ser Ethernet, "Token-Ring", DQDB e outros;

Camada de rede: É o nível de rede que executa as seguintes funções: roteamentos, conexão de redes fim-a-fim, endereçamento e multiplexação, além do próprio gerenciamento.

Neste ponto, é necessário especificar as formas de conexões inter-redes que ao nível físico são chamados de **repetidores** e ao nível de enlace chamados **pontes**, a nível de rede são chamados de **roteadores**, sendo que o **gateway** converte os protocolos até o nível de aplicação.

Camada de transporte: É a camada responsável pela movimentação dos dados de maneira eficiente e confiável entre processos em execução nos equipamentos conectados a uma rede independente da rede física.

Camadas de sessão: Tem por objetivo oferecer à entidades de apresentação cooperantes, os meios de organizar e sincronizar o seu diálogo, garantindo a troca orientada de dados através da conexão de sessão.

Camada de apresentação: Esta camada se relaciona com a sintaxe e a semântica na informação transmitida e tem como um exemplo típico a cifragem da informação.

Camada de aplicação: Implementa as funções de processamento de dados particulares a uma dada aplicação e também ao intercâmbio de informações.

A Figura 1.8 ilustra o processo de transferência de dados entre processos de aplicação dos sistemas abertos.

No software de aplicação do usuário os bits são separados em quadros de mensagens que em cada camada, em direção descendente à camada física, é acrescentado

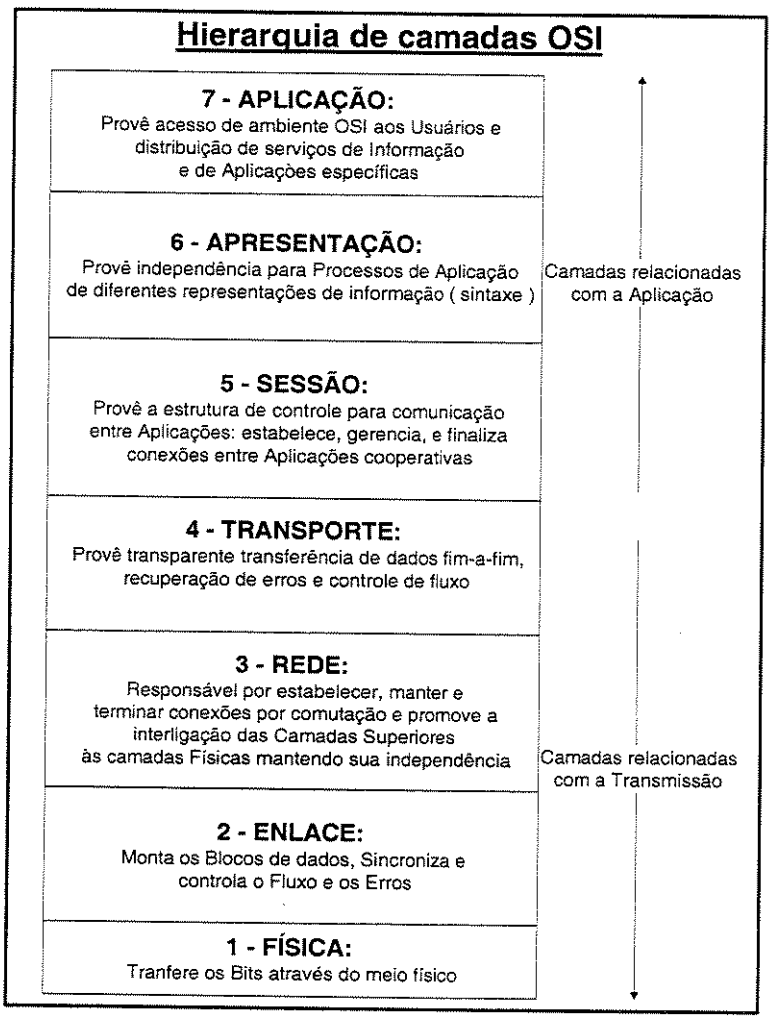

Figura 1.7: Hierarquia de Camadas Funcionais OSI

um novo cabeçalho definido pelo protocolo de cada camada (AH, PH, etc) até chegar à camada mais inferior e assim, transmitido pelo meio físico.

No caso de redes ATM, a os bits de aplicação do usuário são separados em pacotes de 48 bytes na sub-camada chamada camada de adaptação ATM e estes bytes, chegando à primeira camada (subcamada física de convergência física ATM) ganham cabeçalhos formando pacotes fixos de 53 bytes para poderem ser transmitidos. Mais detalhes nas referências sobre sistema OSI da ISO podem ser encontrados em [2] [6] [7] [8] [9], citando apenas alguns autores.

As funções específicas de cada camada são denominadas de subsistemas, que são constituídos de elementos ativos denominados **entidades**. Estas podem ser **pares,** quando dois sistemas abertos se comunicam na mesma camada ou **ímpares,** quando camadas adjacentes de um mesmo sistema aberto se comunicam.

Na Figura 1.9, o ponto de acesso de serviço *SAP* é o ponto de ligação entre entidades ímpares ou de camadas adjacentes, enquanto que a comunicação com camadas

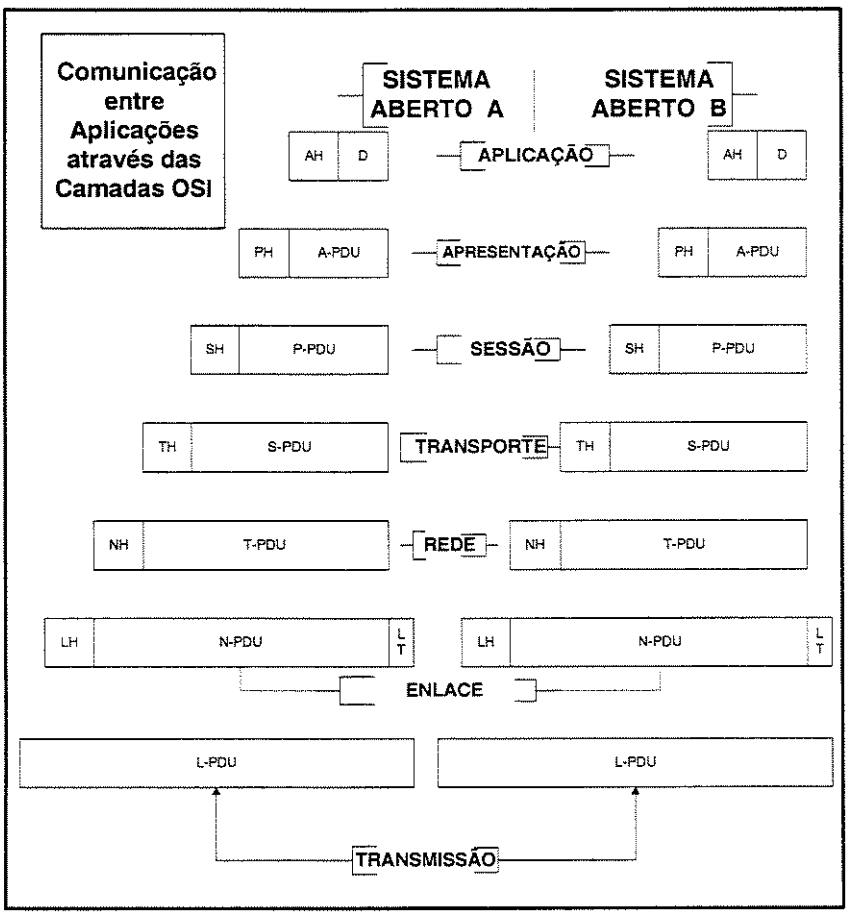

Figura 1.8: Camadas de Aplicação em Comunicação

paralelas é realizada através de protocolos das respectivas camadas.

Com estas definições, pode-se encerrar este "survey" sobre redes de computadores em geral e retoma-se o assunto objeto: redes *ATM*

1.3. Redes ATM

Estas são redes formadas pela interligação de *comutadores*, *multiplexers*, equipamentos do usuário (*terminais ATM* e "*workstations*" de supervisão e controle) que possibilitam a integração de outras redes e serviços entre si com transporte de informação em modo ATM. A Figura 1.10 ilustra esta definição.

Como visto neste exemplo, as interfaces R e S são interfaces do cliente, o "gateway" converte o protocolo de outras redes para o protocolo da rede ATM, U é a *UNI*, interface que conecta o cliente a rede ATM pública e *NNI*, a interface que conecta dois comutadores públicos ou duas redes públicas ATM.

A seguir, examina-se a definição dos elementos objetivados, ou seja: UNI, NNI comutadores e multiplexers e lembrando que **CAC** é um item do controle de tráfego das

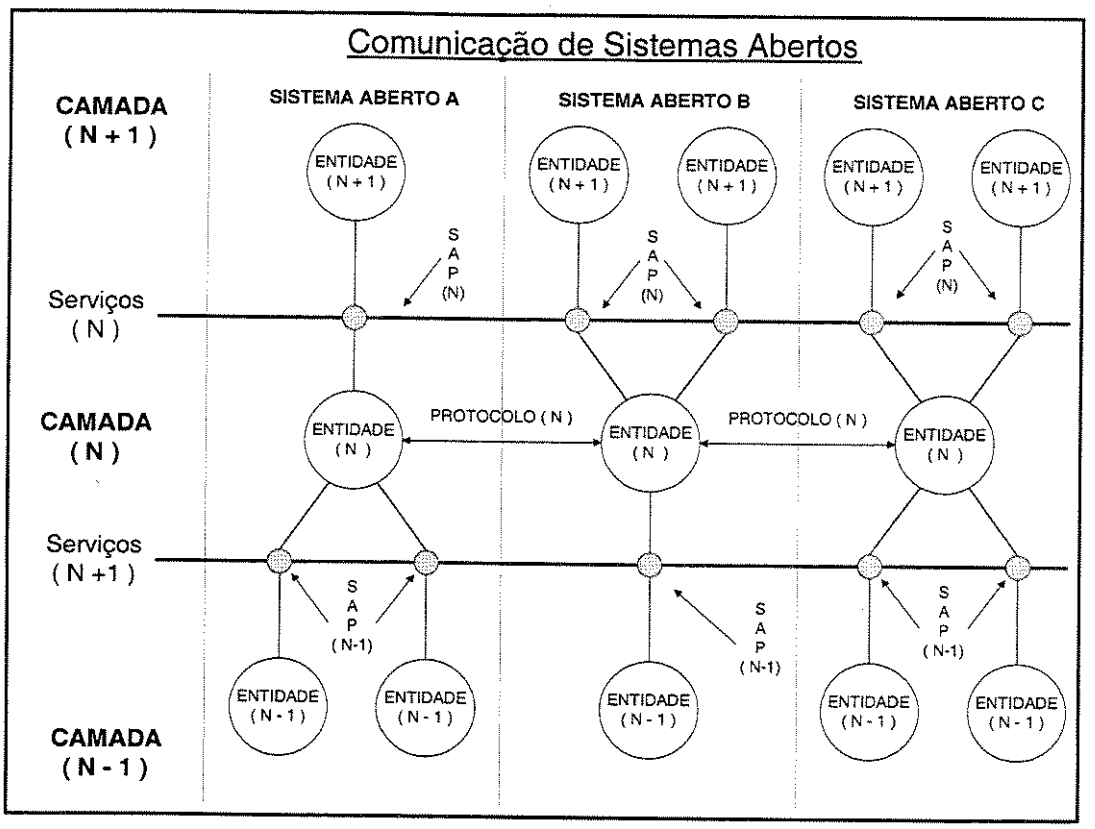

Figura 1.9: Comunicação entre Sistemas Abertos

redes ATM.

A UNI possui as seguintes atribuições: conexão física entre usuário e comutador, montagem e desmontagem do quadro ATM, sincronismo dos sinais na interface, emite células vazias para sincronização de mensagem quando não há o que transmitir, controle de erro no cabeçalho, sincronismo de célula, estruturação da célula, gerência das funções ATM, **controle de tráfego e congestionamento (incluindo CAC, que é interno às UNI, NNI e em roteadores, atuando na camada de gerência de rede nos planos de controle)** e sinalização de rede.

A NNI interliga internamente a rede ATM e **possui basicamente as mesmas funções**, com a diferença em mais bits identificadores de caminhos virtuais (VP's) que uma célula UNI (vide Figura 1.5 - célula ATM), pois sua função é estabelecer caminhos virtuais entre comutadores ATM independentes do controle do usuário.

Os *comutadores ATM* são elementos de rede que realizam várias ações, além da comutação de células. São responsáveis também pela função de: endereçamento (multi-cast); gerenciamento de falhas e gerenciamento de conexões.

Na parte central de um comutador está o **elemento comutador** que realiza a comutação propriamente dita. É a parte mais importante do projeto de um nó de comutação ATM, pois afeta o custo, a eficiência, a estendibilidade e a complexidade do

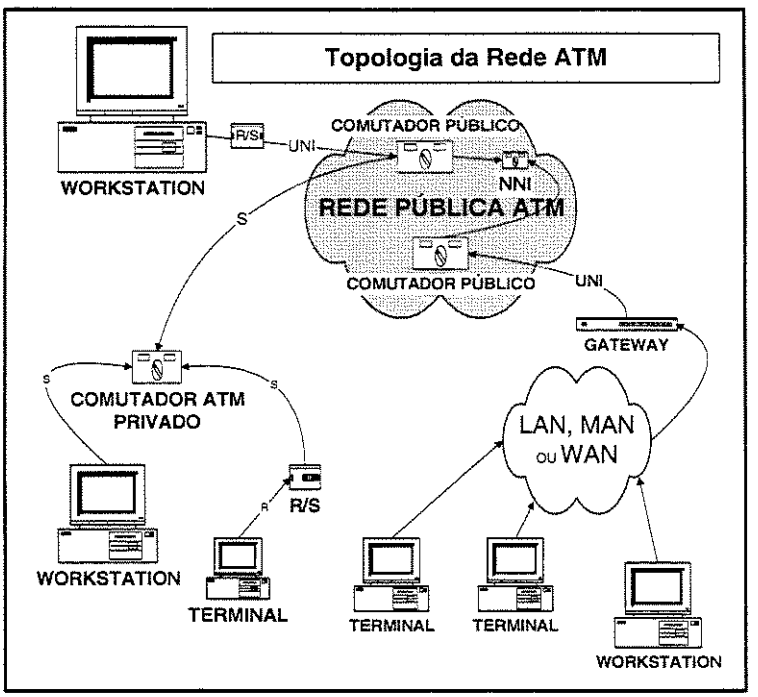

Figura 1.10: Topologia para a Rede ATM

nó.

Na Figura 1.11, vemos a disposição geral de um comutador ATM, com a configuração típica de **"buffers" na saída**, ilustrando o princípio de encaminhamento de células pelo processamento dos cabeçalhos, de acordo com uma tabela ou matriz de comutação.

Como, compreensivelmente, para não se perder a objetividade, este trabalho não pode entrar em detalhes dos comutadores e recomenda-se, para se aprofundar no assunto, [10] como consulta a um texto clássico e [11] [12] [13], para atualização.

O comutador ATM, juntamente com o concentrador e o multiplex ATM constituem um centro de comutação ATM. Este último elemento de rede, o multiplex ATM, pode ser encarado em duas perspectivas: como o elemento que promove a concentração de células de diferentes enlaces de entrada em um enlace de saída ou como dispositivo que se utiliza de propriedades estatísticas do comportamento do tráfego (filas), para concentrar ou difundir conexões [14].

Na Figura. 1.12, segue uma constituição do centro de comutação ATM.

Observa-se que o concentrador apenas torna o tráfego ATM composto deterministicamente de várias fontes semelhantes (cada tráfego de natureza semelhante deve ser encaminhado a um "buffer" distinto na saída para o controle por este "buffer" específico) e serve para otimizar a utilização do multiplex ATM, já a multiplexação pode ocorrer não só determinística, mas também estatisticamente na entrada do comutador ATM, em que todo tráfego ATM será regulado pelos seus "buffers" de saída.

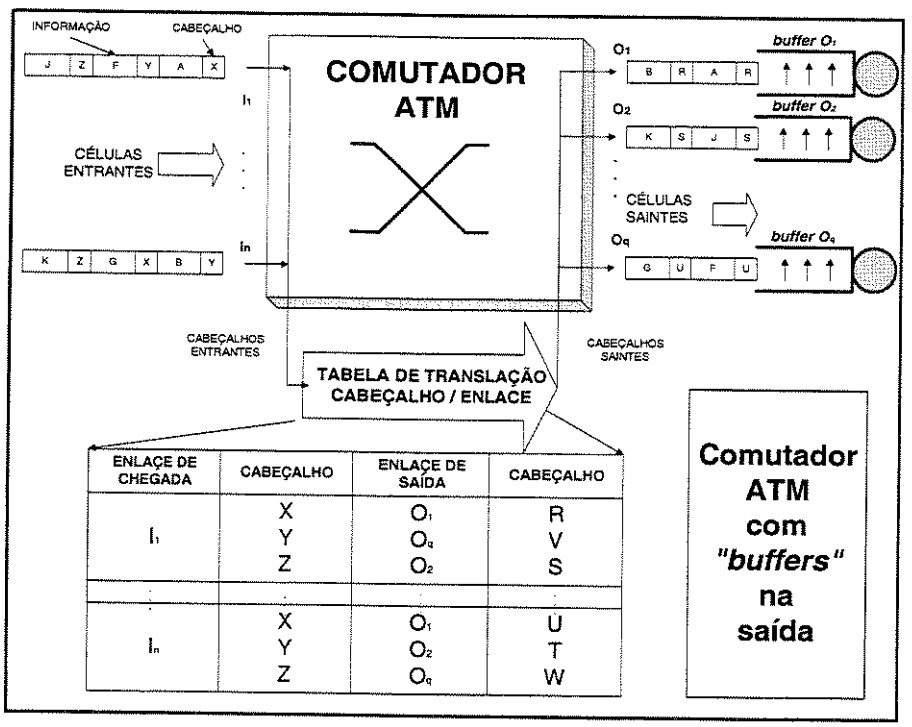

Figura 1.11: Comutador ATM com "buffers" na saída.

As redes ATM também possuem uma arquitetura definida em camadas, veja a figura 1.13. A *Camada física ATM* se subdivide em:

- *Subcamada de meio físico*: Especificação das características mecânicas, elétricas e ópticas do meio, bem como o sincronismo de bits;

- *Subcamada física de convergência de transmissão*: Gera e compõe a célula de 53 bits, verifica os erros de cabeçalho, embaralha e desembaralha para efeitos de códigos de linha o fluxo da camada ATM a partir dos grupamentos de bits das subcamadas de segmentação/meio físico, monta/desmonta as células ATM com respectivos cabeçalhos e campos de informação.

A *Camada de adaptação ATM (AAL)* a partir do agrupamento de bits da subcamada de convergência de transmissão/camada ATM, monta/desmonta a célula ATM com o cabeçalho e carga útil (vide célula ATM). A Camada de Adaptação ATM também é dividida em duas subcamadas, a saber:

- *SAR* - Subcamada de segmentação e recomposição tem a função de decompor as mensagens oriundas das camadas superiores, de forma a adaptá-las para o envio à camada adjacente inferior (camada ATM). O mesmo vale para o sentido inverso.

- *Subcamada de convergência* tem a função de propiciar serviços típicos da camada de transporte do modelo OSI aos serviços das camadas OSI superiores ("casamento" de modelo OSI e ATM).

Note-se, na Figura 1.13, que existem **planos de gerenciamento de camadas e de planos** e são estes planos que geram células especiais de gerenciamento e manutenção-

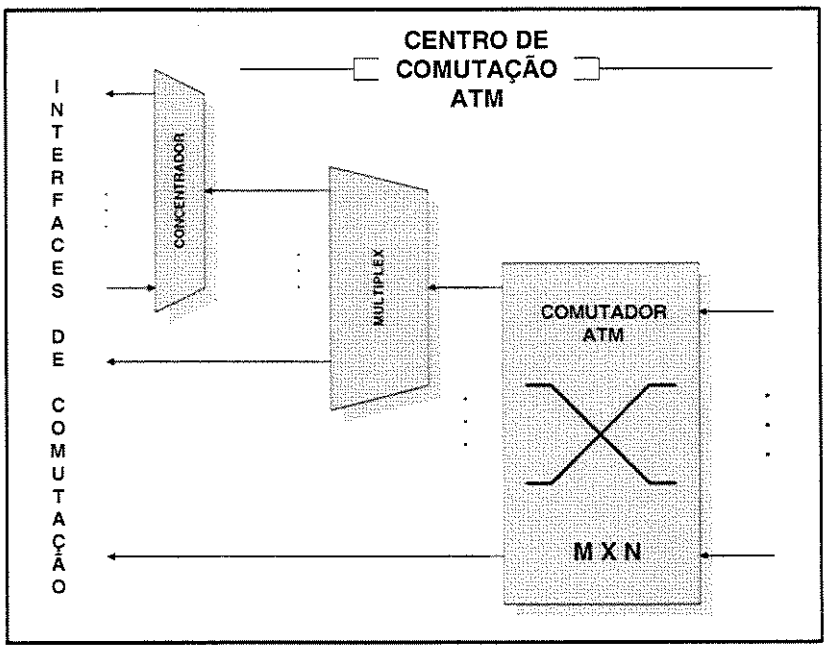

Figura 1.12: Centro de Comutação ATM

OAM, que informam a partir de bancos de dados de acesso comum a todos os elementos de rede, dados tais como: quantas conexões estão em tráfego, quantas conexões solicitam admissão, caminhos virtuais congestionados e livres, parâmetros descritores de tráfego da conexões em tráfego e entrantes, etc.

Os **planos de controle** se destinam a executar tarefas de controle de acordo com diretrizes oriundas dos **planos de gerenciamento de camadas**, e o **plano do usuário** se destina a implementar controles concedidos em contrato com o provedor da rede ATM.

Por outro lado, conforme o tipo de tráfego ATM e a relação entre a fonte e o destino (orientado à conexão ou não) a camada AAL provê quatro *classes de serviços*, com protocolos distintos, conforme tabela da Figura 1.14.

Detalhes destes protocolos podem ser encontrados em [4] e em várias outras referências ITU-T, ATM FORUM, etc.

E, a partir da ação das camadas superiores (enlace, rede e transporte) na UNI, NNI e roteadores são realizadas as funções de gerenciamento/monitoração de tráfego ATM. Na camada de rede ATM e em seu respectivo plano de gerenciamento é realizado o tratamento de **CAC**, objeto deste trabalho. Dos planos de gerência também destacam-se funções como o gerenciador de recursos-**RM** , parâmetro de controle de utilização-**UPC** e o algoritmo genérico de taxa de células-**GCRA** ; que serão abordados mais adiante.

Deve-se observar que, conforme visto até o presente que, em redes ATM, devido a peculiaridades inerentes à sua constituição e operação, a compreensão dos mecanismos de operação tem de ser feita partindo de uma visão geral para uma visão particular

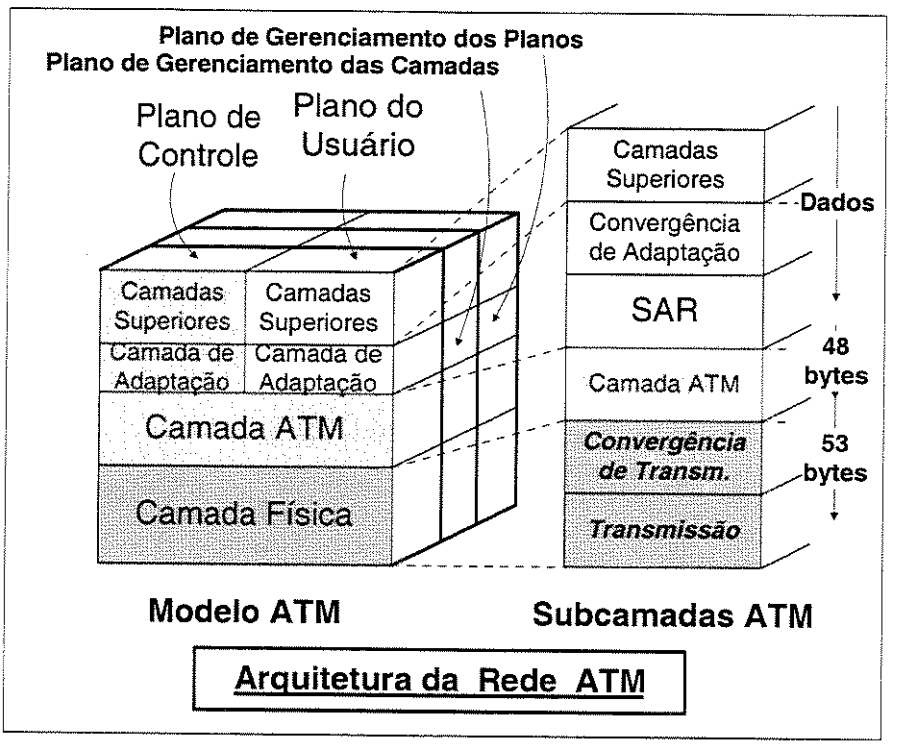

Figura 1.13: Arquitetura das Redes ATM em camadas funcionais

envolvendo estes mecanismos. O CAC, portanto, como parte integrante do controle de congestionamento de tráfego em redes ATM, é um desses mecanismos.

Então, após a camada ATM ter dado formação à célula ATM, esta é liberada à camada física, podendo constituir fontes com diversos tipos de tráfego **por serviço ATM** (denominadas transferências de capacidades ATM-*ATC's*), a saber:

CBR-Requer que uma taxa fixa de dados sendo mantida pelo provedor ATM (estipulado previamente no contrato e através da ação da camada de controle ATM à camada ATM, propriamente dita [15]). O ITU-T [1] a denomina de DBR. Esta é a mais simples das "transferências de capacidades" em ATM e a caracterização do tráfego é a PCR e sua correlata CDVT, definidas em QoS, sendo que este parâmetro pode ser alterado usando sinalização, enquanto a conexão está em andamento, como exemplo, vide aplicação para sinal de vídeo entrante em uma UNI (que, por normatização, tem de chegar ao usuário que se utiliza de tráfego CBR), na Figura 1.15.

VBR (ATM-FORUM [15]), ou *SBR* (ITU-T [1]), esta ATC é a do tráfego que varia sua taxa de geração de células na camada ATM, conforme descrita por três parâmetros de tráfego: PCR, SCR e IBT (vide QoS), sendo que os dois últimos foram definidos em função do algoritmo genérico de geração de células-GCRA. O ITU-T [1] e o ATM-FORUM [15] distinguem ainda as capacidades VBR1, VBR2 e VBR3 dependendo do tráfego ser em tempo real (interativo) ou não, e das classes de QoS empregadas. A utilização dos parâmetros SCR e IBT pode ser por duas vias: como limites definindo

PARÂMETRO / CLASSE	A	B	C	D
Relação Temporal entre a Fonte e o Destino	Requerido		Não Requerido	
Taxa de Bit	Constante	Variável		
Modo de Conexão	Orientado à Conexão			Não Orientado à Conexão
Protocolo de AAL (Camada de adaptação ATM)	Tipo 1	Tipo 2	Tipo 3 / 4	
			Tipo 5	

Figura 1.14: Classes de Serviços ATM

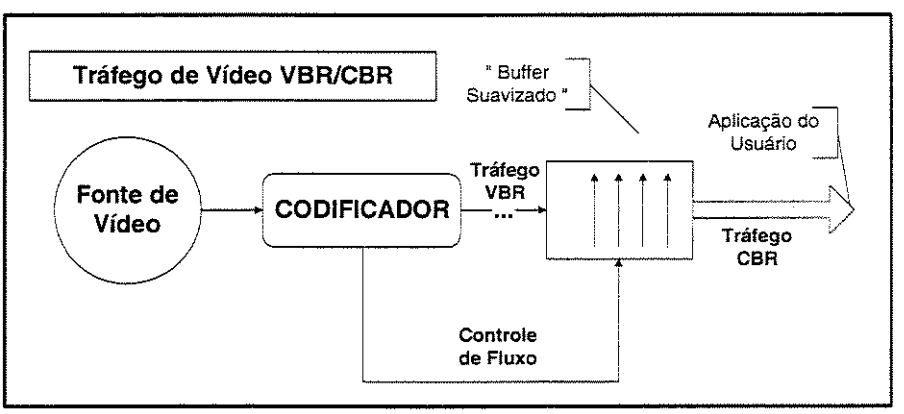

Figura 1.15: Tráfego de Vídeo CBR para o Usuário

um "invólucro" em que o tráfego será sempre enquadrado e formatado ou como uma descrição aproximada das características estatísticas de um tráfego VBR, enquanto ele ocorre. Tripathi, em [16], demonstra a boa eficiência de um mecanismo de planejamento VBR para diferentes taxas incluindo diferentes serviços, como tráfego de vídeo e outros.

ABT, é padronizada somente pelo ITU-T. Neste serviço o próprio usuário é habilitado a definir e controlar uma estrutura de bloco em seu fluxo de dados. O parâmetro negociado para cada bloco é o PCR, pela taxa média extraída do gerenciador de recursos-RM. Geralmente é utilizado no protocolo de reserva rápida, conforme apresentado na descrição dos CAC's (detalhes em [1] [5]).

ABR, é a capacidade de transferência visando evitar congestionamento por um controle preventivo implementado, ou seja, um controle reativo por meio do qual a taxa em que o usuário pode transmitir é dinamicamente ajustada pela rede ATM. O objetivo é a utilização de toda a capacidade da rede atendendo o maior número de usuários, uma vez que estes tenham optado por uma taxa "elástica" de bits. Esta taxa de conexão não é uma característica intrínseca e pode ser delimitada por um máximo e um mínimo valor. O ATM-FORUM [15], e autores como Bonomi e Fendick [17], especificam este mecanismo com mais detalhes, sendo que, em síntese, dois métodos são utilizados para determinar a taxa alocada para uma dada conexão: a taxa é determinada pela

rede, para o usuário por um "bit de indicação de congestionamento" que instrui este a aumentar ou diminuir sua taxa de acordo com um algoritmo definido; ou a taxa é calculada pela rede (com o máxima taxa permitida nos enlaces que constituem o caminho da conexão) e explicitamente comunicado para o usuário. Ajustando as taxas de conexão nos limites dos nós ATM, evita-se a perda de células. Hong & Suda em [18] analisam os efeitos deste tráfego coexistindo com outros tipos de tráfego, tais como, tráfego não-ABR, TCP sobre ATM, e o tráfego denominado interferente.

UBR, aplica-se à conexões em que os parâmetros de tráfego não são declarados formalmente e, portanto, sem garantias QoS. O senso comum é que os próprios usuários implementem controles reativos por ação de protocolos de altas camadas (ex. TCP) ou por outros meios (abordagem detalhada em [4]).

É necessário entender como se comporta o tráfego em uma conexão, explorando os *parâmetros de tráfego* que originaram os fatores de QoS e que medirão qualitativamente este tráfego. O primeiro, **CTD**, é o tempo transcorrido entre dois eventos de célula, ou seja: "CTD refere-se ao tempo entre a transmissão do último bit de uma célula da fonte UNI e a recepção do primeiro bit desta célula à UNI de destino" [22].

Em termos gerais, CTD é uma variável que tipicamente tem a distribuição de probabilidade como na Figura 1.16.

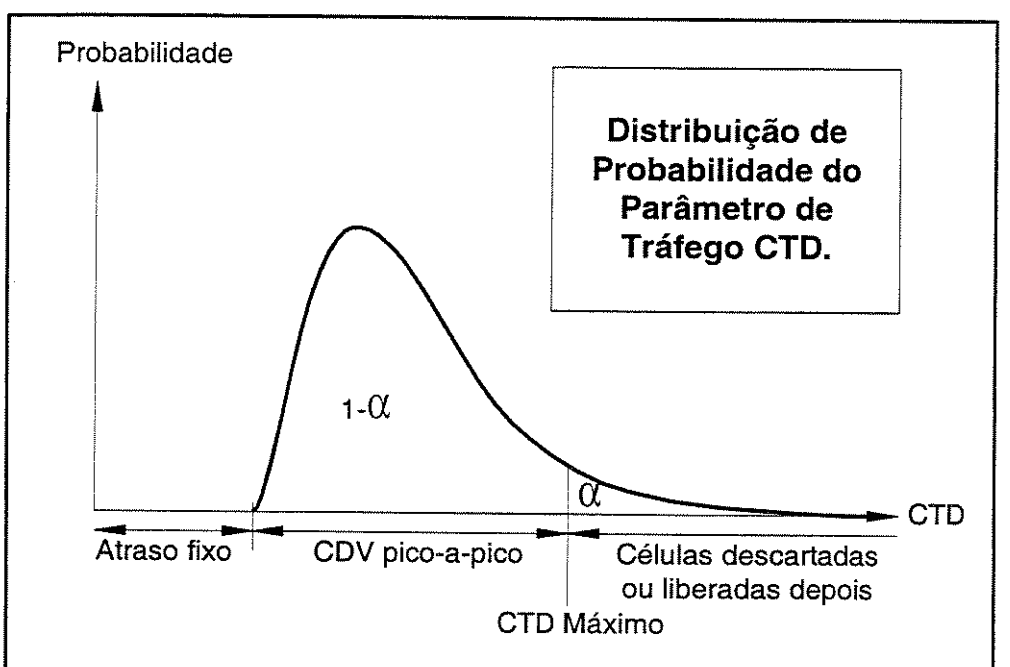

Figura 1.16: Distribuição de probabilidades CTD

Como pode ser visto, existe um atraso mínimo, ou atraso fixo-FD, que inclui atraso de propagação pelo meio físico, pelo sistema de transmissão e componentes fixos de atrasos em comutadores. A parte variável do CTD é o **CDV** e é devido ao gerenciamento do fluxo de células pela "bufferização" nos comutadores e multiplexers.

Na Figura, *maxCTD* define o limite máximo de atraso para uma determinada conexão. A fração (α) de todas as células que excedem este limiar serão descartadas ou liberadas atrasadas e a porção remanescente ($1-\alpha$) está dentro do limite requerido QoS (por definição).

Analisa-se a seguir, como parâmetros de tráfego, as causas do *atraso fixo* e do *CDV*.

O atraso fixo-*FD*, ocorre na camada física, onde existem variadas contribuições para atraso do "trem" de células. De Prycker em [3] as contabiliza como na Figura 1.17

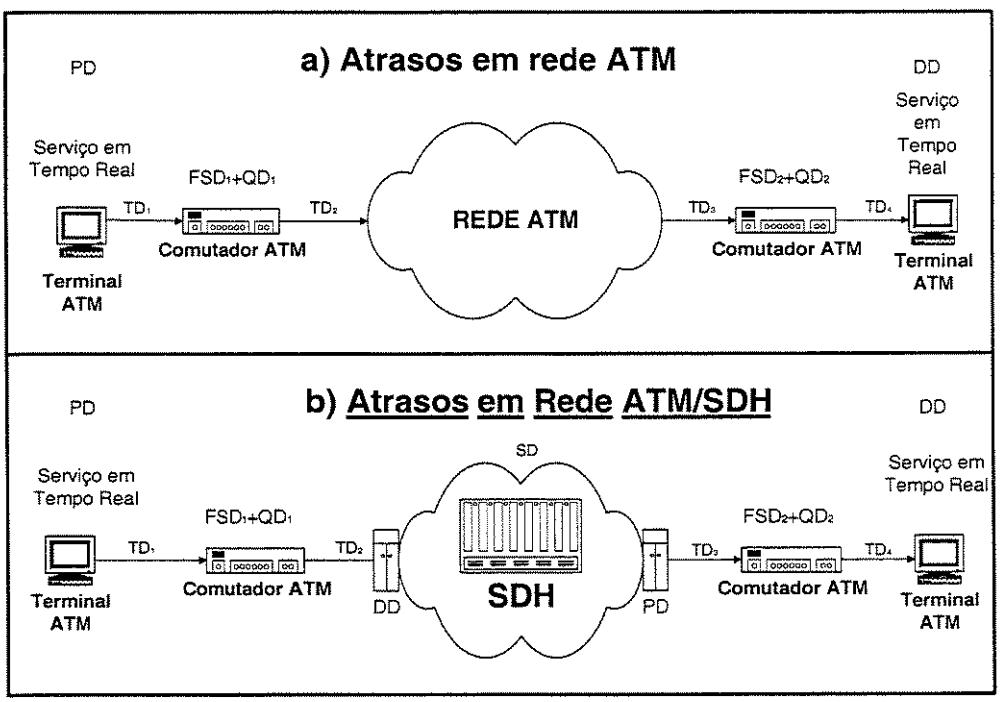

Figura 1.17: Atrasos de célula ATM em redes ATM(a) e redes mistas ATM/SDH(b)

Em que o:

Atraso de transmissão *TD* independe do modo de transferência utilizado e depende da distância fim-a-fim do meio de transmissão, somadas as partes (valor típico de 4 e 5 μseg, por Km);

Atraso de empacotamento *PD* existe toda vez que serviços de dados são convertidos em pacotes ou em interfaces não-ATM;

Atraso de comutação, *SD=FSD + QD* onde o atraso fixo de comutação **FSD** é devido ao hardware do comutador e o atraso de fila **QD** decorre de que os sistemas ATM são estatisticamente multiplexados e comutados e então filas são necessárias para evitar excessivas perdas de pacotes. Este QD varia com a carga da rede e seu comportamento é caracterizado pela distribuição de probabilidades das filas nos "buffers". É responsável, juntamente com DD, a seguir, pela parte variável dos atrasos.

Atraso de desempacotamento *DD* (adiante tambem denominado $D(i)$) é

mais um artifício utilizado pela rede ATM que uma característica, não sendo intrínseco ao meio. Ao desempacotar os bytes no destino, este atraso é estimado e adicionado, para aliviar o atraso QD (estocástico) e reconstruir, com segurança e em seqüencia, o original trem de bits.

Atraso de mudança síncrona *SD* é o atraso verificado quando a célula ATM se insere em um quadro SDH somado ao atraso de desinserção desta do quadro SDH à rede ATM.

Estes atrasos são, no primeiro caso:

$$(FD)_1 = \sum_i (TD)_i + \sum_j (FSD)_j + (PD) \tag{1.1}$$

E, no segundo caso:

$$(FD)_2 = \sum_i (TD)_i + \sum_j (FSD)_j + k \times (PD) + \sum_l (SD)_l \tag{1.2}$$

Nestas expressões, i indica o número de enlaces de transmissão, j o número de comutadores ATM, k o número de pares encapsuladores/desencapsuladores entre as partes ATM e não-ATM da rede incluindo os terminais, e l o número de inserções/desinserções da célula ATM na rede síncrona (para o caso heterogêneo [3]). Na parte variável da distribuição está o principal: o CDV. A seguir analisamos as causas principais do CDV, deixando as causas aleatórias para serem abordadas nos modelos de filas para tráfego ATM.

Na parte variável da distribuição de atraso, o **CDV na UNI** é um evento que ocorre devido à células de uma mesma conexão em uma UNI da rede ATM, e podem ser sinais de voz e vídeo digitalizados e transmitidos como um "trem" de células ATM. O requisito para estes serviços é que o atraso de célula seja pequeno ou desprezível, para que não surjam efeitos indesejáveis na recepção, tais como: defasagem de voz e imagem, eco, etc. Como, na prática, o tráfego ATM é sempre composto (a menos de uma rede corporativa dedicada), este se torna um requisito geral.

Na definição de ATM, esta foi projetada para minimizar o processamento na transmissão, devido ao seu reduzido cabeçalho para que a comutação e roteamento sejam os mais rápidos possíveis. Outro requisito para este serviço é que a taxa de liberação de células para o usuário de destino (recepção) seja constante (CBR mesmo que, intermediariamente, o tráfego seja VBR nos comutadores e MUX's). Por isso, é inevitável que surja alguma variação nesta taxa de entrega interfaceando a rede e o usuário (UNI) e, utilizando a Figura 1.18, pode-se descrever as causas deste atraso.

$D(i)$ representa o atraso fim-a-fim imputado à i-ésima célula (anteriormente denominado DD). Este CDV, **em uma mesma conexão**, ocorre porque o sistema de destino desconhece a exatidão do atraso, ou seja, a célula não leva esta informação em seu conteúdo, sendo por isto, impossível o sincronismo entre a fonte e o destino. Quando a primeira célula chega no tempo $t(0)$ o usuário-alvo atrasa a célula em um tempo adicional de $V(0)$ para liberar a aplicação. $V(0)$ é uma estimativa da variação de atraso da

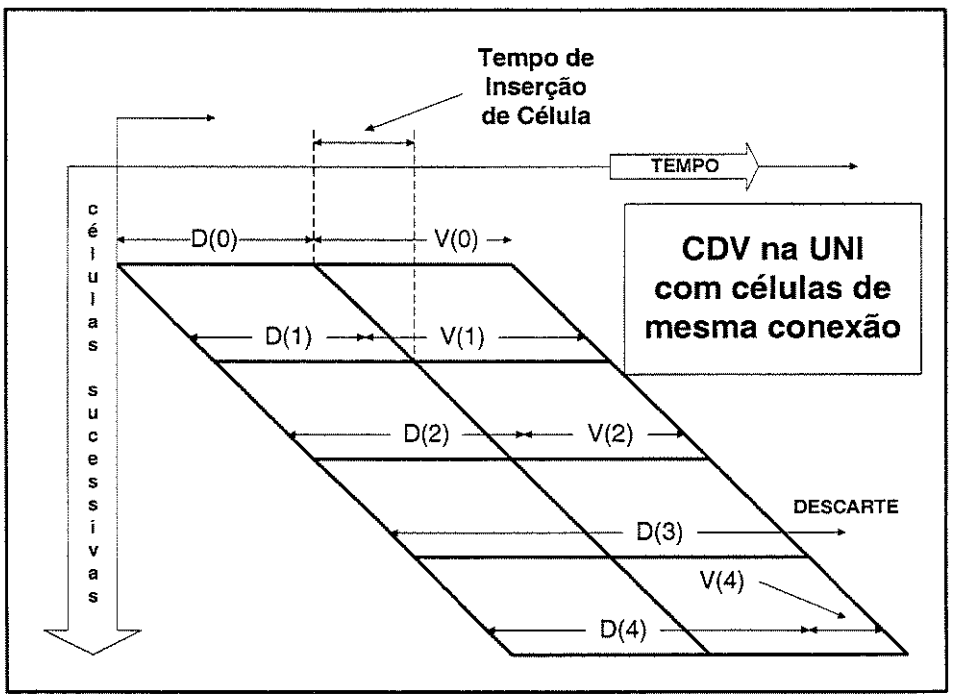

Figura 1.18: CDV na UNI com células de mesma conexão

célula que a aplicação pode tolerar [15]. Células subseqüentes são atrasadas e liberadas para uso à taxa de R células por segundo. O tempo entre os inícios de liberação de células será:

$$\delta = \frac{1}{R} \tag{1.3}$$

e para realizar a taxa constante, a próxima célula é atrasada em $V(1)$ tal que satisfaça:

$$t(1) + V(1) = t(0) + V(0) + \delta$$

$$V(1) = V(0) - [t(1) - (t(0) + \delta)]$$

e generalizando:

$$V(i) = V(0) - [t(i) - (t(0) + i.\delta)]$$

ou:

$$V(i) = V(i-1) - \{t(i) - [t(i-1) + \delta]\} \tag{1.4}$$

Se ocorrer $V(i) < 0$ a célula será descartada e perdida. O resultado é que os dados são liberados à camada superior à taxa constante de bits com eventuais lacunas devido às células descartadas. O valor de $V(0)$, que é atraso médio aplicado à todas as células que chegam em início de conexão, é função de uma previsão de CDV (vide normas em

[15] [19], pois para minimizar o CDV o assinante pode estabelecê-lo no contrato com o provedor da rede).

Outra ocorrência de CDV na UNI é devido às **células de conexões superpostas**, ou seja, se uma aplicação gera dados para transmissão à taxa constante de bits, CDV pode ocorrer abrangendo três camadas do modelo ATM. A Figura 1.19 ilustra as causas potenciais deste CDV, que ocorre na UNI [1].

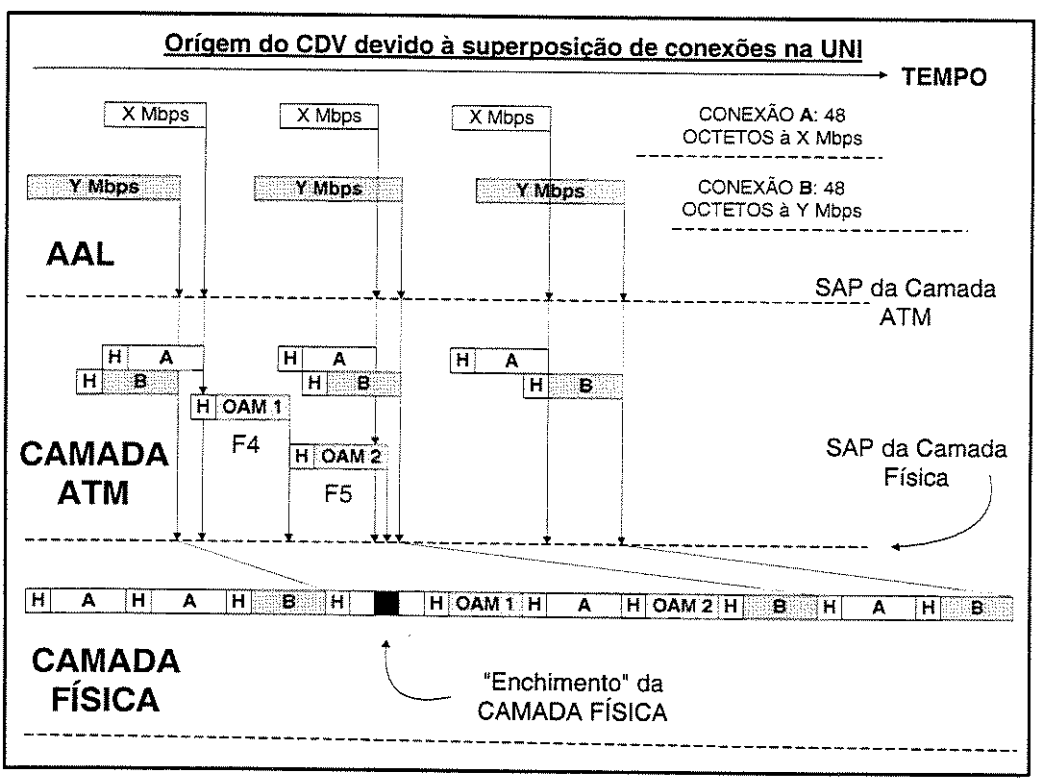

Figura 1.19: CDV devido à superposição de conexões na UNI

Neste, as conexões ATM A e B suportam usuários com taxas de X e Y Mbps, respectivamente. Na AAL, os dados provenientes das camadas superiores são segmentados em blocos de 48 bytes (octetos). Note-se que, no diagrama de tempo, os blocos aparecem de diferentes tamanhos para as duas conexões, isto porque, devido às taxas serem diferentes, os tempos requeridos para a geração do bloco de 48 octetos também são diferentes, ou seja:

$$\text{Conexão A} \quad : \quad \frac{48 \text{ bytes} \times 8 \text{ bits/byte}}{X \text{ bits/seg.}} = \frac{384}{X} \text{ segs.}$$

$$\text{Conexão B} \quad : \quad \frac{48 \text{ bytes} \times 8 \text{ bits/byte}}{Y \text{ bits/seg.}} = \frac{384}{Y} \text{ segs.}$$

A partir do SAP da camada ATM, os blocos ganham o cabeçalho, resultando em células de 53 bytes. Estas precisam ser intercaladas e liberadas para a camada física,

que as transmite à taxa de dados de transmissão da camada física. O "delay" de célula é introduzido neste processo de intercalação de blocos para a camada física (no SAP da camada física) ou seja, neste caso, se duas células de diferentes conexões chegam em uma camada ATM em tempos superpostos, uma das células precisa ser atrasada. Em adição, na camada ATM são introduzidas células OAM (gerenciamento de operação e manutenção), que também precisam ser intercaladas com as células do usuário [19].

Em Onvural [4], define-se o CDV concordando com a definição genérica em CTD, como sendo o atraso fim-a-fim da i-ésima célula ou $D + W_a$, onde $D = D_n + D'$, é uma constante que inclui o atraso D_n devido à transmissão e o atraso D de célula devido a uma conexão e a quantas outras estiverem superpostas. W_a é o componente de atraso aleatório que surge da "bufferização" (multiplex estatístico, distribuição de probabilidades nas filas dos comutadores, etc) da rede. Daí, o tempo entre chegadas de células será:

$$(D + W_{a+1}) - (D + W_a) = \delta \tag{1.5}$$

O tempo entre chegadas de células seria nulo caso $W_{a+1} = W_a$, entretanto, de acordo com a já citada aleatoriedade da rede, W_a é uma variável aleatória e não constante. Dentre as diversas definições de CDV usadas na literatura, além das que fornecemos, outras são:

- Variância do atraso de transmissão de conexão:

$$CDV = E\{(W_a - E[W_a])^2\} \tag{1.6}$$

- Diferença entre os valores de atraso de trânsito de uma conexão:

$$CDV = W_{a+1} - W_a \tag{1.7}$$

- Variância instantânea da média, ou probabilidade que a variação seja maior que w,estipulado no contrato:

$$CDV = Pr\{(W_{a+1} - E[W_a]) < w\} \tag{1.8}$$

O ITU-T [1] fornece alguns limites práticos para o CDV e o CDVT que estão na tabela da Figura 1.20.

A *Qualidade de Serviço-QoS* da camada ATM é uma supervisão da camada de transporte da rede e é medida por um conjunto de parâmetros derivados dos parâmetros de tráfego já apresentados, caracterizando a performance de uma conexão na camada ATM.

Estas medições quantificam a qualidade fim-a-fim da camada. Para os objetivos de análise de performance de redes ATM, de seis a oito parâmetros são especificados por [15]. Três a cinco destes são negociados entre os sistemas-fim e a rede, três obrigatórios e dois opcionais:

Os seguintes parâmetros **são** negociados:

SERVIÇOS	CDV (ms)	CDVT ("Jitter") (ms)
Vídeo-conferência 64 Kbps	300	130
Vídeo NTSC MPEG 1.5 Mbps	5	6.5
Vídeo HDTV 20 Mbps	0.8	1
Voz (Compressão de:) à 16 Kbps	30	130
Vóz MPEG à 256 Kbps	7	9.1

Figura 1.20: Valores ITU para CDV e CDVT

O **CDVT** - "τ".que é derivado e não pode ser confundido com o parâmetro *CDV que geralmente é negociado durante o estabelecimento da conexão* (estabelecimento do tráfego em conexões virtuais comutadas), enquanto que o *CDVT é negociado previamente em contrato e, a partir daí fixado na UNI pelo provedor da rede ATM.*

O tráfego-fonte tem que se ajustar ao CDVT para validar as garantias QoS com o estabelecimento da conexão e isto é checado pela UPC descrito adiante em controle de tráfego. A variação de atraso configurado pelo CDVT é a diferença entre o melhor e o pior caso esperado para o atraso de transferência fim-a-fim de uma célula. Recorramos novamente à Figura 1.14 verificando que o melhor caso é o atraso fixo e o pior é igual ao maxCDV (resultado da inclusão dos fatores aleatórios da rede ATM), sendo que valores típicos podem ser examinados na Figura 1.20.

Em De Prycker [3] ressalta-se que os efeitos aleatórios no intervalo de tempo entre as chegadas de célula nos VPC / VCC, podem ser monitoradas pelo algoritmo genérico de taxa de célula (**GCRA**) de uma UNI ou NNI.

As funções de parâmetro de controle de uso (**UPC**) e parâmetro de controle de rede (**NPC**) não podem somente confiar na taxa de pico de célula PCR para avaliação do tráfego da rede. Definiu-se portanto, a tolerância τ para o CDV tornando o algoritmo de taxa de célula função de dois parâmetros $GCRA(T, \tau)$, onde $T = 1/R_p$ (inverso da PCR (R_p) -taxa de pico de células). O valor de τ seria medido de um tráfego passado e usado para o cálculo do número de células a serem produzidas pelo GCRA, no mesmo intervalo de tempo [4], mas no instante seguinte, e é dada por:

$$N = [1 + \tau/(T - \delta)] \qquad T > \delta,\ \tau \geqslant T - \delta \tag{1.9}$$

Para o caso de tráfego à taxa de bit variável (VBR), utiliza-se dois outros parâmetros para auxiliar a alocação de recursos com mais eficiência, taxa sustentável de célula (**SCR**) e a tolerância de surtos τ_s.

O SCR define o limite superior sobre a taxa média de uma conexão ATM, calculada sobre uma escala de tempo alta em relação à T_{min}. Ela é fundamental para caracterizar a fonte VBR. Este parâmetro habilita a rede a alocar eficientemente recursos para um número de fontes VBR sem dedicar a quantidade da que seria, se fosse o caso

de taxa constante à PCR. O SCR é utilizado somente se $SCR < PCR$ (veja algoritmo "Leaky Bucket", adiante).

Obs.: Para efeito de consulta extra, alguns valores de τ, SCR, e τ_s tolerados em uma UNI baseado em medições de acordo com o ITU-T [1] são mostrados em Black [20], capítulo 10.

Os dois últimos parâmetros são opcionais, mas não menos importantes, pois determinam o número de células que constituem o tamanho máximo de surto (**MBS**) e o **maxCTD** este último, já abordado em parâmetros de tráfego (vide Figura 1.14), representa a soma do atraso fixo e CDV.

O *MBS* é o máximo número de células que pode ser enviada continuamente **durante a ocorrência** do PCR. Se as células são apresentadas à rede e agrupadas no formato apropriado de MBS, então as lacunas inativas entre os agrupamentos têm que ser suficientes para que a taxa no montante não exceda o SCR. Tanto o SCR quanto o MBS são mandatórios para fontes VBR.

$$MBS = [1 + \tau_s/(T_s - T)] \qquad T_s > T, \text{ e } \tau_s \geqslant T_s - T \tag{1.10}$$

A Figura 1.21 mostra a relação existente entre estes parâmetros de tráfego.

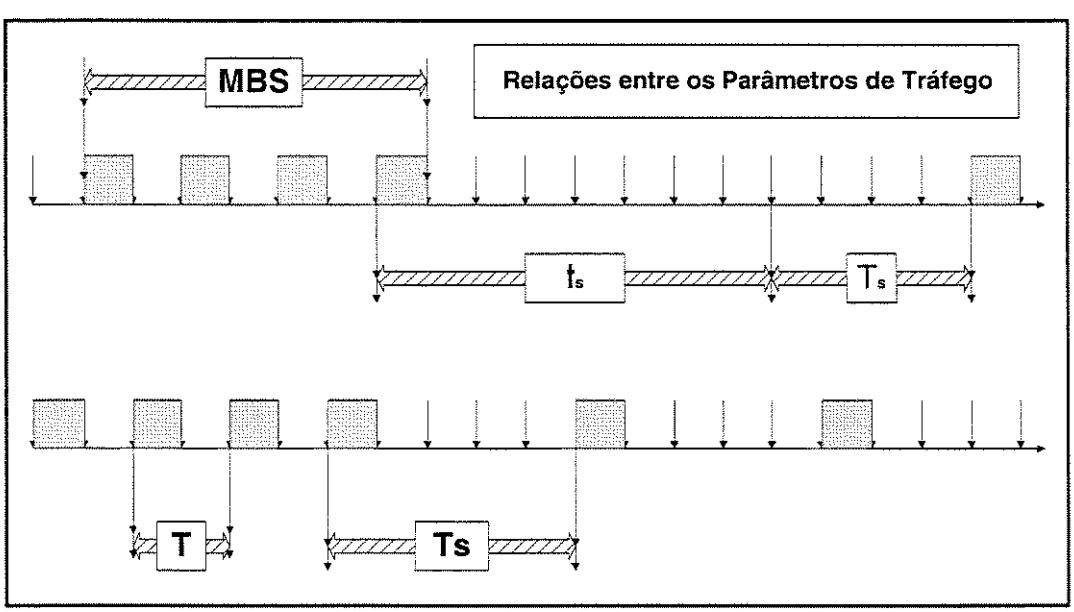

Figura 1.21: Relações entre parâmetros descritores de tráfego ATM

Já o *MCR* é especificado para o serviço ABR. Ele define o mínimo compromisso requerido à rede pela fonte ABR em taxa de células. O valor "0" pode ser usado. O objetivo do serviço ABR é prover rápido acesso à capacidade da rede, que não seria utilizada no serviço CBR, sempre que se puder contar com uma avaliação desta capacidade pela própria rede. A quantidade $[(PCR) - (MCR)]$ representa um componente elástico do fluxo de dados para que a rede forneça a margem de segurança que esta capacidade

terá ao compartilhar vários fluxos de fontes ABR. Este valor pode variar entre zero e PCR e pode assumir valores diferentes para direções diferentes de VCC's (ver detalhes em Onvural & Cherukuri [19]).

E o CLR é definido para uma conexão como:

$$CLR = \frac{\text{Células Perdidas}}{\text{Total de Células Transmitidas}} = CLP \tag{1.11}$$

Obs.: Em dimensionamento de redes ATM, o CLR é comumente tratado como CLP-Cell Loss Probability que, na prática, tem o mesmo sentido que CLR. Para diferenciar do bit CLP (bit de prioridade de célula-vide célula ATM), designaremos este sempre por *"bit CLP"* e o índice de probabilidade de perda de célula por *"CLP"*.

Então, esta definição exemplificada sob a visão de projeto é:

A probabilidade de perda de célula-CLP em filas é a quantidade de células perdidas no decorrer de uma transmissão longa. Dado um "buffer" de tamanho M, este parâmetro tem comportamento distinto em duas situações, [34]:

- Quando o número de células em espera e em conexão, num determinado intervalo de tempo é menor ou coincide com o tamanho do "buffer" ou ;

- Se este número for maior que o tamanho do "buffer", causando transbordamento do mesmo.

Definindo a utilização do "buffer" como sendo a multiplicação da velocidade com que as células entram na rede e o tempo médio de serviço na rede, exemplifica-se resumidamente duas situações para o caso de uma fila M/M/1 com tempos interchegadas $(1/\lambda)$ e de serviço $(1/\mu)$ distribuídos exponencialmente;

Sendo:

q : número de células em espera e em conexão;

M : tamanho do "buffer" (em células);

λ : número médio de células entrantes/tempo;

s : tempo médio de conexão;

ρ : fator de utilização do "buffer";

Então, para duas situações, mostra-se que [34]:

$$CLP(q \leqslant M) \leqslant CLP\,(q = M) = (1 - \rho)\,\rho^M \tag{1.12}$$

$$CLP\,(q > M) = \rho^{(M+1)} \tag{1.13}$$

sendo:

$$\rho = \lambda.s \tag{1.14}$$

Observamos que, em se transbordando o "buffer" $(q > M)$, $\rho > 1$, a perda de células é bem maior que o caso anterior. Vários autores desenvolveram técnicas de projeto em redes ATM envolvendo o conceito e nomenclatura de CLP, dentre eles Virtamo em [5], Onvural em [4] e Robertazzi em [21].

Continuando depois deste "parêntesis", os seguintes parâmetros **não são** negociados:

O *CER* é definida para uma conexão como:

$$CER = \frac{\text{Células Erradas}}{\text{Células Transferidas Corretamente} + \text{Células Erradas}} \tag{1.15}$$

O *SECBR* - Severely-Errored Cell Block Ratio

$$SECBR = \frac{\text{Blocos de Células Severamente Erradas}}{\text{Total de Blocos de Células Transmitidas}} \tag{1.16}$$

Como um bloco de células é uma seqüência de N células transmitidas consecutivamente em uma dada conexão, um bloco de células severamente errado ocorre quando mais de M células erradas, células perdidas ou células mini-inseridas são observadas no bloco de células recebidas.

O *CMR* - Cell Mininsertion Rate:

$$CMR = \frac{\text{Células Mini- Inseridas}}{\text{Intervalo de Tempo}} \tag{1.17}$$

É causado por erro de cabeçalho em células que são inseridas em blocos de outras conexões.

E assim, com estas definições, o próximo capítulo tratará dos conceitos sobre controle de congestionamento, policiamento e modelos de tráfego em redes ATM;

Capítulo 2

Congestionamento, policiamento e tráfego ATM

2.1. Controle de congestionamento e policiamento de tráfego ATM

As técnicas de controle e congestionamento são de vital importância para a operacionalização de redes ATM. Sem tais técnicas, o tráfego proveniente do nó do usuário pode exceder a capacidade da rede causando transbordamento nos "buffers" dos comutadores ATM e, conseqüentemente, perda de dados e informações.

Devido a alta velocidade e o tamanho das células ser pequeno, as redes ATM apresentam maior dificuldade de controlar efetivamente seu fluxo. O limitado número de bits inseríveis no cabeçalho para se compor este controle dificulta a solução da questão e, por isto, o controle de congestionamento é tema de pesquisa corrente [22].

O ITU-T [1] tem especificado, através da norma I.371, instruções com mecanismos simplificados para assegurar uma eficiência de funcionamento dentro de requisitos de qualidade estabelecidos previamente no **contrato de prestação de serviços** (vide QoS).

O ATM-FORUM [15] também detalhou esta forma de controle pelas normas "Traffic Management V.4" (96) e seu "Addendum for ABR" (97), ressaltando o compromisso do **contrato de serviço** com parâmetros de QoS.

Os controles de tráfego originalmente empregados para outras redes são inadequados para redes ATM, pelos seguintes motivos:

A-O tráfego ATM possui componentes não submissos ao controle de fluxo tradicional. Por ex., tráfego de voz e vídeo não podem impedir suas fontes de gerar células quando a rede está em "overflow".

B-Qualquer realimentação para se reduzir abruptamente o tempo de transmissão de célula é lenta comparada à propagação de "delay" nas redes ATM.

C-Redes ATM tipicamente suportam uma larga gama de aplicações requerendo, em boa parte do tempo, uma variação de banda de alguns kilobits por segundo a várias centenas de megabits por segundo.

D - Da mesma forma, a diversidade de aplicações em redes ATM podem gerar variadas configurações de tráfego (ex. fontes CBR compostas com fontes VBR, etc).

E - Em seqüência, para redes ATM, diferentes aplicações requerem diferentes serviços (por ex. fontes de voz e vídeo, sensíveis à "delay", requerem serviços diferentes de uma fonte exclusiva de dados).

F - Finalmente, a geralmente elevadíssima velocidade de comutação e transmissão resultam que as redes ATM são bem mais "voláteis" ao controle de tráfego.

Com base no exposto, o ATM FORUM tem delineado o número de parâmetros descritores de tráfego já citados, que caracterizam as variáveis de controle de um fluxo de células em uma conexão ATM. Este tráfego precisa ser visualizado sob diferentes perspectivas.

Primeiro, deve-se considerar a natureza intrínseca do tráfego gerado por uma determinada fonte em particular e submetido à rede através da UNI.

Segundo, este fluxo de células pode ser modificado dentro da própria rede, pelas conexões ATM, pela variabilidade de "delays" ao longo das mesmas e pelo tratamento das células que não se enquadram no modelo de tráfego esperado da fonte. Para cada uma destas abordagens são usados seus próprios descritores de tráfego de fonte e de conexões.

Uma importante função de gerenciamento de tráfego para as redes ATM é o *RM*, que é mais genérico e tem como objetivo prover um nível aceitável de conexões de blocos de células. Os requisitos de tráfego determinam a topologia da rede ATM, o número de enlaces, suas BW´s e dos comutadores com seus nós de acesso. Além disto, o número de usuários e o montante de tráfego gerado e os tipos de aplicações utilizadas são dinâmicas e não determinísticas.

Com isto, os VP's são conexões temporárias com BW alocadas deterministicamente por este provisionamento de recursos (RM), lembrando que dentro de um VP existem vários circuitos virtuais (CV´s) com suas conexões e BW´s sendo alocadas via CAC's. Se um VP, por exemplo, entre dois comutadores estiver subutilizado, esta sobra de BW_{PV} não poderá ser utilizada em outros VP´s e a rede estará desperdiçando recursos, mesmo que os melhores CAC´s estejam sendo empregados nos CV´s.

Este tipo de gerenciamento de VP é sustentado por uma grande gama de informações sobre a rede e atuando, como já citado, na camada de gerenciamento ATM pelas células OAM devido principalmente ao fato de que as alterações na rede ATM se processam rapidamente. A questão de se definir o conjunto de parâmetros para uma dada topologia de rede também é objeto de pesquisa corrente. Depois que o CAC atua sobre os VC´s entra em cena o controle de parâmetro de uso (*UPC*) para verificar se os recursos cedidos foram excedidos e isto é feito através do controle do PCR e SCR.

A rede ATM apresenta, além do RM, outras funções de gerenciamento de rede, dentre as quais o próprio CAC, conforme Onvural [4] classificou pela Figura 2.1, cada uma destas atuando em sua escala de tempo, sendo que a função CAC atua em duração

de conexão. Além destas, a formatação de tráfego e o policiamento de tráfego são também empregados e atuam em nível de célula.

O **policiamento de tráfego** atua de acordo com as especificações de conformidade ou seja, dados os parâmetros de tráfego de fonte, a rede regula o tráfego usando o GCRA, que define em uma maneira operacional a relação entre os parâmetros de tráfego e cada célula que chega como de ou não conformidade.

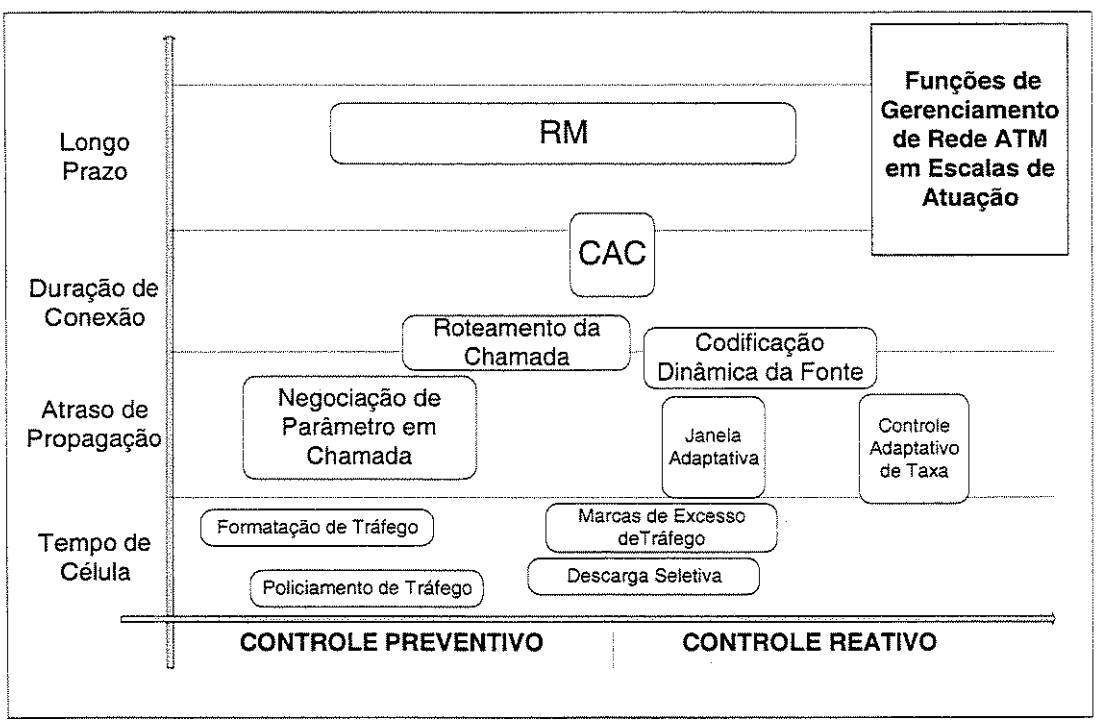

Figura 2.1: Funções de Gerência de Fluxo de Tráfego em redes ATM

O *GCRA* é um algoritmo que confere todas as células para ver se há conformidade com os parâmetros requeridos de QoS para um circuito virtual. O GCRA tem dois parâmetros: $PCR(Rp)$ e τ $(CDVT)$. A recíproca de PCR, $T = 1/Rp$, é mostrada na Figura 2.2(a).

Se um cliente, por exemplo, não envia mais que 100.000 células/seg, então $T = 10$ μseg. No caso máximo, uma célula chega prontamente a cada 10 μseg. A um cliente é sempre permitido espaçar células consecutivas maior que T, como mostrado na parte (b) e estas, por definição, estão em conformidade (Tanenbaum em [9]).

O problema surge quando clientes enviam células mais rápidas, como mostram as partes (c) e (d). Se a célula chega um pouco antes (na ordem de $t_1 + T - L$) isto está em conformidade, mas a próxima célula é esperada à $t_1 + 2T$ (e não a $t_2 + T$), para prevenir que o emitente transmita célula L μseg. adiantado, aumentando ilegalmente o seu PCR.

Se a célula chega mais que L μseg adiantada, ela é declarada em não conformidade.

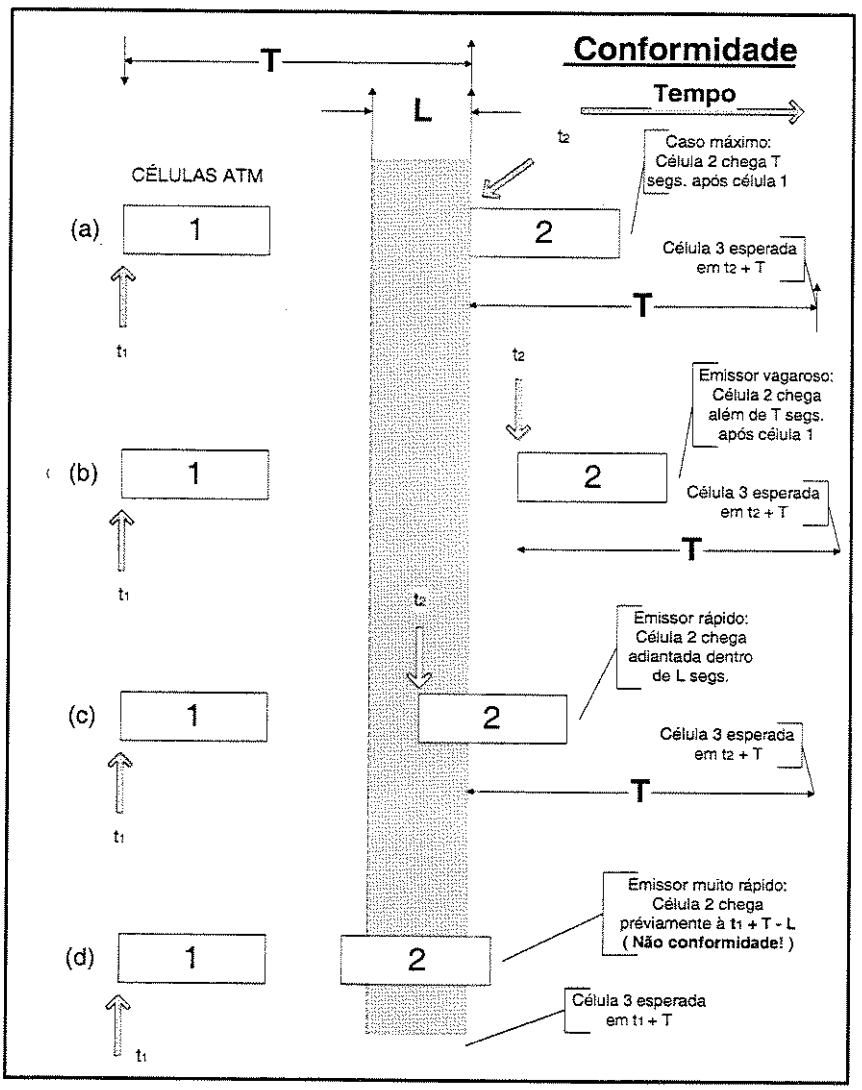

Figura 2.2: Conformidade de Célula ATM

O tratamento de não conformidade está reservado ao provedor da rede ATM e alguns provedores simplesmente a descartam, outros podem armazená-la, alterando o bit CLP para marcá-lo como baixa prioridade. Assim que estes chegam aos comutadores ATM podem ser descartados na eminência de congestionamento..

Se o bit CLP já estiver como prioritário, ele permanece como tal, não sendo descartado. Na Figura 2.3, mostra-se o que acontece quando um usuário tenta burlar a rede com um pequeno bit.

Em vez de aguardar o transcorrer do tempo T para transmitir a célula 2, este usuário transmite (ilicitamente) um pequeno bit antes ao tempo $T - \varepsilon$ onde, digamos $\varepsilon = 0.3T$, como exemplo. A célula será aceita sem problemas. É enviada então a célula 3, novamente à $T - \varepsilon$ após a célula anterior, isto é, a $T - 2\varepsilon$ da primeira. Novamente é aceita.

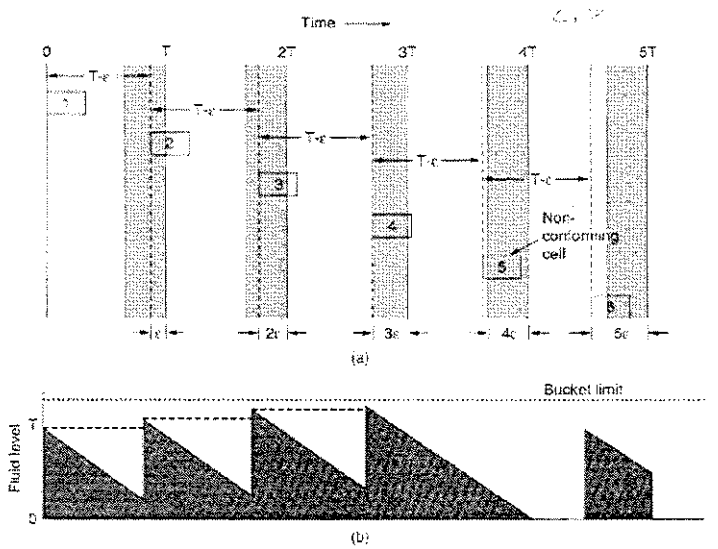

Figura 2.3: Conformidade com "Balde Furado"

Entretanto, toda célula sucessiva acumula a aproximação com o limite $T - L$ e quando a célula 5 chega a $T-4\varepsilon = T-1.2T = -0.2T$, é considerada de não conformidade e tratada como tal.

O GCRA **policia** esta não-conformidade e é conhecido também como **algoritmo de gerenciamento virtual**, mas também pode ser nomeado como **algoritmo do balde furado** ("Leaky Bucket"). Isto porque, (aproveitando a Figura 2.3), imagine-se um balde com capacidade de T unidades de fluido com um furo que escapa 1 unidade de fluido (por μseg, por exemplo). O fluído que chegar à taxa de 1 unidade a cada T μseg, sempre vai encontrar o balde vazio e escapará pelo furo sem problemas. Ao ocorrer a situação de adiantamento já descrita, o volume do balde se elevará, aproximando-se do volume máximo T e, em conseqüência descartará o fluído que chegar até o furo esvaziar o balde Figura 2.3(b).Onvural [4] apresenta o GCRA na Figura 2.4, já baseado no algoritmo leaky bucket, sendo:

X = valor do contador do balde furado;
LCT = última célula com tempo concordante;
t(k) = tempo de chegada da K-ésima célula;
I = incremento;
L = capacidade (limite) do balde;

Considerando o balde como um contador limitado acima por L. Sobre a chegada da primeira célula, ao tempo t(1), X e LCT são respectivamente inicializados para 0 e t(1). No GCRA, o contador é decrescido de 1 toda unidade de tempo, enquanto o contador cresce por I cada vez que uma célula em conformidade chega. Se, sobre a chegada de uma célula (antes, o contador é acrescido de 1) o valor do contador é menor ou igual que seu limite L, a célula é considerada em conformidade, caso contrário é uma célula em não conformidade. O valor do contador não é atualizado (acrescido) quando

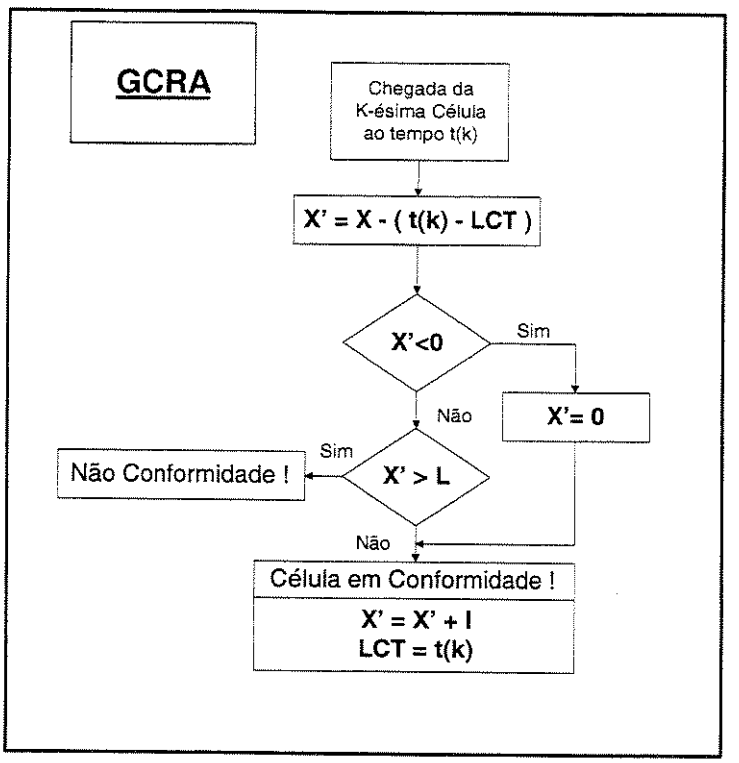

Figura 2.4: Algorítmo Genérico de Taxa de Células (GCRA)

da chegada de uma célula em não conformidade, caso contrário será acrescido de I.

Pela Figura 2.4, ao tempo de chegada da k-ésima célula, o "buffer" é temporariamente atualizado para X', que é igual ao valor do contador após a chegada da última célula em conformidade (X), menos o montante acumulado no balde. O conteúdo do balde é sempre não-negativo. Se X' é menor ou igual a L, a célula está em conformidade e X é "setado" para X' mais I, e LCT é setado para o tempo corrente t(k). Se X' é maior que L, então a célula está em não conformidade e os dois parâmetros X e LCT permanecem imutáveis, conseqüentemente, GCRA depende somente dos parâmetros I e L, ou seja GCRA(I,L).

2.2. Controle de admissão e o tráfego em redes ATM

Controle de Admissão de Chamada (CAC) é um mecanismo que avalia se uma nova chamada ATM que solicita conexão (com largura de banda e performance requeridas) a uma rede ATM, pode ser suportada por esta rede, resultando assim, na sua admissão ou não. O comprimento do "buffer", bem como o número de usuários na rede e a banda suportada por esta são parâmetros fundamentais para o dimensionamento eficaz desse controle.

2.2.1. Modelos de tráfego ATM

Os modelos de tráfego ATM são decisivos para o projeto e análise das redes ATM e seus CAC's devido à variabilidade de tráfego oriundo das fontes de acordo com os serviços prestados. O estudo de tráfego já vem ocorrendo desde o advento dos circuitos telefônicos comutados com a necessidade de controlar o fluxo de tráfego de chamadas telefônicas. Mais tarde, este estudo adaptou-se às redes de dados comutadas virtualmente.

Mas os modelos de tráfego, em qualquer caso, assumem tal importância por causa do problema fundamental das redes, que é a distribuição ótima de recursos aos usuários que as assessem e, conforme dito, a integração de voz, vídeo-pacotes, imagens codificadas e tráfego gerado por computador, cada serviço com seus próprios requerimentos QoS, necessitam de modelos mais sofisticados para projeto e análise.

Assumindo o conhecimento prévio de teoria das filas (várias bibliografias podem ser consultadas, podemos indicar Robertazzi [21], Kleinrock [24] [25], Dshalalow [26] e Gross & Harris [27], como algumas das mais citadas nos trabalhos de redes), foi verificado que o modelo de Poisson, pura e simples usado extensivamente em análise de performance de redes, nem sempre funciona quando se trata de chegadas de tráfego de fontes independentes de vários tipos de tráfego (mais tarde, através da lei dos grandes números, multiplexando vários tipos de fontes independentes chegou-se ao modelo de Poisson composto)

Medidas efetuadas em LAN's e WAN's (INTERNET) indicaram que o modelo de Poisson falha em várias aplicações de situação real (vide Leland & outros em [28] e Paxson & Floyd em [29]). Nestes estudos, aplicando métodos estatísticos nas medidas, verificou-se que tais casos se classificam como tráfego auto-similar ou fractal.

O termo auto-similar significa que a caracterização estatística deste tráfego é essencialmente invariante com a escala de tempo, ou seja, as mesmas propriedades estatísticas são observadas se a escala de tempo muda de centenas de segundo para segundos ou milisegundos. Este fenômeno foi inicialmente descrito por Mandelbrot, que achou este comportamento não somente em séries temporais geradas por tráfego como em muitos fenômenos naturais [30]. Conseqüentemente, o modelamento do tráfego fractal trouxe componentes decisivos para a análise e projeto de redes ATM, além dos modelos tradicionais de Poisson.

Na introdução deste trabalho, em fontes de tráfego ATM, classificam-se os fluídos de tráfego em classes A (voz CBR), B (vídeo VBR fluído) e C/D (dados em surtos variáveis). Estes tipos de tráfego caracterizam-se individualmente e após, ajustam-se modelos destes, multiplexados, utilizando estatísticas de ocupação de "buffer", tempo de espera de fila e probabilidade de bloqueio, definidos sobre parâmetros QoS.

Para a modelagem estocástica as referências Onvural [4] e Leduc [14] sintetizam as mais importantes famílias de processos estocásticos aplicáveis ao tráfego ATM (vide Figura 2.5), que podem ser classificados em dois grandes grupos: **processos discretos no tempo** e **processos contínuos no tempo**. Para não se dispender detalhes exces-

sivos que fogem ao escopo deste trabalho apenas fornecemos a classificação destes na Figura 2.5, com suas interligações conceituais, recomendando também os "papers" [31] e [32], que fornecem raras sínteses de todos estes processos, incluindo as derivações para modelos auto-similares.

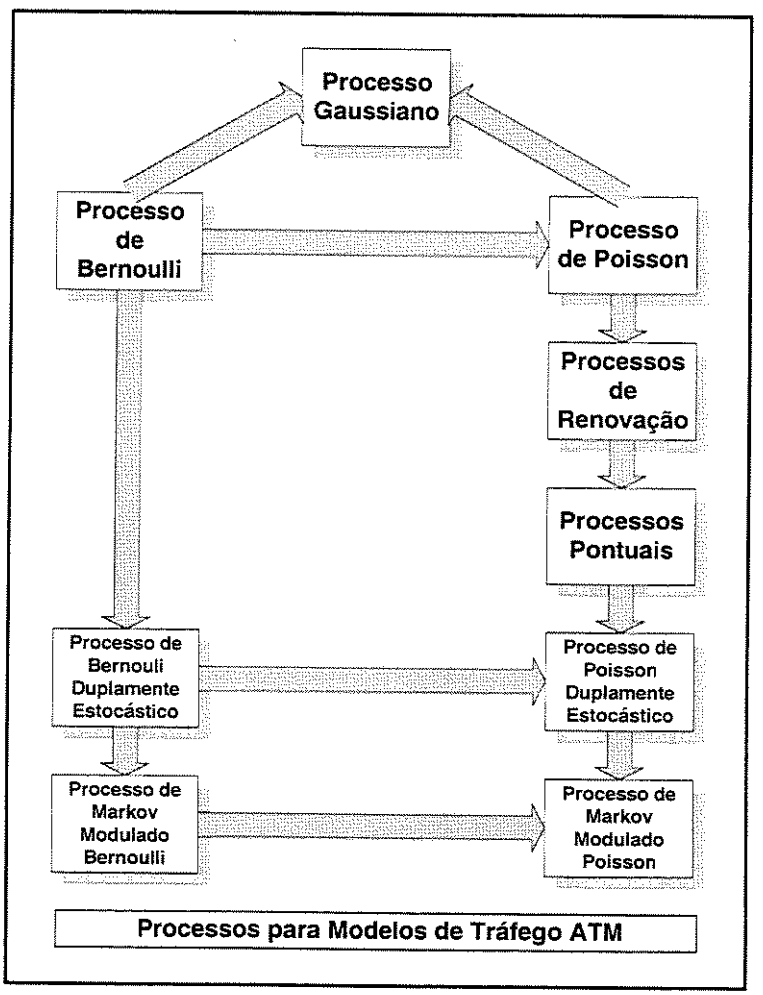

Figura 2.5: Processos Estocásticos para Modelos de Tráfego ATM

Com a evolução dos modelos de análise, surgiu a separação destes por escalas de tempo (inicialmente por Hui em [33]). Verificou-se que a natureza do tráfego de dados se comporta de diferentes maneiras conforme a escala de tempo, exceto para o tráfego auto-similar. Esta separação ocorre em três momentos: escala de chamada ou conexão, escala de surto e escala de células.

Esta hierarquia é ilustrada na Figura.2.6.

Na **escala de tempo de célula**, as próprias células constituem elementos discretos. Tratando-se da **escala de tempo de jatos**, a granularidade da célula é tornada desprezível, aproximando-a ao modelo de fluxo fluído. A **escala de chamada ou conexão** é caracterizada pelo tempo total de duração da chamada, sendo a maior das

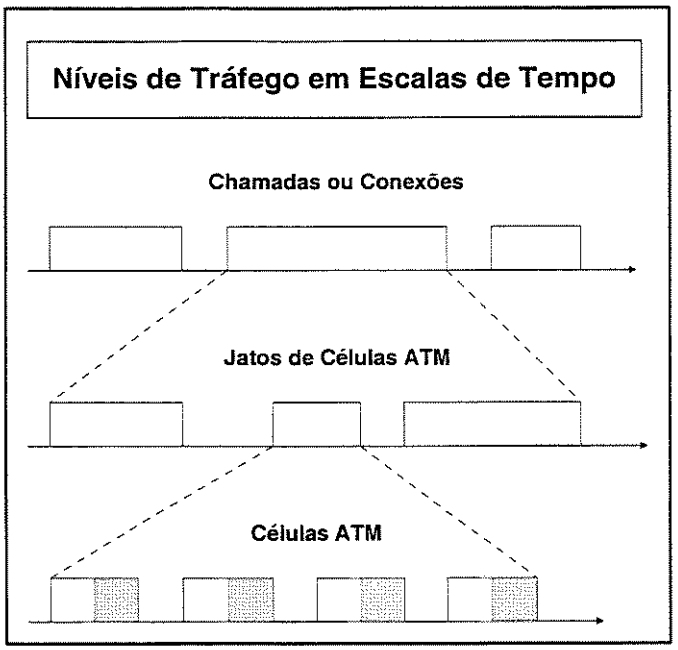

Figura 2.6: Hierarquia de Escalas de Tempo

escalas de tempo para modelos de tráfego. Em cada uma das escalas, o enfoque e os problemas são diferentes [22].

Enquanto que na escala de célula, a questão é tratar para que as flutuações aleatórias de taxas de trem de células não ultrapassem a capacidade do enlace, na escala de surto o problema maior é quantificar o CLP para calcular as perdas. Na escala de chamada a preocupação é com a $(BW)_{efetiva}$, fator que afeta a totalidade de tempo de conexão e os requisitos QoS desta.

É neste ponto que o fenômeno da auto-similaridade se faz presente, pois a nível de conexão ele é capaz de alterar as fronteiras entre esta e a escala de surto [5], podendo afetar conexões com ingerências a nível de surto.

A seguir, procede-se a uma abordagem de alguns modelos de tráfego que atendem as redes ATM.

2.2.2. Modelo de fonte de fluxo de fluído de pacotes de voz.

Uma fonte de voz pode ser representada por um processo de dois estados, porque a voz humana consiste de sequências alternadas de atividade (média entre 0,4 a 1,2 segundos) e inatividade (média entre 0,6 a 1,8 segundos).

Este fenômeno foi grandemente utilizado para otimizar a utilização de centrais analógicas de comutação (TASI) e recentemente aplicado a telefonia digital [35] para multiplexar sinais de voz digitalizados (DSI).

Em fontes de tráfego, a Figura 2.7 é a que melhor descreve este fenômeno. As-

sumindo, para ambos os estados (silêncio e ativo), uma distribuição de comprimento exponencial, a fonte de voz pode ser representada pelo modelo de nascimento-morte de dois estados dado na Figura 2.7.

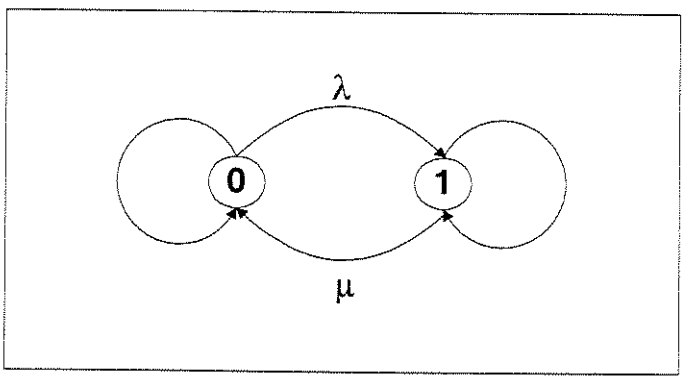

Figura 2.7: Modelo de voz de estados "OFF" (0) e "ON" (1)

O parâmetro λ representa a taxa de transição de saída do estado de silêncio ("0" para "1") e μ do estado ativo para silêncio ("1" para "0").

O tempo médio de atividade será: $\frac{1}{\mu}$ seg.

O tempo médio de silêncio (intervalo) é: $\frac{1}{\lambda}$ seg.

A proporção de atividade (fala) será: $\frac{\lambda}{\lambda+\mu}$

Obs.: O sistema TASI utiliza a proporção de $\frac{1}{\mu} = 0,4$ seg. para o estado de fala e $\frac{1}{\lambda} = 0,6$ seg. para silêncio o que fornece a proporção de atividade de $0,4$, ou seja, em média numa conversação, fala-se 40% do total do tempo transcorrido.

Este fato faz com que possamos acomodar pelo menos $\frac{1}{0,4} = 2,5$ vezes circuitos TASI para a voz. Em mais detalhes, a voz digitalizada (período ativo) executa amostragem a cada 125 μseg. com codificação de 8 bits (256 níveis). Cada amostragem representa um octeto de 8 bits e, com 47 octetos (serviço AAL1 da camada ATM), formamos a carga útil de voz para célula ATM de 53 octetos. Com isto, no período ativo, a fonte gera 170 células ATM/segundos nesta camada ATM.

Para N fontes independentes de voz multiplexadas, conforme Figura 2.8, em um acesso de "buffer" da rede. Cada fonte gera V células/segundos, no estado ativo e devido a vantagem estatística já mencionada, a saída do enlace pode ser menor que $N \times V$, sendo este o número máximo de células que podem ser geradas. A capacidade do link de saída será VC células e devido as propriedades estatísticas já mencionadas, $VC < VN$ ou $C < N$, sendo C um parâmetro adimensional.

O número médio de geração de células será: $VN\frac{\lambda}{\lambda+\mu}$, e o parâmetro C pode ser achado por:

$$VN\frac{\lambda}{\lambda+\mu} < VC => \ C > N\frac{\lambda}{\lambda+\mu}$$

Isto representa a condição de estabilidade e plena utilização de C, para um "buffer" infinito, na prática apenas ρC (onde ρ é o fator de utilização) será utilizada e

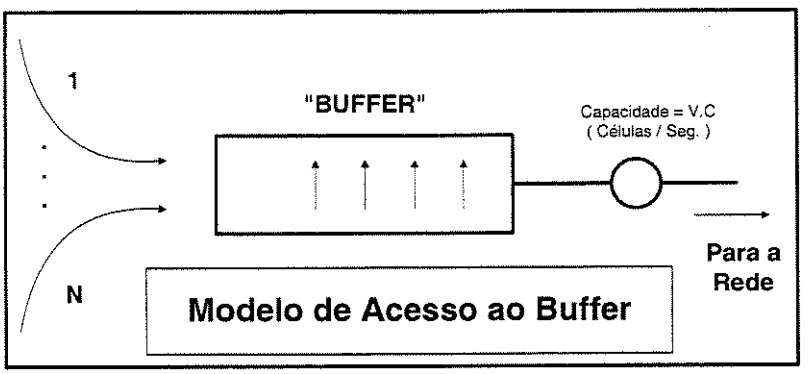

Figura 2.8: Modelo de acesso ao Buffer para voz

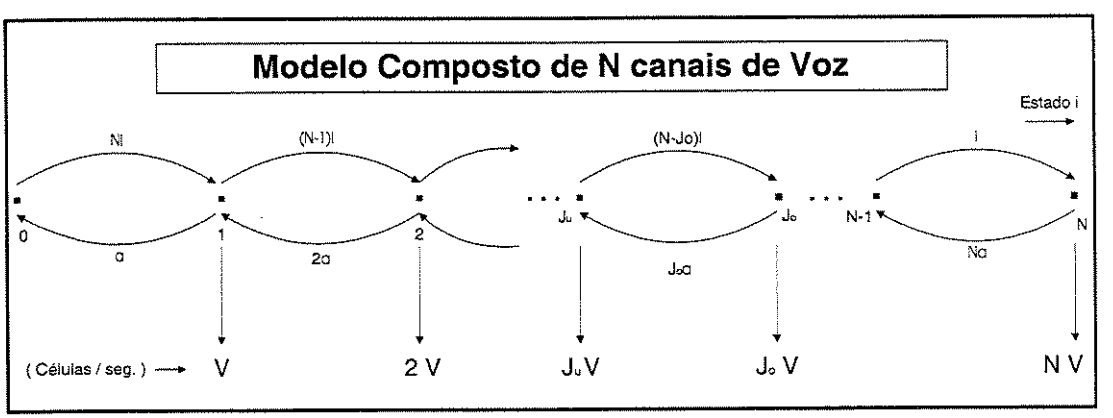

Figura 2.9: Modelo composto de N canais de voz

dividindo por C obtemos $1 > \left(\frac{\lambda}{\lambda+\mu}\right)\left(\frac{N}{C}\right) = \rho$ ou:

$$\rho = \left(\frac{\lambda}{\lambda+\mu}\right)\left(\frac{N}{C}\right) \tag{2.1}$$

Dado o modelo de dois estados pode-se expandi-lo para N fontes conforme a Figura 2.9.

Este modelo é chamado de $(N+1)$ estados de nascimento e morte.

No estado i, i fontes estão ativas e a taxa média de liberação de células será $i \times V$ células / seg. Os estados J_u e J_o são definidos como:

$$J_u = \lfloor C \rfloor \qquad J_o = \lceil C \rceil$$

J_u é o estado abaixo da carga ou abaixo de C (parte inteira de C, abaixo) e,
J_o é o estado além-carga ou parte inteira C, acima.

Estes parâmetros são definidos porque, se o número de fontes chega até J_u, a fila no "buffer" 1 tende a esvaziar-se, mas se ocorre J_o a fila tende a encher-se. A taxa de mudança para qualquer estado i é $V(C-i)$ células/seg. Então uma fonte composta

pode estar no estado i com probabilidade;

$$\pi_i = \binom{N}{i} \left(\frac{\lambda}{\lambda+\mu}\right)^i \left(\frac{\mu}{\lambda+\mu}\right)^{N-i} \tag{2.2}$$

ou seja, esta é a probabilidade que i de N fontes de dois estados, estejam em atividade (cada uma com probabilidade $\frac{\lambda}{\lambda+\mu}$) enquanto que as fontes que sobram, $(N-i)$, estejam inativas (probabilidade de $\frac{\mu}{\lambda+\mu}$). Reescrevendo a função π_i temos:

$$\pi_i = \binom{N}{i} \left(\frac{\lambda}{\mu}\right)^i \left(1+\frac{\lambda}{\mu}\right)^{-N} \tag{2.3}$$

Considerando agora o processo genérico de nascimento e morte na Figura a 2.10, em que o parâmetro λ_i representa a taxa de transição do estado i para $i+1$, e μ_i a taxa de i para $i-1$ então, da Teoria de Filas.

$$\pi_i = \frac{\lambda_0 \lambda_1 ... \lambda_{i-1}}{\mu_1 \mu_2 ... \mu_i} \pi_0 \tag{2.4}$$

obs.: Em relação ao caso anterior $\mu_1 = \mu$, $\mu_2 = 2\mu$.....com π_0 sendo achado considerando $\sum_{i=1}^{N} \pi_i = 1$.

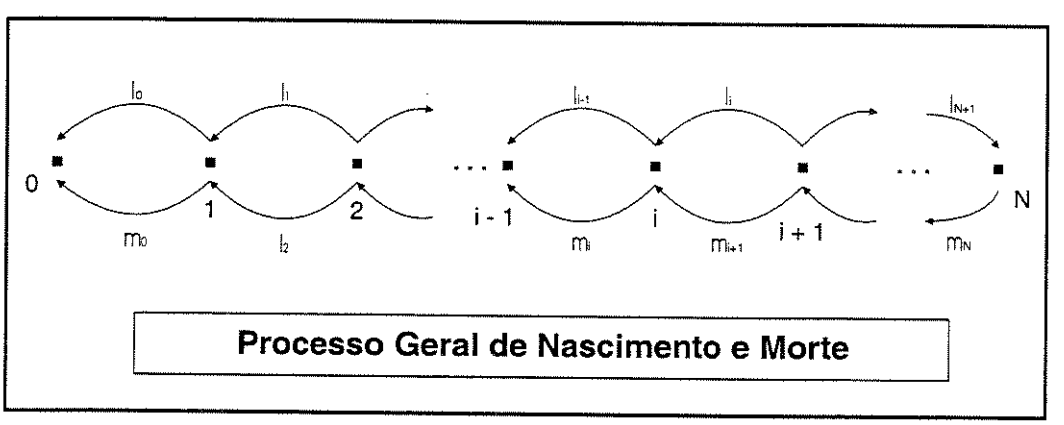

Figura 2.10: Processo geral de nascimento e morte

Também pode-se concluir que existe um conjunto de probabilidade formando um vetor coluna $\overrightarrow{\pi}$ tal que:

$$\overrightarrow{\pi} = [\pi_0, \pi_1, \pi_2 ..., \pi_N]'$$

e tal que $\overrightarrow{\pi}.\overrightarrow{M} = 0$ onde $\overrightarrow{M} = \begin{bmatrix} -N\lambda & N\lambda & 0 & ... \\ \mu & -[\mu + (N-1)\lambda] & (N-1) & ... \\ 0 & 2\mu & -(N-2)\lambda - 2\mu & ... \\ 0 & 0 & 3\mu & ... \\ ... & ... & ... & ... \end{bmatrix}$

A equação matricial acima é um caso especial de **cadeia de Markov contínua no tempo** e a matriz M é conhecido como matriz de geração infinitesimal. Então, dado o modelo para N fontes estatisticamente multiplexados (fontes de voz em pacotes representados pelas equações 2.4 à 2.7, vamos associá-los para calcular a performance e parâmetros de projetos tais como atraso, perdas estatísticas e tamanho de "buffer" requerido. Segundo M. Schwatz [35], três abordagens têm sido necessárias para caracterizar os tráfegos de voz e vídeo:

- O primeiro tenta captar o processo de chegada a taxa V de cada fonte utilizando um processo semi-Markov;

- O segundo modelo aproxima a geração de células (pacotes) por uma fonte no estado ativo como um processo de Poisson produzindo pacotes de comprimento de distribuição exponencial;

- O terceiro modelo assume que cada fonte ativa transmite informação uniformemente, com o enlace de transmissão e o servidor operando da mesma forma. Tal modelo é chamado **modelo de fluxo fluido.**

Este último modelo será enfocado, devido a sua importância e possibilidade de generalização de seus resultados.

O **modelo de fonte de fluxo fluído de pacotes de voz** se forma assumindo que o número de células geradas durante o período ativo é tão grande que pode ser considerado um fluxo contínuo de fluido. Esta aproximação é válida na prática, pois as redes ATM assumem dimensões grandes e distribuídas que, além do número de células ser considerado grande o número de fontes N e a capacidade VC também o são.

Nestas condições de discretização no "buffer", do fato de células chegando e saindo podendo ser desprezado, faz-se então da ocupação do "buffer" uma variável aleatória (V.A) contínua X. Este é o fluido referido na expressão análise de fluxo fluido. As unidades de X são definidas como sendo o número de células chegando durante o período de fila.

Então, conforme Figura 2.11.

Aqui, a fonte de voz, gerando células a taxa de V célula/segundos durante o período ativo de voz de comprimento médio $\left(\frac{1}{\alpha}\right)$ segundo, causará incremento de x por (V/α) células (pela média) durante este período ativo. Isto se denomina "unidade de informação". Então, o sistema original com capacidade VC células/seg. terá uma capacidade equivalente de$(VC)(V/\alpha) = \alpha C$ "unidades de informação"/seg. tornando o servidor de capacidade original VC, agora de capacidade equivalente αC "unidades de informação"/seg.

A ocupação do "buffer" é calculada através das estatísticas da V.A. X. Seja l o estado do "buffer" em unidades de células e que $X = x$ aumenta por (V/α) células durante a média do período ativo de fala, então $l = x\,(V/\alpha)$ "unidade de informação". A probabilidade $P\{l > i\}$ que o "buffer" exceda sua "ocupação" em algum limite é dado

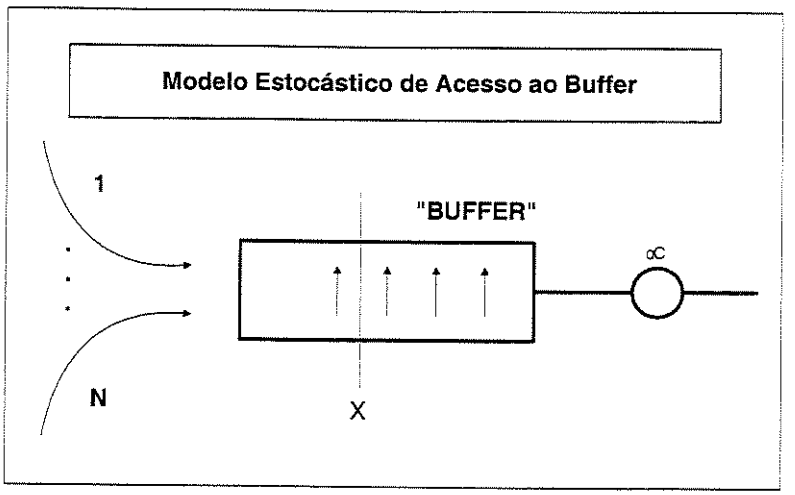

Figura 2.11: Modelo estocástico de acesso ao buffer

por:

$$P\{l > i\} = P\left\{x > \frac{\alpha i}{V}\right\} \tag{2.5}$$

que fornece a probabilidade que l exceda em um limite de i células é equivalente a afirmar que a probabilidade de que x exceda a $\left(\frac{\alpha i}{V}\right)$ "unidade de informação". De acordo com o trabalho pioneiro de D. Mitra em [36], na Figura 2.11, se existem i fontes em estado ativo de fala, elas bombeiam (αi) "unidade de informação" para o "buffer", sendo que este se esvazia a taxa de (αC) "unidade de informação"/seg.. No estado não equivalente, Figura 2.11, o "buffer" está enchendo se $i > C$ e esvaziando se $i < C$. A diferença é que as variáveis passaram de determinísticas a aleatórias contínuas.

D. Mitra define $F_i(t, x)$ como sendo a distribuição cumulativa de probabilidade no tempo t, com o sistema no estado i, e assumindo uma fila infinita e a função geradora $F_i(t + \Delta t, x)$ para o modelo considerado de dois estados, com taxas de esvaziamento e enchimento de "buffer" e através da Teoria de Filas chega à:

$$P\{l > i\} = \rho e^{\frac{(1-\rho)(1+\gamma)}{(1-C)}\left(\frac{\alpha i}{V}\right)} \tag{2.6}$$

onde $\gamma = \frac{\lambda}{\alpha}$, ou seja, a razão entre as taxas de fala (λ) e de silêncio (α), e (ρ) é a taxa de "ocupação" do "buffer" ou fator de utilização.

Observe que o fator de utilização do "buffer" (ρ) depende inversamente de C, a capacidade do servidor em células, já que as taxas de transição de estados $\lambda\,(off - on)$ e $\alpha\,(on - off)$ são consideradas constantes, sobre a média, para a voz.

Para N fontes de voz, chega-se ao fator de utilização:

$$\rho = \left(\frac{\gamma}{1-\gamma}\right)\left(\frac{N}{C}\right) \tag{2.7}$$

e $P\{l > i\}$, para N fontes de voz, como:

$$\begin{aligned} P\{l &> i\} \approx -A_N \rho^N e^{-r\left(\frac{\alpha i}{V}\right)} \text{ (a)} \\ r &= \frac{(1-\rho)(1+\gamma)}{1-\frac{C}{N}} \text{ (b)} \\ A_N &= \left(\frac{N-C}{N-2C+\frac{C}{\rho}}\right)^N \left[\frac{\rho}{C(1-\rho)} - \frac{1}{N-C}\right] \sum_{i=0}^{\lfloor C \rfloor} \frac{(C-i) f_i}{\lambda^i \left(\frac{N}{C}-1\right)} \text{ (c)} \end{aligned} \tag{2.8}$$

onde f_i é dado em [36], como função de geração de probabilidades para cada caso i.

O importante é que o comportamento dominante de $P\{l > i\}$ é determinado pela exponecial $e^{-r\left(\frac{\alpha i}{V}\right)}$ e que apenas é multiplicado em escala por A_N e ρ^N, note também que basta aumentar C na mesma proporção que N que este comportamento assintótico dado por $e^{-r\left(\frac{\alpha i}{V}\right)}$ não se altera.

Outro modelo bastante aplicado à multiplexação de voz é o de **Markov modulado por Poisson**, que pode ser bem entendido por [4] [5] e [35] e veja adiante este modelo para tráfego heterogêneo.

2.2.3. Modelo de vídeo em fonte fluida de pacotes ATM

Para pacotes de vídeo, adotaremos procedimento semelhante ao de pacotes de voz e utilizando o método de **processo equivalente de fontes superpostas** [37], como a seguir. Quando N fontes de vídeo são multiplexadas juntas, cada fonte é caracterizada por suas estatísticas de primeira e segunda ordens (taxa de bit média $\overline{\lambda}$ e função de covariância para cada fonte $C_i(n)$). Para se determinar a "ocupação" do "buffer", suas estatísticas e o seu CLP, a Figura 2.12 mostra os modelos caracterizando o multiplex, sendo a capacidade C do MUX em bits/pixel para o contexto de vídeo.

Na parte a) da figura está representada a multiplexação de fontes de Vídeo originalmente como ela ocorre, na parte b) o modelo de processo equivalente com M mini-fontes equivalentes cujo modelo de dois estados em c). Cada mini-fonte move seu estado para frente e para trás exponencialmente entre um estado "ON" e um estado "OFF", em que A bits/pixel são liberados para o "buffer". Tanto na parte a) quanto b) a taxa de entrada no "buffer" é $\lambda(t)$ denotando a equivalência dos modelos.

As M minifontes equivalentes podem ser representadas por um processo composto de cadeia de Markov de $(M+1)$ estados sendo $M >> N$, conforme Figura 2.12, com taxas de transição dependentes do estado sendo a taxa de bit variante no tempo $\lambda(t)$ "quantizada" para os valores $0, A, 2A, ..., MA$ bits/pixel.

Agora o próximo passo será determinar os parâmetros α, β, e A do modelo equivalente, para isto adota-se a abordagem de combinar as estatísticas de primeira e segunda ordem com estatísticas medidas. Sabendo-se que a taxa entrante no "buffer"

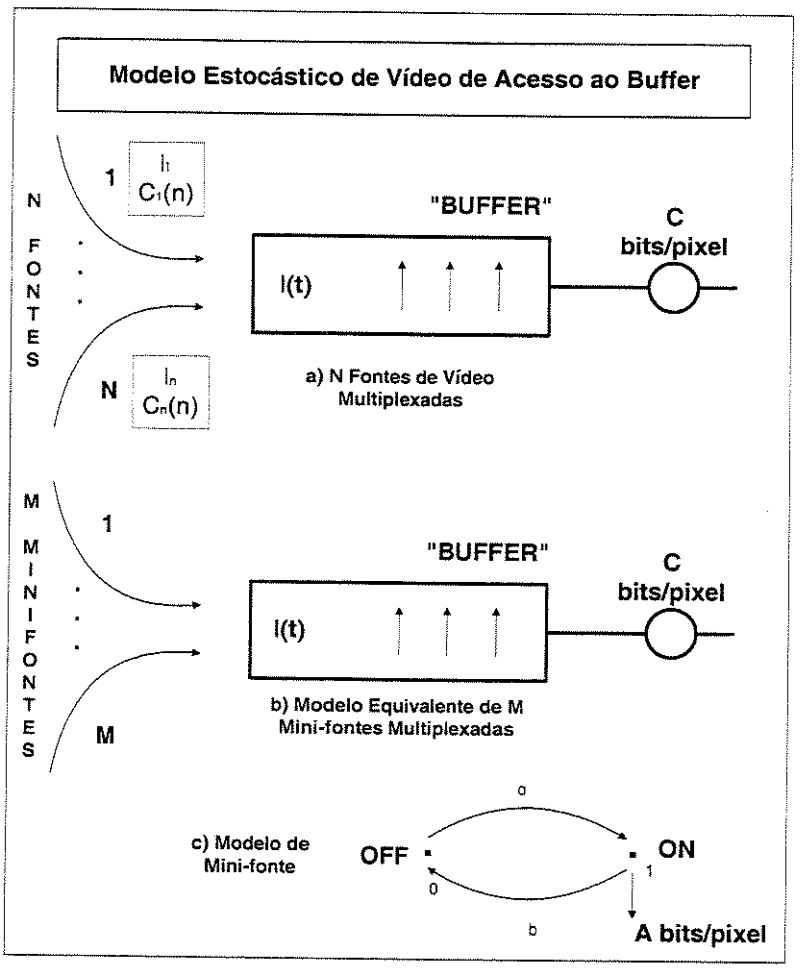

Figura 2.12: Modelo estocástico de vídeo de acesso ao buffer

$\lambda(t)$ é a multiplexação das fontes i então;

$$\lambda(t) = \sum_{i=1}^{N} \lambda_i(t) \tag{2.9}$$

e:

$$C(\tau) = \sum_{i=1}^{N} C_i(\tau) \tag{2.10}$$

aqui $C_i(\tau)$ é a função de autocovariância da fonte i, dada por:

$$C_i(\tau) = E[\lambda_i(t)\,\lambda_i(t+\tau)] - E^2(\lambda_i) \tag{2.11}$$

e sabemos que;

$$E(\lambda) = \sum_{i=1}^{N} E_i(\lambda_i) \tag{2.12}$$

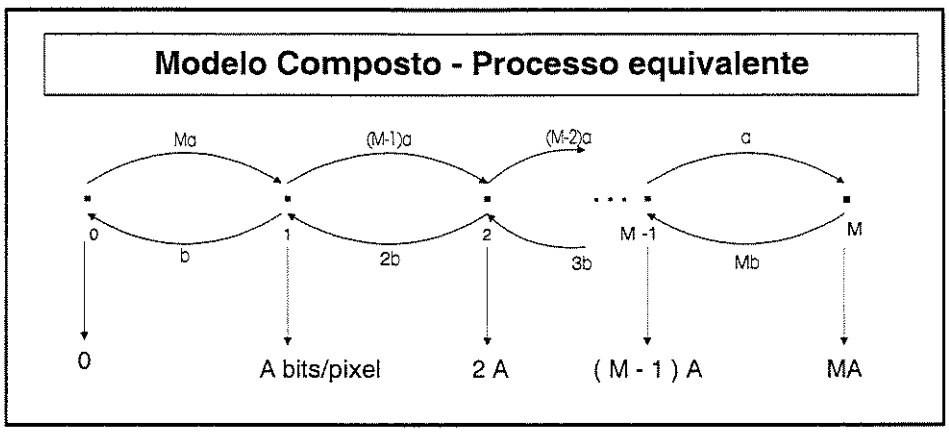

Figura 2.13: Modelo composto de vídeo por processo equivalente

e que;

$$\sigma^2 = \sum_{i=1}^{N} \sigma_i^2 \tag{2.13}$$

o parâmetro σ_i^2 representa a variância da fonte de vídeo i, enquanto σ^2 é a variância do sinal composto, observando que $C(0) = \sigma^2$ e $C_i(0) = \sigma_i^2$, e que para o caso especial de N fontes independentes: $C(\tau) = NC_i(\tau)$, $E(\lambda) = NE(\lambda_i)$ e $\sigma^2 = N\sigma_i^2$

Pode ser mostrado que, do modelo de c) a autocovariância de cada fonte i é (M. Schwartz em [35]);

$$C_i(\tau) = A^2 \frac{\alpha\beta}{(\alpha+\beta)} e^{-(\alpha+\beta)\tau} \tag{2.14}$$

que representa a cadeia de Markov em tempo contínuo de dois estados trocando o valor "0" por "A" nas taxas α e β, sendo denominado por Papoulis em [38], de *sinal telegráfico aleatório*. A probabilidade da minifonte estar em ON é $p = \frac{\alpha}{(\alpha+\beta)}$ e de estar em OFF será $(1-p)$, assim;

$$C_i(\tau) = A^2 p(1-p) e^{-(\alpha+\beta)\tau} \tag{2.15}$$

Aquí o comportamento da covariância das mini-fontes é exponencial com constante de tempo igual à $\frac{1}{(\alpha+\beta)}$ e, agora, a covariância e a esperança (taxa média de bit) do sinal composto, considerando todas as fontes independentes é:

$$C(\tau) = MA^2 p(1-p) e^{-(\alpha+\beta)\tau} \tag{2.16}$$

$$E(\lambda) = MpA$$

Com estes parâmetros determinados vamos utilizar a técnica de fluxo fluído empregada nos modelos para voz para determinar a ocupação do "buffer" que, como na

Figura 2.12 é a V.A. X e da mesma forma define-se a função de distribuição de probabilidade de mini-fonte i como $F_i(X = x)$ como a probabilidade conjunta de que a ocupação do "buffer" é menor ou igual que x com i mini-fontes em "ON". Nesta linha e usando novamente a teoria de filas, D. Mitra [36], demonstra que:

$$\begin{aligned} P\{l \geq x\} &\approx -A_M \rho^M e^{-\beta r x/KA} \text{ (a)} \\ r &= (1-\rho)(1+\alpha/\beta)/[1-(C/MA)] \text{ (b)} \\ \rho &= N\bar{\lambda}/C = MpA/C \text{ (c)} \end{aligned} \tag{2.17}$$

onde $A_M = A_N$ já citado em modelo de fonte fluida de voz.

Novamente, constatamos o comportamento assintótico a) dominante de r b) agora em função das M mini-fontes equivalentes, que pode ser obtida da relação c).

2.2.4. Modelo heterogêneo em pacotes ATM

Para propósitos de modelagem, uma rede ATM pode ser vista como uma coletânea de filas conectadas em uma determinada maneira por uma topologia de rede [21]. Um servidor corresponde a um enlace de transmissão e existe um "buffer" finito associado a cada servidor que temporariamente armazenam as células que chegam quando a taxa de chegada é maior daquela em que o servidor pode transmitir. Este pode ser o caso, por exemplo, quando mais de uma conexão está ativa simultaneamente sobre um enlace com múltiplas conexões.

Teoricamente, é possível desenvolver um modelo Markoviano de uma rede de comunicações e resolvê-lo numericamente incluindo a nova conexão para determinar se a rede pode acomodá-la ou não. Na prática cada modelo de filas consiste de dezenas e centenas de filas com tráfego gerado de centenas de fontes, sendo impraticável resolver cada modelo numericamente, em tempo real, a não ser para pequenas redes, que para o momento atual, isto é utopia. A solução é decompô-las em pequenas filas individuais e analisá-las e isto é que tem sido feito, caracterizando a chegada do *trem* de células, o processo efetivo de serviço e capacidade do "buffer". Tudo isso assumindo que o comportamento da fila isolada será o mesmo que inserida na rede ATM

Inicialmente, considere a rede de filas a um tempo em que a célula parte da fila i e tenta entrar na fila j. Se há um espaço disponível na fila j, então esta célula junta-se a fila, se não, é descartada e excluída do sistema. Com isto, o estado ou o processo de serviço no nó i não é afetado pelo estado dos nós anteriores, assumindo que os nós intermediários não retransmitem células ou trocam mensagem com seus vizinhos, como ocorre em redes ATM [4].

Assim, o processo de chegada de uma fila superposta do outro lado da rede ATM é a superposição dos processos de partida do *trem* de células em filas das fontes ATM. Isto torna as coisas mais simples, mas existem dificuldades. Se assumirmos que o comportamento estocástico do processo de chegada das células de uma conexão em um

nó intermediário seja o mesmo que proveio da fonte ATM, então o enlace de transmissão i na rede de capacidade C pode ser modelado como uma simples fila com N *trem* de células (N conexões sendo multiplexadas no nó i) chegando, com um "buffer" finito de tamanho M, e tempo de serviço constante, que para redes ATM é:

$$t_s = \frac{(53 \text{ bytes}) \times (8 \text{ bits/byte})}{C} = \frac{424 \text{ bits}}{C \text{ (bps)}}$$

Se cada fonte for modelada por uma fonte Markoviana de dois estados, então o CLP nesta fila exigirá a solução de $2^N (K+1)$ equações lineares [4], por isso a análise se tornará impraticável quando N crescer muito. Um tratamento típico para amenizar esta dificuldade é reduzir o número de *trens* de células pela superposição de todos os processos componentes de chegada a um número menor de *trens* de células, reduzindo a dimensionalidade do problema. Uma fila pode ser analisada como dois fluidos de células chegando: uma correspondendo à nova conexão a ser submetida ao CAC e outra representando a superposição fluída de células em tráfego. Outra opção é analisar um fluxo com a nova conexão já inserida, constituindo um fluido simples de células.

O processo estocástico apropriado para esta análise é o processo de Markov modulado por Poisson-**MMBP**, pois o tempo neste é discretizado em intervalos fixos que, por conveniência, será o tempo t_s de serviço ATM dado acima e será denominado "slot" de tempo. A probabilidade que um "slot" contenha uma célula é um processo Bernoulli com um parâmetro variando com o estado r do processo de Markov, que é independente do processo de chegada.

Ao fim de cada "slot", o processo de Markov muda do estado i para o estado j com probabilidade P_{ij} ou fica no estado i com probabilidade P_{ii} tal que $\sum_{j=1}^{r} P_{ij} = 1$ para todo $i = 1, ..., r$. Quando no estado i, um "slot" contém uma célula, evento com probabilidade α_i ou nenhuma célula com probabilidade $(1 - \alpha)$.

A probabilidade de chegada de célula e o processo de Markov envolvido são assumidos como independentes.

O MMBP é caracterizado pela matriz probabilidade de transição P e a matriz diagonal Λ de probabilidade de chegadas:

$$P = \begin{bmatrix} P_{11} & \dots & P_{1r} \\ \dots & \dots & \dots \\ P_{r1} & \dots & P_{rr} \end{bmatrix} \tag{2.18}$$

$$\Lambda = \begin{bmatrix} \alpha_1 & 0 & \dots & 0 \\ 0 & \alpha_2 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \alpha_r \end{bmatrix} \tag{2.19}$$

Se denominarmos o tempo t_n o intervalo entre o estado $(n-1)$ e a chegada n-ésima, podemos escrever para o "slot" k, que o coeficiente de autocorrelação do tempo

entre chegadas é:

$$\Psi_k = \frac{E\{t_n, t_{n+k}\} - E^2\{t_n\}}{Var\{t_n\}} \tag{2.20}$$

Se chamarmos T_i o intervalo de tempo para a próxima chegada, considerando que o processo de Markov está no estado i, e sendo T o intervalo de tempo de uma célula e considerando o tempo em que ocorre uma chegada de célula quando o processo de Markov está no estado 1:

No próximo "slot", o MMBP pode:

- permanecer no estado 1 e uma chegada pode ocorrer, com probabilidade $p\alpha$;

- mudar para o estado 2 e uma chegada de célula pode ocorrer com probabilidade $(1-p)\beta$;

- permanecer no estado 1 e nenhuma chegada ocorrer, com probabilidade $p(1-\alpha)$;

- mudar para o estado 2 e nenhuma chegada ocorrer, com probabilidade $(1-p)(1-\beta)$

E, para o MMBP de dois estados, as matrizes probabilidade de transição P e de probabilidade de chegada Λ podem ser reescritos:

$$P = \begin{bmatrix} p & 1-p \\ 1-q & q \end{bmatrix} \tag{2.21}$$

$$\Lambda = \begin{bmatrix} \alpha & 0 \\ 0 & \beta \end{bmatrix} \tag{2.22}$$

com isto,Onvural [4], no seu apêndice A.1, determina a função geradora $T(z)$ e, após manipulações chega a que a função de autocorrelação do tempo interchegadas, para o "slot" $k = 1$, será:

$$\Psi_1 = \frac{Cov\{T_{n-1}T_n\}}{Var\{T_n\}} = \frac{E\{T_{n-1}T_n\} - E\{T_{n-1}\}E\{T_n\}}{Var\{T_n\}}$$

$$\Psi_1 = \frac{\alpha\beta(\alpha-\beta)^2(1-p)(1-q)(p+q-1)^2}{c^2(2-p-q)[\alpha(1-q)+\beta(1-p)+\alpha\beta(p+q-1)]^2} \tag{2.23}$$

onde:

$$c^2 = \frac{2[(1-q)\alpha+(1-p)\beta]}{(1-q)\alpha+(1-p)\beta+\alpha\beta(p=q-1)} - \frac{(1-q)\alpha+(1-p)\beta}{2-p-q} + \ldots$$

$$\ldots + \frac{2[(1-p)\alpha+(1-q)\beta][(1-q)\alpha+(1-p)\beta](p+q-1)}{(2-p-q)^2[(1-q)\alpha+(1-q)\beta+\alpha\beta(p+q-1)]} - 1$$

e que, pela expressão de Ψ_1, conclui-se que, quando $\alpha = \beta$ o MMBP tem somente um estado e torna-se um processo Bernoulli puro. Se α ou β forem iguais a zero então o MMBP degenera para um IBP (Interrupted Bernoulli Process - Processo de Bernoulli Interrompido).

A **Superposição de MMBP's** é denominada de processo de Bernoulli comutado em lote. Neste Processo, o tempo em um SBBP é dividido em "slots" de igual comprimento, como convém às redes ATM. As chegadas durante um "slot" ocorre como um Processo de lote de células, em vez de apenas uma célula com o tamanho do lote obedecendo a uma distribuição de acordo com o K-estado da cadeia de Markov.

Para simplificar, esta SBBP será abordado para N MMBP's, cada um com dois estados e parâmetros P_i e Λ_i, onde refere-se ao estado j-ésimo processo componente MMBP. Então, o processo superposto tem 2^n estados em que cada estado $(i_1, i_2, ...i_N)$ denotamos os estados dos N processos MMBP. Em particular, ao final de um "slot", cada processo componente no estado 1 muda para o estado 2 com probabilidade $(1 - p_i)$ ou fica no estado 1 com probabilidade p_i. As respectivas probabilidades para processos componentes que estão no estado 2 são iguais à $(1 - q_i)$ e q.

Definimos então $p\left(i_j \longrightarrow i_j^*\right)$ como a probabilidade que o j-ésimo processo componente está no estado i_j^*, dado que ele estava no estado i_j, durante o "slot" anterior, e $p\left(i \longrightarrow i'\right)$ como a probabilidade que o SBBP está no estado i', dado que ele estava no estado i durante o "slot" anterior.

Então:

$$p\left(i \longrightarrow i'\right) = \prod_{j=1}^{N} p\left(i_j \longrightarrow i_j^*\right) \tag{2.24}$$

as probabilidades $p\left(i \longrightarrow i'\right)$ definem os elementos da matriz probabilidade de transição do processo superposto. Para definir completamente o SBBP, chamamos B_i a V.A. denotando a distribuição do tamanho do lote quando o processo superposto está no estado $i = (i_1, i_2, ...i_N)$. Em seqüência $\gamma\left(i_j\right)$ é a probabilidade que ocorre a chegada de uma célula no processo componente j, $\gamma\left(i_j\right) = \alpha_j$ se o processo está no estado 1 ou $\gamma\left(i_j\right) = \beta_j$ se ele está no estado 2.

Aos dois extremos, nenhum dos processos componentes poderá gerar uma célula com probabilidade: $\prod_{i=1}^{N} \left\{1 - \gamma\left(i_j\right)\right\}$, ou cada processo componente pode gerar uma célula, isto é, com probabilidade $\prod_{i=1}^{N} \left\{\gamma\left(i_j\right)\right\}$. Quando são geradas m células, m saem de N fontes geradoras de células, enquanto nenhuma chegada ocorre das remanescentes fontes existem $\frac{N!}{m!(N-M)!}$ combinações de m células saindo de N fontes.

Faz-se S_j o conjunto de m-etuplas de índices e $S = \{1, ..., N\}.$, e $q \in S_l$ seja o índice do processo componente l que gera células, enquanto que aquele que não gera está em $(N - q)$. A distribuição de probabilidade de B_i pode agora ser escrita como segue;

$$P\left\{B_i = l\right\} = \sum_{r \in S_l} \prod_{i_j \in q} \gamma\left(i_j\right) \prod_{i_j \in (S-q)} \left\{1 - \gamma\left(i_j\right)\right\} \tag{2.25}$$

e então o processo superposto é um SBBP de 2^n estados com matriz probabilidade $P = \{p(i \longrightarrow i')\}$ de transição e distribuição de tamanho $P\{B_i = l\}$de lote para cada estado i e $l = 0, ..., N$.

Concluindo, esta teoria pode ser utilizada a partir desta última equação para proceder estudos semelhantes de ocupação do "buffer" realizados em fontes fluidas de voz e vídeo desde que sejam identificados os processos em superposição.

2.2.5. Modelos de tráfego auto-similar

Em escala de conexão, verificou-se nos últimos anos [28] [29] [39], que o tráfego de dados das redes de dados Ethernet apresentava características de "auto-similaridade", ou seja, um fenômeno que já havia sido detectado em várias ocorrências da natureza e que, basicamente, consiste na manifestação das mesmas características estatísticas em escalas de tempo diferentes.

Na Figura 2.13, há um exemplo em [40], de tráfego ATM como processo estocástico auto-similar. Na parte a) o sinal é semelhante em várias escalas de tempo, o que não ocorre na parte b). Para se medir o grau de auto-similaridade os autores mencionados acima se utilizaram do *Parâmetro de Hurst* descoberto por H. E. Hurst [41], ao estudar as séries temporais que descreviam o processo de armazenamento e fluxo de água em vários rios e reservatórios do mundo.

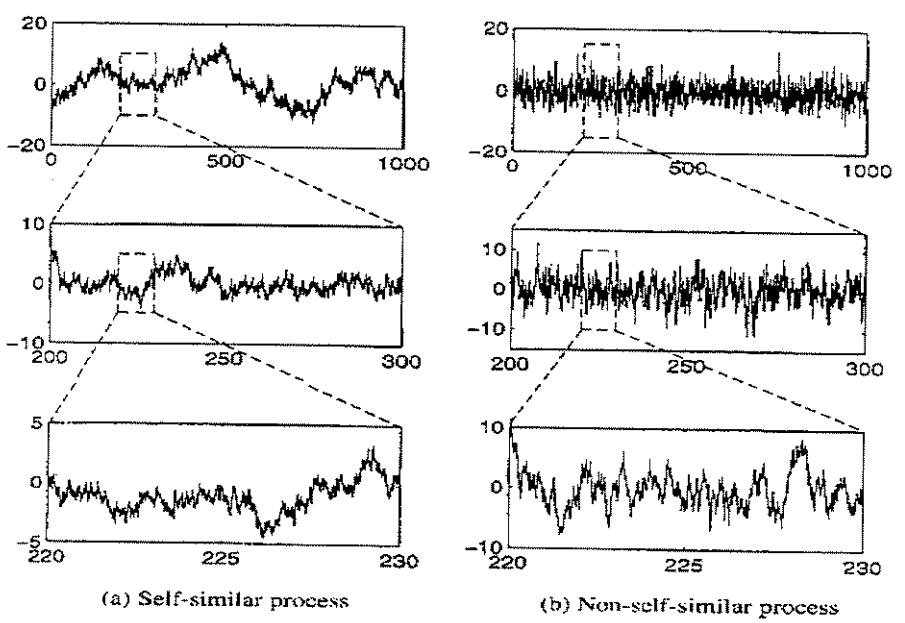

Figura 2.14: Tráfego Auto-similar(a) e não Auto-similar(b)

Hurst verificou empiricamente que a variação $R(N)$ dos níveis de água observados em um grande número N de anos se relacionava com a variância S dos dados coletados

da seguinte forma;

$$R/S \sim (N/2)^H \quad \text{com } H > 0,5$$

e que, para qualquer outro processo de curto prazo, R/S se tornava assintóticamente proporcional à $N^{\frac{1}{2}}$. Este parâmetro (veja detalhes de método de estimação de H na Tese de Mestrado UNICAMP [42], e em [43]) foi mais tarde adotado na descrição de tráfego de redes ATM e até para o dimensionamento destas redes [44] [46].

Existem duas definições para processos estocásticos auto-similares (PEF-Processo Estocástico Fractal), a primeira chamada de **definição em tempo contínuo** é que um PEF $\mathbf{x}(t)$ é estatisticamente auto-similar com parâmetro $H\,(0,5 \leq H \leq 1)$ para qualquer real $a > 0$, se o Processo $a^{-H}\mathbf{x}(at)$ tiver as mesmas propriedades estatísticas, com três condições:

$$\begin{aligned} &1)\ \text{Média} \ : && E[\mathbf{x}(t)] = \frac{E[\mathbf{x}(at)]}{a^H} \\ &2)\ \text{Variância} \ : && Var[\mathbf{x}(t)] = \frac{Var[\mathbf{x}(at)]}{a^{2H}} \\ &3)\ \text{Autocorrelação} \ : && R_s(t,s) = \frac{R_s(at,as)}{a^{2H}} \end{aligned} \tag{2.26}$$

ressaltando que $H = 0,5$ indica ausência de auto-similaridade e cujo grau aumenta quando este se aproxima de 1.

Utilizando estas definições considera-se o importante movimento Browniano fracionário (fBm) [47], em função de H, que é definido como: $B_H = Xt^H$ onde X é uma V. A. com média 0 e variância 1 e sua densidade de probabilidade $f_{B_H}(x,t) = \frac{1}{\sqrt{2\pi t^{2H}}} e^{-x^2/2t^{2H}}$ e deduz-se [40] que, em regime estacionário no instante s:

$$1)\ Var[B_H(t) - B_H(s)] = |t-s|^2$$

$$2)\ R_{B_H}(t,s) = \frac{1}{2}\left(t^{2H} + s^{2H} - |t-s|^{2H}\right)$$

Outra definição é a de **tempo discreto**, que considera a série temporal $\mathbf{x}$ como a soma de m-agregadas séries temporais (diversos exemplos em [42]), ou seja $\mathbf{x}_k^{(m)} = \frac{1}{m}\sum_{i=km-(m-1)}^{km} \mathbf{x}_i$ e assim, cada valor de $\mathbf{x}_k^{(m)}$ é uma média temporal de $\mathbf{x}$ e este processo é dito auto-similar com parâmetros β $(0 < \beta < 1)$ se, para todo $m = 1, 2, ...$;

Variância

$$Var\left(\mathbf{x}_k^{(m)}\right) = \frac{Var(x)}{m^\beta} \tag{2.27}$$

Autocorrelação

$$R_{x^{(m)}}(k) = R_x(k) \tag{2.28}$$

o parâmetro β está relacionado com o parâmetro H como $H = 1 - \left(\frac{\beta}{2}\right)$. assim, quando $\beta = 1$,temos um **processo ergódico** [38] [48] estacionário, sendo que a variância da média temporal cai a zero a uma taxa de $1/m$. Para Processos Estocásticos Fractais (PEF), a variância da média do tempo decai mais vagarosamente, ou mais exatamente, um processo x é dito **assintóticamente** PEF se, para um k grande e $m \longrightarrow \infty$, as relações acima forem observadas, principalmente $R_{x^{(m)}}(k) \longrightarrow R_x(k)$

Como vemos, a auto-correlação do Processo Agregado assume a mesma forma que o Processo original, ressaltando que, para processos estocásticos tradicionais geralmente utilizados em dados, a autocorrelação converge a zero quando $m \longrightarrow \infty$. E mais, a variância de $x^{(m)}$, que é $\sim \left(1/m^{\beta}\right)$ decai mais vagarosamente que $(1/m)$, pois em PEF, $m \longrightarrow \infty$ e $\beta < 1$.

Na seqüência destes esclarecimentos, torna-se importante a definição de **dependência de longo prazo**, detalhes em [29] [49] [50] [51] [52] [53] [54], para a modelagem de tráfego ATM, que é dada em função do comportamento da função **auto-covariância** $C(\tau)$ **quando** τ **aumenta.** Para muitos processos (Poisson, por ex.), $C(\tau)$ decai rapidamente com o aumento de τ e quando isto acontece, mais explicitamente, quando $C(\tau)$ decai assintótica-exponencialmente, estes são ditos *Processos de Curto Prazo.*

Em contraste com estes, nos **PEF**'s ocorre a *Dependência de Longo Prazo* (LRD) onde observa-se o decaimento assintótico-exponencial de $C(k)$ com $(-\beta)$ tal que $C(k) \sim |k|^{-\beta}$ com $|k| \longrightarrow \infty$ e $0 < \beta < 1$ refletindo um fenômeno de persistência no tráfego da rede de dados no domínio do tempo.

Quando a LRD é formulada no domínio da freqüência, obtém-se a *densidade espectral de potência* que se reduz também a uma forma assintótica, como uma lei de potência [40] perto da origem: $S(w) \sim 1/|w|^{\gamma}$ com $w \to 0$, $0 < \gamma < 1$ sendo $\gamma = 1 - \beta = 2H - 1.$, sendo que, em tempo discreto:

$$\begin{aligned} S(w) &= \sum_{k=-\infty}^{\infty} R(k)\, e^{-j2kw} \\ S(0) &= \sum_{k=-\infty}^{\infty} R(k) \end{aligned} \tag{2.29}$$

e em contraste, a dependência de curto prazo é caracterizado por uma densidade espectral que converge para um número finito quando $w \longrightarrow 0$ ou quando $H = 0,5$, $\gamma = 0$ e em termos da função de auto-correlação, um infinito valor de S(0) é ocasionado se valores de R(k) não decaem suficientemente rápidos para $k \longrightarrow \infty$, e **isto é utilizado para se testar a auto-similaridade.**

Estas três definições: séries temporais agregadas, dependência de longo prazo e densidade espectral são de certa forma equivalentes, mas podemos utilizar outra caracterização comumente denominada "distribuição pesadamente caldal" (HTD- Heavy-Tayled Distribution), e uma distribuição assim denominada é aquela que, para uma V.A.

X ocorre:

$$1 - F(X) = P\{X > x\} \sim i/x^{\alpha} \text{ com } x \longrightarrow \infty \text{ e } \alpha > 0$$

sendo que nos processos HTD, geralmente ocorrem uma elevada ou infinita variância. Um exemplo deste é dado em Paxson & Floid [29], que é o **processo de Pareto,** muito utilizado em comunicações.

Para **modelagem e estimação de tráfego auto-similar de dados** basta determinar o grau de auto-similaridade deste tráfego, para dar-lhe tratamento diferenciado ou não em relação às técnicas tradicionais de projeto de redes ATM ou somente de controle de tráfego (enquanto I. Norros [47] [55] demonstrava os efeitos da auto-similaridade nas redes, consolidando os estudos iniciados por Leland e outros em [28], Tsybakov e Georganas em [44] [45] [46] [56], aplicaram a teoria em dimensionamento de redes com seus "buffers" e "switches" levando em conta a natureza auto-similar do tráfego).

O primeiro método para se estimar H é o de *plotagem variância-tempo* que utiliza a formulação dada na definição de tempo discreto, ou seja, recorrendo à série temporal agregada $x^{(m)}$ de um processo auto-similar, sua variância para um $m \longrightarrow \infty$:

$$Var\left(x^{(m)}\right) \sim \frac{Var\left(x\right)}{m^{\beta}} \tag{2.30}$$

onde o parâmetro H esta inserido em: $H = 1 - (\beta/2)$, que em termos logarítmicos, fica;

$$\log\left[Var\left(x^{(m)}\right)\right] \sim \log\left[Var\left(x\right)\right] - \beta \log\left(m\right) \tag{2.31}$$

Se, baseados nesta última equação, plotarmos $Var\left(x^{(m)}\right)$ versus m em escalas log-log verificando que $\log\left[Var\left(x\right)\right]$ é uma constante independente de m, o resultado será uma reta com inclinação $-\beta$ e isto pode ser facilmente obtido de dados da série $x\left(t\right)$ gerando processo agregado em diferentes níveis de agregação e computando sua variância. Este método foi utilizado em [28] e [57] encontrando-se valores de β entre -1 e 0 indicando auto-similaridade em alguns processos.

O método de *plotagem* R/S ou estatística R/S para um processo estocástico em tempo discreto $\{x_t, t = 0, 1, 2, ...\}$, a faixa re-escalada de $x(t)$ sobre o intervalo de tempo N é definido como a razão R/S:

$$\frac{R}{S} = \frac{\max\limits_{1 \le j \le N}\left[\sum\limits_{k=1}^{j}\left[X_k - M\left(N\right)\right]\right] - \min\limits_{1 \le j \le N}\left[\sum\limits_{k=1}^{j}\left[X_k - M\left(N\right)\right]\right]}{\sqrt{\frac{1}{N}\sum\limits_{j=1}^{N}\left(X_k - M\left(N\right)\right)^2}} \tag{2.32}$$

$$M\left(N\right) = \frac{1}{N}\sum_{j=1}^{n} X_j \tag{2.33}$$

o numerador é a medida da faixa do processo e o denominador é o desvio padrão amostrado, sendo que, para um processo auto-similar ($H > 0,5$) esta razão apresenta a seguinte característica [42]:

$$R/S \sim (N/2)^H \tag{2.34}$$

O método seguinte é o *estimador de Whittle* originado dos estudos do matemático alemão P. Whittle, em 1953, que visava a resolução de um problema clássico de teoria de processos estocásticos, que é a estimação da potencia espectral S(w) de um processo estacionário x(t) somente em termos de uma simples realização de um segmento finito do processo, ou seja, através apenas de uma amostragem cobrindo um período finito de tempo.

Relembrando, para um processo estocástico estacionário em tempo discreto, a autocorrelação e densidade espectral são definidos como:

$$R(k) = E[X(t) X(t+k)] \, eS(w) = \sum_k R(k) e^{-jwk}$$

se assumirmos que o processo é ergódico em correlação ou seja, as médias de tempo são iguais às médias de uma "ensemble" amostrada, a função de autocorrelação será;

$$\hat{R}_N(k) = \frac{1}{N} \sum_{n=0}^{N-1} X(n+k) X(n) \tag{2.35}$$

Desde que a densidade espectral $S(w)$ é a transformada de Fourier da função de autocorrelação $R(k)$, ao aplicarmos sobre um processo x(t) definido sob as instâncias de tempo discreto $\{x_t, t = 0, 1, 2, ...\}$, obtemos sobre um período de tempo N:

$$I_N(w) = \frac{1}{2\pi N} \left| \sum_{k=1}^{N} x_k e^{jkw} \right|^2 \tag{2.36}$$

relação esta conhecida como *períodograma* ou *função intensidade.*

Suponhamos agora que a série temporal observada é de um PEF com parâmetro H e que um formato fBn é achado e expresso como $S(w, H)$, onde a forma da densidade é conhecida, mas o parâmetro H não. Então, pode ser mostrado que H pode ser estimado [5] determinando o valor de H que minimiza a expressão:

$$\int_{-\pi}^{\pi} \frac{I_N(w)}{S(w, H)} dw$$

este é conhecido como estimador de Whittle estudado em [49] [58] [59].

Se a seqüência $\{x_k\}$ tem comprimento N, então a integral pede ser entendida como uma soma discreta sobre as freqüências $w = 2\pi/N, 4\pi/N, ...2\pi$ e o estimador é assintóticamente normal, então chega-se a:

$$Var\left(\hat{N}\right) = 4\pi \left[\int_{-\pi}^{\pi} \left(\frac{\partial \log S(w)}{\partial H} \right)^2 dw \right] \tag{2.37}$$

a vantagem desta abordagem é que ela produz não só uma estimativa de H, como também sua variância e serve para plotar tanto gráficos de variância-tempo como R/S.

2.3. Conclusão

Enquanto o **controle de congestionamento** procura minimizar ou evitar os efeitos de forte tráfego proveniente do nó do usuário, em relação a capacidade da rede, o **policiamento** de tráfego (também uma forma de controle) garante que o fluxo de células ATM estejam em conformidade de acordo com os parâmetros QoS para cada tipo de serviço na rede ATM.

Pelo exposto, é importante a caracterização das fontes de tráfego (nó do usuário) por modelos apropriados para garantir um tratamento diferenciado para as diversas modalidades de serviços oferecidos pelas redes ATM.

Pelos efeitos já citados, o parâmetro de auto-similaridade H deve ser encarado como um "divisor de águas" na caracterização das fontes de tráfego e sua estimação assume grande importância para encaminhamento de tráfego ao controlador de congestionamento apropriado. Como o **gerenciador de recursos-RM** administra genericamente os recursos da rede ATM, a nível de caminho virtual (englobando vários circuitos virtuais), o controlador apropriado mais próximo a nível de circuito virtual é o **controle de admissão de chamadas-CAC**.

Portanto torna-se prudente que o tráfego deva ser encaminhado ao CAC que mais se ajuste às suas características de auto-similaridade antes de compor um tráfego heterogêneo em fluxo na rede ATM, e o CAC apropriado e os efeitos do parâmetro H serão objetos de estudo nos próximos capítulos deste trabalho.

Capítulo 3

Descrição dos algoritmos de CAC

3.1. Classificação dos esquemas de CAC

Controle de Admissão de Chamadas - CAC objetiva controlar a admissão de novas conexões, ou seja, permitir estabelecer novos caminhos virtuais na rede ATM para novos usuários, promovendo o controle de congestionamento para garantir os requisitos de qualidade de serviço (QoS) estabelecidos pelo ATM FORUM.

Como já mencionado, com a expansão das redes ATM, tanto em número de usuários por rede quanto em taxa de transmissão por usuário (multiserviço), tornou-se necessário o aprimoramento dos métodos de controle de admissão de novas conexões integrado com o dimensionamento apropriado dos "buffers" de entrada desta rede. Tudo isto para evitar o descarte de células além do permitido pela QoS.

Na parte introdutória deste trabalho mencionou-se as conseqüências de uma forte expansão de tráfego ATM devido ao crescente número de usuários de dados, e em função deste fato nestes últimos anos um substancial número de esquemas de CAC's tem sido propostos para redes ATM visando a melhoria desta forma de controle.

As principais classificações para estes esquemas são fornecidos por [4] [5] [60] e sendo que neste último já se começa a incorporar as recentes pesquisas sobre características de auto-similaridade de tráfego ATM.

Em Giordano e outros [61], encontramos esquemas de CAC baseados plenamente na natureza auto-similar do tráfego multimídia ATM e, em anos recentes, tem sido dada ênfase na ampla incorporação do fenômeno da auto-similaridade em conjunto com a utilização de sistemas Neuro-Fuzzy em esquemas de CAC.

Seguindo então uma certa ordem cronológica, dentre vários autores (para justo registro, estes estudos se iniciaram em 1991, na IBM, com Guérin, Ahmadi e Naghshineh - vide [62] para detalhes) e principalmente hoje Onvural [4], Virtamo [5] e Elwallid [60], classificam os esquemas de CAC em termos de alocação de largura de banda (BW).

A cada conexão envolvida no processo de admissão deve ser atribuída uma largura de banda compatível e que satisfaça os requisitos de QoS pretendidos. Isto deve ocorrer com todo o conjunto das conexões em tráfego, ou seja, considera-se neste tipo de processo

a chamada a ser admitida e as já em tráfego na rede ATM.

Neste enfoque de alocação de banda, os esquemas de CAC se classificam como **determinísticos** ou **estatísticos**.

Na alocação ou multiplexação estatística, a largura de banda é alocada para uma fonte e denomina-se **largura de banda estatística** (EBW) que é sempre menor que a largura de banda de pico, mas sempre maior que a largura de banda média desta fonte. Neste caso, em um enlace, a soma das taxas de pico das conexões admitidas pode ser maior que a largura de banda total deste enlace, mas a soma das larguras de banda estatística é sempre menor ou igual que a largura de banda estatística do enlace.

Podemos afirmar que existe uma "eficiência estatística" nesta alocação e que esta aumenta se a EBW de cada conexão aproximar-se de sua taxa média de bit, caso contrário, se a EBW de cada conexão crescer e se aproximar das suas taxas de pico de bit, esta eficiência diminui.

A principal dificuldade de se calcular o valor da EBW para uma conexão é a continuidade e manutenção das garantias de QoS, tanto para a nova conexão que está sendo admitida, quanto para a condição imediatamente posterior a esta admissão, pois a EBW que será atribuída não depende somente das características estocásticas da chamada em admissão, mas da característica "conjunta" das chamadas em andamento na rede ATM

Segundo Onvural [4], a eficiência ou ganho estatístico que justifica a alocação de EBW é resultado de avaliação e comparação de várias características da chamada e parâmetros da rede, como ilustra a Figura 3.1.

Assim, a alocação determinística é recomendada quando a eficiência estatística é baixa, ou seja, a razão entre a taxa de pico de bit e a sua taxa média é baixa; ou mesmo que esta seja mais alta, a razão entre a taxa de pico de bit e a capacidade do enlace seja alta (isto implica que dispomos de capacidade "de sobra" no enlace) ou mesmo que esta seja baixa, se o comprimento dos **surtos** for muito longo, haverá poucos "**slots**" a serem estatisticamente aproveitados, aproximando perigosamente a taxa de pico da alocação da EBW.

Mas a tendência que se verifica é que as fontes estão sendo codificadas de tal maneira, que o tráfego agregado tenha surtos menores e sempre com um grande número de fontes, fazendo com que a capacidade do enlace seja aproveitada ao máximo, tudo indicando adoção da alocação estatística de banda.

Observa-se que o comprimento de *surto* é decisivo e que, estatisticamente, quando o tráfego é heterogêneo, as características das fontes com taxas elevadas são predominantes em relação às taxas mais baixas, por isto a influência do tráfego composto de vídeo, que requer taxas mais elevadas, e sua codificação são determinantes na utilização de um ou outro método de alocação de banda. De forma geral, a classificação e os algoritmos disponíveis estão mostrados na Figura 3.2.

Geralmente, nas referências, o único método de alocação de banda determinística é o de **alocação por taxa de pico**, mas adotamos neste trabalho, como uma

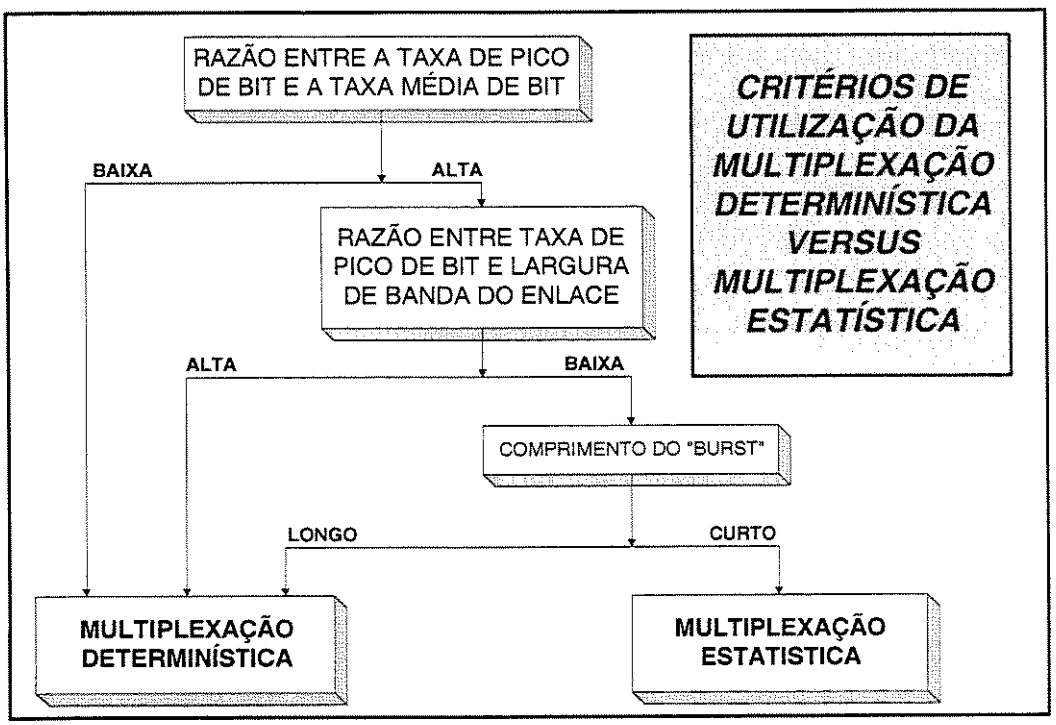

Figura 3.1: Critérios de alocação Determinística e Estatística

proposta nova, os algoritmos predominantemente determinísticos como determinísticos e os predominantemente estatísticos como estatísticos, mesmo que um apresente alguma característica que indique a classificação de outro.

A seguir,a descrição de cada um dos onze algoritmos CAC escolhidos, dentro de um contexto didático, dentre muitos outros que hoje existem;

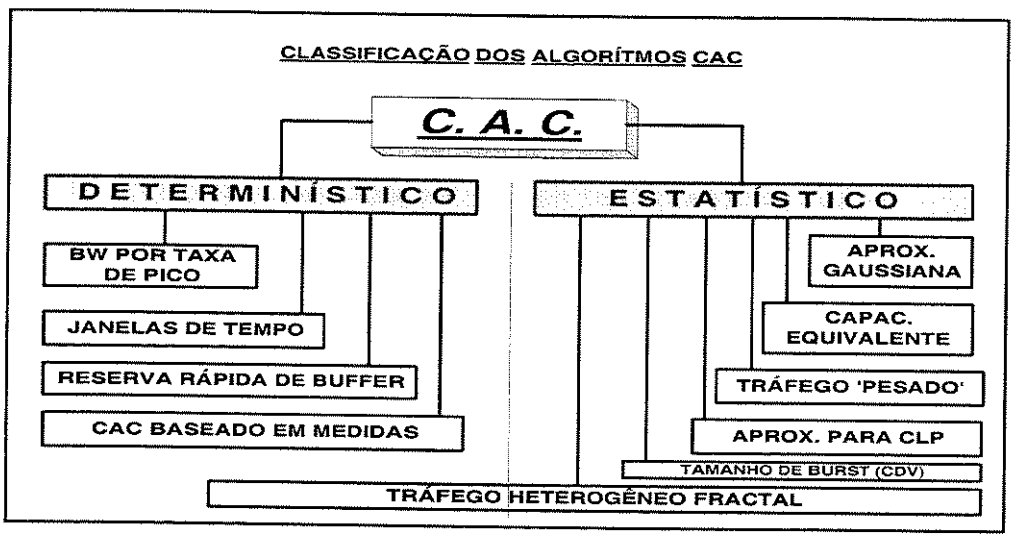

Figura 3.2: Classificação do Esquemas de CAC

3.2. Alocação determinística de banda;

Nesta classificação, os esquemas se caracterizam pela alocação de BW de forma determinística ou seja, os métodos para subsidiar a decisão de admissão de chamadas são predominantemente determinísticas. e seguem um padrão previamente determinado pela rede ATM, apresentada a seguir:

3.2.1. Alocação por taxa de pico;

Neste esquema de alocação determinística, podemos afirmar que, na multiplexação determinística na rede ATM, a camada de gerenciamento de rede aloca a cada conexão sua largura de banda de pico correspondente, garantindo que os requisitos QoS serão assegurados em 100% do tempo da conexão.

Por outro lado, isto causa grande desperdício do montante de largura de banda disponível da rede, principalmente para conexões de tráfego em *surtos* que apresentam elevada razão numérica entre as taxas de bit de pico e taxa média de bit, indicando intervalos de *"time slots"* ou simplesmente *"slots"* que poderiam ser aproveitados para outras conexões e que desta forma não podem ser admitidas na rede. Verficou-se ainda que, apesar deste método, reduzir o nível de congestionamento a quase zero, com a desvantagem do fator econômico há ainda o fato de haver probabilidade não-zero de sobrecarga nos "buffers", que determinou a busca de outras alternativas.

3.2.2. Janelas de tempo

Este esquema de CAC tem por princípio básico a armazenagem e transmissão em períodos determinados de tempo ("janelas de tempo"), análogos ao TDM. Além do

"buffer" do enlace, existem "buffers" para cada fonte na entrada da rede. Para cada fonte é dada permissão para transmitir, descarregando seu "buffer" numa janela de tempo, quando a taxa resultante for um valor limite para a rede ATM ou até ser taxa de pico de célula do enlace- PCR_{link}.

Assim, neste método,todas as fontes ou conexões em tráfego ou em admissão somente poderão transmitir em "sua" janela de tempo, com uma taxa uniforme para todas as conexões, resultando em um fluxo contínuo e uniforme por toda a rede ATM.

Observemos que as conexões em tráfego são des-sincronizadas com suas respectivas fontes e sua implementação tornou-se complexa com a interconexão de várias redes ATM entre si, ou mesmo em uma grande rede ATM, com muitas fontes e com tráfego heterogêneo com taxas muito diferenciadas dificultando o processamento para o efetivo controle.

Então, posteriormente, o método foi aperfeiçoado para que o controle fosse feito nó-a-nó da rede ATM, ou seja, para cada conexão, o nó receptor permite a liberação de células pelo nó transmissor somente acima de um limite de taxa a cada janela de tempo. O número de células é regulado utilizando o artifício de "créditos" dentro de uma "janela de tempo", assim, se uma conexão não conseguir transmitir todos os seus créditos em uma janela de tempo, ficará com créditos para a próxima janela.

Por outro modo, se uma chamada exaurir seus créditos antes do término da janela de tempo, nenhuma célula será transmitida até o término da janela, fazendo que, na média de créditos de um e outro caso, o limite de taxa do nó seja preservado.

O princípio de utilização de créditos pode ser mais detalhadamente estudado em [65].

3.2.3. Reserva rápida de "buffer"

Esta técnica é desenvolvida a partir dos princípios de gerenciamento e reserva de recursos, que já vinham sendo empregados em controle de tráfego em roteadores de redes ATM [9] [22], aliados a técnicas de CAC determinísticas ou estatísticas a nível de conexão, como mostrado na Figura 3.3.

Nesta Figura observamos que além da seleção das rotas de tráfego realizadas pelo roteamento e a base de dados de rotas, através de células OAM (células da camada de gerência de rede ATM) a reserva de recursos e o agente de gerência são alimentados por informações sobre o tráfego.

As bases de dados de rotas e de controle de tráfego são constantemente atualizadas com informações de toda a rede ATM em tempo real. A reserva de recursos utiliza um protocolo denominado RSVP (ReSerVation Protocol-Protocolo de Reserva de Recursos), que é responsável por reservar previamente largura de banda por todo o caminho da rota, dado um novo fluxo de pacotes requerendo garantias QoS.

Garantidas as reservas de banda, o CAC decide através de outros parâmetros, se este fluxo será aceito ou não. Após esta etapa as bases de dados são novamente

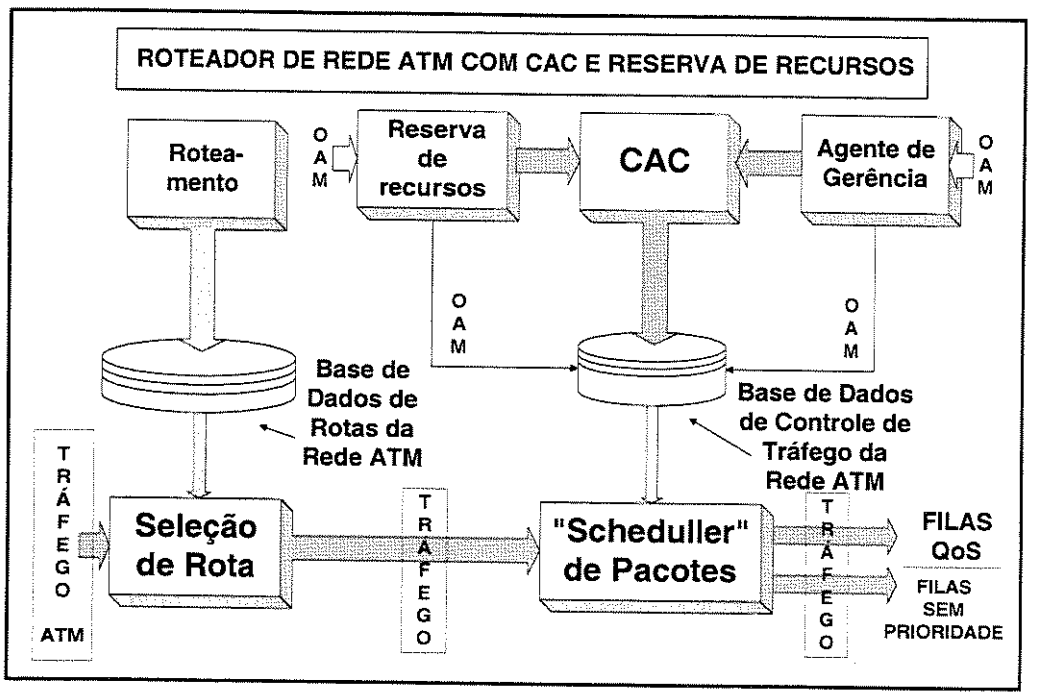

Figura 3.3: Roteador ATM com CAC e reserva de recursos

atualizados, e assim por diante, mas tudo a nível de chamada, ignorando os níveis de surto e célula.

Por isso, como observa Elwallid [60], **nenhum recurso é alocado nos "buffers" deste caminho virtual à nível de surto de tráfego causando perdas destes e até invalidando conexões.** Para resolver esta questão, utilizou-se de princípios análogos aos de reserva de recursos mas **a nível de surto**, ou seja, quando uma fonte está pronta para transmitir um surto, neste instante a rede ATM aloca os recursos necessários nos "buffers" para a duração deste surto.

Este esquema apresenta principalmente características determinísticas. através de um algoritmo que utiliza células marcadas, mas também apresenta características estatísticas ao se utilizar de probabilidade de demanda ou de contenção do "buffer", como auxílio à decisão de admissão.

Em termos genéricos, nesta técnica uma fonte em surtos é caracterizada por um modelo Markoviano de dois estados. Quando esta se torna ativa, um pré-especificado número de "slots" de "buffer" no "buffer" do enlace é reservado para a duração deste período ativo. ao final deste período ativo, todos os "slots" são liberados. Este processo é repetido ao longo de toda a duração da chamada ou conexão.

Enssle, Briem e Kröner em [66], apresentam uma análise completa de um protocolo de reserva rápida para redes ATM, enfocando seu desempenho, com resultados numéricos favoráveis em relação aos métodos de alocação estatística.

Segue então, uma descrição sintetizada do método, utilizando células marcadas

para especificar as transições entre os estados ativo e de silêncio.

Definindo as variáveis abaixo:

B : número de *"slots"* de "buffer", no *"buffer"* de enlace;

B_i : número de *"slots"*de "buffer" para serem reservados para a conexão i;

b_i : número de *"slots"*em uso corrente;

S_i : estado da conexão i (ativo ou inativo);

a operação do **algoritmo de reserva rápida** pode ser sumarizada como segue:

R - 1) Reduza b_i de 1 (um) "slot" toda vez que uma célula pertencente à conexão i é transmitida a partir do "buffer";

R - 2) Se a conexão está no estado de silêncio e:

R - 2.1) Se é recebida uma célula indicando o início de um período ativo e;

R - 2.1.1) Se $B_i \geq B$, a célula é dercartada e nenhuma reserva é feita.

R - 2.1.2) Se $B_i < B$ então;

A - Um contador de tempo de ocupação de "buffer" é ativado;

B - S_i é mudado do estado de silêncio para ativo;

C - Aciona-se um contador $B = B - B_i$;

D - Se $b_i < B_i$, então;

D.1) $b_i = b_i + 1$;

D.2) Coloque uma célula no "buffer" como uma célula não marcada;

E - Se $b_i = B_i$, coloque a célula no "buffer" como marcada;

R - 2.2) Se a célula recebida não especifica o início do período ativo, ela é descartada;

R - 3) Se a conexão já se encontra em estado ativo, então;

R - 3.1) Se a célula recebida não especifica o fim do estado ativo, então;

R - 3.1.1) O contador de tempo de "buffer" é zerado;

R - 3.1.2) Se $b_i < B_i$, então;

A - $b_i = b_i + 1$;

B - Coloque a célula no "buffer" como uma célula não marcada;

R - 3.1.3) Se $b_i = B_i$, então coloque a célula no "buffer" como uma célula marcada;

R - 3.2) Se a célula recebida especifica o fim de um período ativo ou o contador do "buffer" expira, então;

R - 3.2.1) Muda-se o estado de S_i de ativo para silêncio;

R - 3.2.1) $B = B + B_i$

Assim, um contador de tempo é usado para forçar o retorno para o estado de silêncio e garantir que os "slots" reservados sejam liberados. Os "slots" no "buffer" são reservados somente no momento da recepção de uma célula, denominada célula-de-início-de-surto que indica o início de período ativo.

Se o número de "slots" requeridos no "buffer" não estiver em disponibilidade, então todas as células da conexão são descartadas até que outra célula-de-início-de-surto seja recebida.

Os surtos de cada aplicação são transmitidos por células marcadas início-de-surto, seguidas de células intermediárias e finalizadas por células marcadas por fim-de-surto. Com este esquema, se o "buffer" não tiver espaço disponível para a primeira célula então, como discutido, todo o surto é descartado.

O algoritmo FBR pode comportar tráfego tolerante à perda com relativa facilidade. Neste caso, cada célula de surto, exceto a última é transmitida como célula-de-início-de-surto. Se uma célula é perdida, a célula consecutiva ainda tem a chance de entrar no "buffer". Em adição aos três tipos de células marcadas indicando o início, fim e meio de um surto, células marcadas solitárias são definidas para especificar células de baixa prioridade, indicando agora a classificação em: aplicações sensíveis à perdas (alta prioridade), tolerantes à perdas (média prioridade) e de baixa prioridade.

A definição de quatro tipos de células requer o uso de dois bits no cabeçalho da célula ATM e isto pode ser feito pelos dois últimos bits do campo VCI (Identificador de Canal Virtual-vide Célula ATM) mas, como sendo um procedimento não padronizado podem ocorrer problemas na interconexão de redes ATM.

Para remediar, opta-se pela utilização do campo do bit CLP como segue: quando em estado de silêncio, uma célula com bit CLP "1" é tratada como solitária, caso contrário ela será tratada como célula de início de surto. Similarmente, quando em estado ativo, uma célula com bit CLP "1" é tratada como célula de fim de surto, caso contrário é tratada como intermediária.

Com este esquema, células de baixa prioridade não podem ser enviadas dentro de um surto. Além disto, a rede não pode filtrar ou cortar surtos, pois isto resultaria em diminuição de eficiência desta rede.

Para decidir se uma conexão pode ser multiplexada com as conexões já em tráfego em um enlace, calcula-se a probabilidade de excesso de demanda ou a probabilidade de se requerer mais "slots" do "buffer" que o disponível.

Se a probabilidade de excesso de demanda é maior que um valor pré-definido, a nova conexão é rejeitada, caso contrário, ela é aceita.

Então, se chamarmos:

R_i=Taxa de pico de célula da conexão i

m_i=Taxa média de célula da conexão i

X_i=Variável aleatória representando o número de "slots" de "buffer" necessários para a conexão i, assume valor "0" para o estado de silêncio e B_i para o estado ativo;

$$P\{X_i = B_i\} = m_i/R_i \qquad e \qquad P\{X_i = 0\} = 1 - m_i/R_i \tag{3.1}$$

Usando a razão taxa-de-pico-para-enlace, o número de "slots" do "buffer" requerido por uma fonte ativa é assumida para ser igual a:

$$B_i = \lceil M\frac{m_i}{C} \rceil \tag{3.2}$$

onde $\lceil Z \rceil$ é o menor inteiro maior ou igual a Z, M é o número total de "slots" no "buffer" (tamanho do "buffer") do enlace e C a capacidade do enlace ou taxa do enlace.

Considere agora que o enlace transporta N conexões correspondendo às demandas $X_1, X_2, ..., X_N$. A demanda total do "buffer" X é a soma de N variáveis aleatórias, isto é:

$$X = \sum_{i=1}^{N} X_i \tag{3.3}$$

Considerando que os X_i's são mutuamente independentes, a função geração de probabilidades de X, $f_x(z)$, é:

$$f_x(z) = \prod_{i=1}^{N} \{(1 - p_i) + p_i z^{B_i}\} = A_0 + A_1 z + A_2 z^2 + ... + A_K z^K \tag{3.4}$$

onde p_i é a probabilidade de ocorrer a conexão i e $K = \sum_{K=1}^{N} B_i$;

Temos que $P\{X = j\} = A_j$ denota a probabilidade que a demanda total do "buffer" seja igual a A_j. A probabilidade de excesso de demanda é igual a:

$$1 - P\{X = j\} = 1 - \sum_{i=1}^{L} A_j \tag{3.5}$$

e a questão se resume, então, em se obter os valores de A_j's;

Se uma nova conexão solicita uma demanda X_{N+1} do "buffer", para que seja também multiplexada com as N conexões existentes, então a nova função geradora de probabilidade $f_x^*(z)$ será.

$$f_x^*(z) = \prod_{i=1}^{N+1} \{(1 - p_i) + p_i z^{B_i}\} = \{(1 - p_{N+1}) + p_i z^{B_{N+1}}\} \prod_{i=1}^{N} \{(1 - p_i) + p_i z^{B_i}\} \ , \tag{3.6}$$

o que significa que os novos coeficientes A_j's podem ser extraídos [67] a partir dos prévios A_j's (provenientes das N conexões já multiplexadas), ou seja:

$$A'_j = (1 - p_{N+1})\, A_j + p_{N+1} A_{j-B_{N+1}} \tag{3.7}$$

para todo j, ou seja, quando os "buffers" da conexão i são liberados, os novos coeficientes são calculados dos atuais, e assim $A'_j = A_j(1-p_i) + A_{j-B_i}/p_i$ para todo j, e para $j < 0 \Longrightarrow A_j = 0$.

Entretanto, para que não haja anomalias devido ao fato de várias probabilidades $p'_i s$ concorrerem não simultaneamente ao mesmo "buffer" resultarem em disponibilidade menor que a calculada de forma como se ocorressem simultaneamente, utiliza-se a probabilidade de que o número de "slots" no "buffer" sendo requeridos não esteja disponível no tempo em que a fonte i transmite seu surto.

Esta probabilidade é limitada acima, pela probabilidade $P\{(X - X_i) > (L - B_i)\}$ e é denominada como probabilidade de contenção da conexão i que, em [67] é obtida como:

$$P\{(X - X_i) > (L - B_i)\} \leq (1/p_i)\, P\{X > L\} \tag{3.8}$$

e, conseqüentemente, **desde que p_i não seja muito pequeno, a probabilidade de excesso de demanda não é muito maior que a probabilidade de contenção** (os dois diferem muito quando p_i diminui muito).

Quando p_i é pequeno, apertados limites são obtidos sobre a **probabilidade de contenção** são obtidas:

$$P\{(X - X_i) > j\} = \{1/(1-p_i)\} \sum_{h=o}^{K-1} \{-p_i/(1-p_i)\}^h P\{X \rangle j - hB_i\} + \\ + \{-p_i/(1-p_i)\}^K P\{X \rangle j - hB_i\} \tag{3.9}$$

onde K é agora determinado fazendo com que $\{p_i/(1-p_i)\}^K$ seja suficientemente pequeno.

O procedimento de CAC, neste contexto, é baseado na condicional da probabilidade de excesso de demanda quando p_i é pequeno ou sobre a probabilidade de contenção se p_i é grande.

Ou seja: **Se p_i é pequeno então $p = P\{X > L\}$, caso contrário, se p_i não for desprezível, $p = P\{(X - X_i) > j\}$. Para os dois casos, se $p < (CLP)_{QoS}$ a nova conexão será aceita, caso contrário, será rejeitada.**

3.2.4. CAC baseado em medições

Este método surgiu da constatação de que é impossível a modelagem de tráfego para CAC que forneça completa precisão. As medições efetuadas em tempo real podem subsidiar a decisão de admissão, com melhor precisão no que se refere à previsão de recursos necessários para manter a QoS global da rede ATM.

Um dos métodos, proposto por Kesidis e outros [101], trabalha com estimativas de $CLP(N+1, M, C)$ baseados nas medições em tempo real de $CLP(N, M, C)$ sobre o "buffer", sendo N o número de conexões em tráfego, M o tamanho do "buffer" e C a

capacidade do enlace. Saito [76], utiliza a medida do número de células que chegam em um intervalo fixo de tempo aliado a descritores de tráfego para estimar o $CLP(N+1)$.

Outro método mais recente baseado em medições "on line" é o proposto por Bensaou e outros [97], que emprega um **algoritmo em lógica FUZZY**, para incrementar a decisão de admissão da chamada. O estimador FUZZY emprega uma aproximação de CLP baseada em duas escalas diferentes de eventos; uma escala se refere a "buffers" pequenos, baixa taxa de serviço, etc; e a outra escala se refere ao comportamento assintótico um "buffer" grande com taxas elevadas e muitos terminais.

Conforme Figura 3.4, este método utiliza medidas em pequenos intervalos de tempo reduzindo o "buffer" principal em pequenos "buffers" virtuais com capacidade do enlace também particionada e é observada a perda de células CLP_i sob pequena variância nesses "buffers" virtuais. Estas medidas subsidiam um estimador Fuzzy que fornece uma BW equivalente para o modulo de decisão. Este decidirá de acordo com a capacidade do enlace e CLP(QoS).

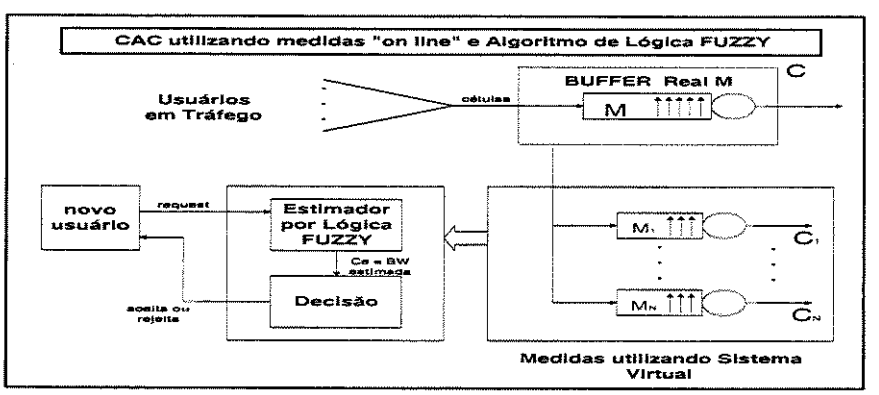

Figura 3.4: CAC baseado em medidas com estimação FUZZY

3.3. Alocação estatística de banda;

Os métodos a seguir visam empregar fatores de decisão estatísticos para a admissão de chamadas e, conseqüentemente otimizando o aproveitamento dos recursos da rede ATM.

Pelo fato de que, nestas redes, o tráfego heterogêneo multimídia estar prevalecendo sobre o tráfego proveniente de fontes homogêneas, estes métodos ganham espaço cada vez maior, pois a alocação determinística subutiliza em demasia estes recursos.

3.3.1. Aproximação gaussiana;

Apesar de pouco adotado atualmente devido às suas limitações, serviu de "plataforma" para o desenvolvimento de algoritmos mais avançados. Neste algoritmo, cada conexão é caracterizada por sua taxa média de bit m_i e o desvio padrão desta taxa σ_i.

Com n conexões multiplexadas, a questão é a determinação da banda requerida C_0 por estas n conexões e a probabilidade que a taxa de bit instantâneo agregada A, maior que C_0, seja menor que um dado valor ε.

Como vimos, A é a variável aleatória associada à taxa de bit instantâneo agregada de n conexões multiplexadas, temos que determinar:

$$P\{A > C_0\} \leqslant \epsilon \tag{3.10}$$

Assumindo que a taxa de bit agregado tenha distribuição gaussiana (daí o nome do algoritmo) teremos:

$$P\{A > C_0\} = P\{(A-m)'\sigma > (C_0-m)'\sigma\} \approx P\{A_{01} > (C_0-m)'\sigma\} = P\{A_{m\sigma} > C_0\}$$

onde: $m = \sum_{i=1}^{n} m_i, \sigma^2 = \sum_{i=1}^{n} \sigma_i^2$ e A_{01} e $A_{m\sigma}$ são variáveis aleatórias gaussianas com médias e desvios-padrão $(0,1)$ e (m,σ), respectivamente, então de [62], temos que:

$$P\{A > C_0\} \approx P\{A)m + \alpha\sigma\} \approx \epsilon \tag{3.11}$$

ou seja, foi aproximada $C_0 \approx m + \alpha\sigma$, sendo que α é o inverso da distribuição gaussiana e que, de [62] também obtemos o valor bem aproximado de α;

$$\alpha = \sqrt[2]{(-2ln\epsilon - ln2\pi)} \tag{3.12}$$

na prática, C_0 é o limite superior da banda instantânea requerida para que o tráfego tenha probabilidade de perda de célula menor ou igual à ϵ $(CLP \leqslant \epsilon)$.

Observa-se que o tamanho do "buffer" não é considerado no cálculo de C_0 e que, na prática a taxa agregada instantânea de bit excede C_0 por um período de tempo até o "buffer" encher, absorvendo em parte a imprecisão introduzida pelo método.

Esta taxa freqüentemente exibe uma longa "calda" na distribuição quando esta, assumida gaussiana, não condizer com a realidade. Neste caso $P\{A > C_0\} > \epsilon$, quando

C_0 é aproximado para $(m + \alpha\sigma)$. Para resolver este problema, a abordagem é estendida como segue.

A distribuição de taxa de bit de cada conexão é assumido como sendo "embutida" em envoltória gaussiana de parâmetros (m_i*, σ_i*) sendo que; $m_i^* = m_i + a\sigma_i$, $\sigma_i^* = b\sigma_i$ ou seja, o montante de largura de banda reservada para n conexões multiplexadas (para "algumas" constantes a e b) são dadas por:

$$C_0 = \sum_{i=1}^{n} m_i + a\sigma_i + \alpha b \sqrt{\sum_{i=1}^{n} \sigma_i^2} \tag{3.13}$$

para se determinar a e b, é necessário trabalhar com estimativas de medidas através de monitoramento de cada fonte individualmente (o que pode ser feito pelo uso de células especiais, vide Chen e outros[68]).

3.3.2. Alocação por capacidade efetiva;

Desenvolvido e apresentado por Guérin e outros, [62], que teve como base um artigo de D. Mitra e outros, [36] e, neste método, uma conexão é caracterizada por um modelo de dois estados em que o fluxo de bits é gerado a uma taxa de bits **de pico** durante o período ativo, enquanto nenhum bit é gerado no período de silêncio.

Ao contrário de outras técnicas que serão discutidas, este método é baseado no modelo de fluxo. Em geral, temos Z uma V.A. representando a taxa de bit de uma fonte alimentando um link com um "buffer" finito e uma velocidade de transmissão C.

A dinâmica da fila no modelo de fluxo é definida como segue;

E1 - Se $Z = z < C$ e;

E1.1 - O "buffer" está vazio, por conseguinte ele permanece vazio.

E1.2 - O "buffer" não está vazio, então seu conteúdo diminui a uma taxa constante $C - z$;

E2 - Se $Z = z = C$, então o conteúdo do "buffer" não se altera;

E3 - Se $Z = z > C$ e:

E3.1 - O "buffer" não está cheio, então seu conteúdo cresce a uma taxa constante de $z - C$;

E3.2 - O "buffer" está cheio, então as células são perdidas a uma taxa constante de $z - C$.

Chamemos:

R_i:Taxa de bit de pico da conexão i;
m_i: Taxa média de bit da conexão i;
b_i: Duração média do período ativo ou comprimento médio do *surto* ;
ρ_i: Fator de utilização da fonte ou a probabilidade em que ela esteja no período ativo;
μ_i: Taxa de transição de saída do estado ativo ($\mu = 1/b$);
λ_i: Taxa de transição de saída do estado de silêncio ($\lambda = \rho/[b(1-\rho)]$);
C: Taxa de capacidade do enlace;
M: Capacidade do "buffer", em células;

Como dissemos, assumindo que cada fonte é caracterizada como um modelo de dois estados ON-OFF e a duração de cada estado é exponencialmente distribuída e independente um do outro, o montante de BW requerida por uma conexão é estimado em [62], que responde a seguinte questão:

"Se uma conexão com parâmetros (R, m, b) entra em um enlace com "buffer" de capacidade M, qual deverá ser a taxa deste enlace para que a probabilidade de *"overflow"* no "buffer" seja menor que ϵ ?"

A resposta encontrada foi:

$$c_i = R_i \frac{y_i - M + \sqrt[2]{(y_i - M)^2 + 4M\rho_i y_i}}{2y_i} \text{ sendo } y_i = \alpha(1-\rho_i)R_i \tag{3.14}$$

nesta direção, o BW total C_e de n conexões multiplexadas é igual à soma das capacidades equivalentes das conexões individuais c_i, isto é:

$$C_e = \sum_{i=1}^{n} c_i \tag{3.15}$$

Entretanto, como a interação entre as conexões individuais não é considerada, a BW total de n conexões multiplexadas acaba sendo sobreestimada e o valor de C acaba sendo maior que o necessário, por isso utiliza-se a aproximação Gaussiana em conjunto com a capacidade equivalente, ou seja:

$$C_e = min\{(m + \alpha\sigma) \ , \ \left(\sum_{i=1}^{n} c_i \right)\} \tag{3.16}$$

sendo $\alpha = \sqrt[2]{(-2ln\epsilon - ln2\pi)}$ como na aproximação gaussiana.

Para as conexões individuais:

$$m_i = R_i b_i \Longrightarrow m = \sum_{i=1}^{n} m_i \tag{3.17}$$

$$\sigma_i^2 = m_i(R_i - m_i) \Longrightarrow \sigma^2 = \sum_{i=1}^{n} \sigma_i^2 \tag{3.18}$$

Para tráfego auto-similar ($H_i > 0,5$), Tsybakov em [44] e Norros em [55] desenvolveram modelos para **capacidade efetiva em função do parâmetro** H, de Hurst (veja ítem 2.2.6) que é dada por:

$$c_i = m_i + \left(H_i^{H_i}\,(1-H_i)^{(1-H_i)}\,\sqrt[2]{-2ln\epsilon}\right)^{\frac{1}{H_i}} a_i^{\frac{1}{2H_i}} M^{-\frac{(1-H_i)}{H_i}} m_i^{\frac{1}{2H_i}} \tag{3.19}$$

onde:

m_i: Taxa média de bit da conexão i;
H_i: Parâmetro de Hurst da conexão i;
ϵ: Limitação superior de CLP para garantir QoS;
a_i: coeficiente de variação(ou variância σ^2) da conexão i;
M: Capacidade do "buffer", em células;

utilizando a expressão de C_e, os passos para admissão de chamada (AC) são:

AC1) Dados os parâmetros $(m_{n+1}, R_{n+1}, b_{n+1})$ de uma conexão que requer admissão

e os valores correntes das conexões em tráfego m, σ, e $\sum_{i=1}^{n} c_i$;

AC1.1) Estime o parâmetro de Hurst H_i da conexão i;

AC1.2) Se $H_i \approx 0,5$, calcule c_{n+1}pela equação 3.14 ou se $H_i > 0,5$ deve-se calcular c_{n+1} pela equação 3.19;

AC1.3) Calcule os novos valores de $m' = m + m_{n+1}$ e
$\sigma'^2 = \sigma^2 + m_{n+1}(R_{n+1} - m_{n+1})$

AC2) Calcule C_e' da situação pós-admissão em condição de evento simulado, levando em conta os valores de 1.2) e 1.3);

AC2.1) Se $C_e' < C$ do enlace, a chamada é aceita; caso contrário será descartada.

Obs.: A equação 3.19 pode ser escrita como:

$$\begin{aligned} c_i &= m_i + f^{-1}(\epsilon) \times \gamma \\ f^{-1}(\epsilon) &= \left[H_i^{H_i}\,(1-H_i)^{(1-H_i)}\,\sqrt[2]{-2ln\epsilon}\right]^{\frac{1}{H_i}} \\ \gamma &= a_i^{\frac{1}{2H_i}} M^{-\frac{(1-H_i)}{H_i}} m_i^{\frac{1}{2H_i}} \end{aligned}$$

e que, preparada para simulação numérica para o próximo capítulo, normalizando-se c_i, m_i, a_i pela capacidade total do enlace C (que no caso da simulação numérica, cap 4, será 155Mbps) obtém-se $c_k = \frac{c_i}{C}$, $m_k = \frac{m_i}{C}$ e $v_k = \frac{\sqrt[2]{a_i}}{C}$ e que:

$$\begin{aligned} c_k &= m_k + f^{-1}(\epsilon) \times \gamma_s \\ f^{-1}(\epsilon) &= \left[H_i^{H_i}\,(1-H_i)^{(1-H_i)}\,\sqrt[2]{-2ln\epsilon}\right] \\ \gamma_s &= v_k^{\frac{1}{H_K}} m_k^{\frac{1}{2H_k}} \left[C/(424 \times M)\right]^{\frac{1-H_k}{H_k}} \end{aligned} \tag{3.20}$$

onde $H_k = H_i\ C$ é dado em Mhz e M é dado em células ATM.

Diversos autores trabalharam com estes conceitos, em Elwallid e Mitra [69] foi dada uma nova abordagem, mostrando que a capacidade equivalente de uma fonte fluida modulada Markoviana é, aproximadamente, o máximo real autovalor de uma matriz derivada dos parâmetros da fonte, recursos do MUX e CLP, ou seja, sendo o tráfego caracterizado por L estados, Q a matriz de geração infinitesimal e o vetor chegadas $\overrightarrow{\lambda} = (\lambda_1, \lambda_2, ..., \lambda_L)$, o valor de C é dado pelo máximo autovalor real da matriz: $\Lambda - \frac{1}{\xi}Q$ onde $\Lambda = diag(\overrightarrow{\lambda})$ e $\xi = \frac{\ln(\epsilon)}{K}$, sendo ϵ o limite para a probabilidade de "**overflow**" nos "buffers" e K o número de circuitos virtuais considerados na operação

Em Kesidis e outros [70], é introduzida a **medida de ganho estatístico** como parâmetro para admissão de chamada, em conjunto com a capacidade equivalente para um tráfego heterogêneo com diferentes requerimentos de QoS. Já Chang e outros [71] incorpora a **teoria dos grandes desvios** derivada da mecânica estatística, onde aplica conceitos de domínio de energia e entropia.

3.3.3. Aproximação de fluxo para taxa de perda de célula

Como controlar a admissão de chamadas sem monitorar a carga da rede ATM? A seguir, é apresentado duas versões clássicas para responder a esta questão, tomando por base modelos ON-OFF ou IFP (Interrupted Fluid Processes-processo de fluido interrompido), sem considerar a distribuição do processo de chegada.

Versão Galassi [4]

Galassi considera uma fonte VBR alternando entre os estados de atividade e silêncio, como de costume, e define a probabilidade de uma fonte i estar ativa é igual à m_i/p_i e, em decorrência, a probabilidade desta mesma fonte estar em silêncio é $(1-m_i/p_i)$ sendo p_i a taxa de bit de pico e m_i a taxa média de bit.

Então, um enlace com N fontes independentes, com capacidade de C células/seg., um "buffer" de tamanho M possui como variáveis aleatórias r_i e R correspondendo à distribuição de chegada de célula da conexão i e o tráfego agregado de N conexões multiplexadas, ou seja:

$$R = \sum_{i=1}^{N} r_i \tag{3.21}$$

com isto, podemos afirmar que o fluxo contínuo de células chega à taxa R à fila do enlace e sai à taxa C deste enlace.

Com q sendo o número de "slots" no "buffer" em disponibilidade, segundo Galassi, a taxa em que as células são perdidas quando R é o valor $X > C$ é:

$$L = \sum_{X>C} (X - C).P(R = X).[1 - P(q > 0|R = X)] \tag{3.22}$$

e o "supremum" de L em relação ao comprimento de surto é:

$$supL = \sum_{X>C} (X - C)P(R = X) \tag{3.23}$$

daí extrai-se o limitante superior de CLP que denominamos $CLPmax$

$$CLPmax = supL / \sum_{i=1}^{n} m_i \tag{3.24}$$

e daí aplica-se o **teorema de limitação de Chernoff** (que, enunciado em [38] [73], originou a teoria dos grandes desvios que é hoje utilizada em análise de tráfego de redes ATM [5] [71] [74] [75] e, então, para algum "s" positivo nós temos:

$$P(R > X) < E\{e^{sR}\}/e^{sX} \tag{3.25}$$

e, se denominarmos a soma das taxas de pico das conexões de Φ, teremos:

$$\Phi = \sum_{i=1}^{N} p \qquad \text{e} \qquad \sup L < E\{e^{sR}\} \int_{C}^{\Phi} e^{-sX} dx < E\{e^{sR}\}/(se^{sC}) \tag{3.26}$$

então a função $f(s) = E\{e^{sR}\}/se^{sC}$ é um limitante superior paramétrico para CLP_{max}.

De outra forma $f(s) = E\{e^{s(R-C)}\}/s$ é uma função convexa que possui um mínimo valor $s = s*$, que é a raiz da equação a seguir para $\Phi > C$:

$$\frac{\partial \ln E\{e^{sR}\}}{\partial s} - \frac{1}{s} - C = 0 \tag{3.27}$$

e considerando que: $R = r_1 + r_2 + ... + r_N = \sum_{i=1}^{N} r_i$, e r_i representa o tráfego associado a cada conexão e $ri = 1$ com probabilidade (m_i/p_i) e $r_i = 0$ com prob. $1 - m_i/p_i$ e assim, é obtido:

$$E\{e^{sR}\} = \prod_{i=1}^{N} E\{e^{sr_i}\} = \prod_{i=1}^{N} (m_i/p_i)\{e^{sp_i} - 1\} + 1 \tag{3.28}$$

e das expressões 3.25 e 3.26 acima podemos obter $s*$ de:

$$\sum_{i=1}^{N} \frac{m_i e^{s*.p_i}}{(m_i/p_i)\{e^{s*.p_i} - 1\} + 1} - \frac{1}{s*} - C = 0$$

e o $CLPmax$:

$$CLPmax = \frac{\prod_{i=1}^{N} (m_i/p_i)\{e^{s*.p_i} - 1\} + 1}{s * e^{s*C} \sum_{i=1}^{N} m_i} \tag{3.29}$$

e é este $CLPmax$ que torna-se o parâmetro que é utilizado em decisões de admissão de chamadas de acordo com um $(CLP)_{QoS} = \epsilon$ especificado em contrato.

Se $CLPmax < \epsilon$ a chamada i será aceita, caso contrário será rejeitada.

Observemos que o CLP de cada conexão $(\text{CLP})_i$ é obtido pela fórmula delimitadora:

$$(CLP_{\max})_i = supL\frac{p_i}{X} = \sum_{X>C}(X-C)P(R=X)\frac{p_i}{X} \qquad para \qquad i = 1, 2, ..., N \quad (3.30)$$

Assim, cada conexão individual pode ter seu $(CLP_{\max})_i$ calculado para uma carga X corrente no "buffer" e comparado com requisitos QoS da conexão individual.

Versão Saito [76]

H. Saito, preocupado com aspectos de dimensionamento de redes ATM, observou que os métodos existentes na época não relacionavam o tamanho do "buffer" com CLP, mas apenas comparavam a carga instantânea deste com um nível limite para CLP.

Buscando estas soluções propôs um esquema de CAC para garantir um padrão de CLP, considerando o tamanho do "buffer" do enlace e os efeitos de fila neste. É baseado somente nos parâmetros especificados pelo usuário e não requer um modelo do processo de chegada.

Considere um enlace de transmissão de C bps, seu "buffer" de tamanho M e o $CDV_{\max}$ neste "buffer", sob uma disciplina FIFO será: $M = (CDV_{\max})\frac{C}{L}$, sendo L o comprimento da célula ATM, em bits;

Então, **a estratégia é definir o CAC em função de requisitos de CLP após estabelecer o tamanho do "buffer" com $CDV_{\max}$ e a capacidade do enlace.**

Assumindo que um intervalo fixo de tempo ΔT (pode ser generalizado para um ciclo de r "slots" de "buffer", como tempo de observação) é aquele em que $M/2$ (metade do "buffer") células são transmitidas ou seja $\Delta T = (CDV_{\max})/2$, o conjunto de parâmetros de tráfego especificados pelo usuário é formado por:

- número médio de células que chegam em ΔT: ANA (Average Number of Cells Arriving-Average NA);

- número máximo de células (número inteiro) que podem ocorrer em ΔT: MNA (Maximum of NA);

- variância do número de células que chegam em ΔT: VNA (Variance of NA);

- $(CLP)_{QoS}$ é a taxa de perda de célula ou o principal requisito de Qualidade de Serviço (QoS) a ser observado, geralmente especificado em contrato entre o provedor de serviços de rede ATM e o usuário.

De acordo com os parâmetros fornecidos pelo usuário, pode-se extrair:

Limite superior de CLP derivado de MNA e ANA:

Assumindo que N chamadas ou conexões são transmitidas por um enlace e a probabilidade que j células da i-ésima conexão chegue dentro de ΔT é:

$$p_i^*(j), i = 1, 2, ..., N; j = 0, 1, 2, ... \tag{3.31}$$

então, no "buffer", o CLP é superiormente limitado por:

$$B(p_1^*, ..., p_N^*; M/2)$$

e definido como abaixo (a demonstração encontra-se no apêndice de [76]):

$$CLP \leq B(p_1^*, ..., p_N^*; M/2) \triangleq \frac{\sum_{k=0}^{\infty} [k - M/2]^+ p_1^* * ... * p_N^*(k)}{\sum_{k=0}^{\infty} k p_1^* * ... * p_N^*(k)} \tag{3.32}$$

onde:

$*$ é a operação de convolução;

$[x]^+ = x$ se $x \geq 0$ e $[x]^+ = 0$ se $x \leq 0$;

k é o k-ésimo intervalo de célula ao chegar ao limite de $M/2$;

Adota-se, a seguir, para simplificar: $MNA = R_i$ e $ANA = a_i$ para a i-ésima chamada que agora será especificada pelos parâmetros (a_i, R_i) e $\theta_i = \{\theta_i(j)\}$ a distribuição Bernoulli desta chamada, ou seja:

$$\theta_i(j) = \left\{ \begin{array}{c} (a_i/R_i) \text{ para } j = R_i; \\ (1 - a_i/R_i) \text{ para } j = 0; \\ 0 \text{ para demais casos;} \end{array} \right\} \tag{3.33}$$

ou também, que θ_i é a distribuição que representa o máximo número de células chegando ou nenhuma célula chegando e se seu número médio de células chegando é a_i e Saito, em [77] (no apêndice B) deduz que:

$$CLP \leq B(\theta_1, ..., \theta_N; M/2) = \frac{\sum_{k=0}^{\infty} [k - M/2]^+ \theta_1, ..., \theta_N(k)}{\sum_{k=0}^{\infty} k\theta_1, ..., \theta_N(k)} \tag{3.34}$$

Assim, se uma chamada $N+1$ solicitar admissão calcula-se

$$B(\theta_1, ..., \theta_N, \theta_{N+1}; M/2)$$

se, dado um $(CLP)_{QoS}$ ocorre

$$B(\theta_1, ..., \theta_N, \theta_{N+1}; M/2) \leq (CLP)_{QoS}$$

a nova conexão será aceita, caso contrário, rejeitada.

Limite Superior de CLP derivado de ANA e VNA

Agora o usuário especifica somente a média (ANA) e a variância (VNA) das chamadas que chegam no intervalo $M/2$ e, para simplificar chamamos $ANA = a_i$ e $VNA = \sigma_i^2$, e o conjunto $p = p_1^* * ... * p_N^*$ em que Saito, [77] (no apêndice C) obtém a limitação superior:

$$CLP \leq B\left(p_1^*, ..., p_N^*; M/2\right) \leq B(q_N; M/2) \triangleq \frac{(i^* - M/2)\left\{l\left(l-1\right) - a\left(2l-1\right) + \sigma^2 + a\right.}{-a\left(l - i^*\right)\left(i^* - l + 1\right)} \tag{3.35}$$

onde:

$$a = \sum_{i=1}^{N} a_i, \sigma = \sum_{i=1}^{N} \sigma_i, l = \lceil a \rceil, \tag{3.36}$$

e i^* (i^* é definido no apêndice C de [77] citado, pelas relações C.11 e C.12).

Em resumo, quando a chamada N+1 solicita admissão calcula-se $a = \sum_{i=1}^{N+1} a_i$, $\sigma = \sum_{i=1}^{N+1} \sigma_i$, $l = \lceil a \rceil$ **e** i^* **para esta chamada e ela será aceita somente se** $B(q_{N+1}; M/2) \leq (CLP)_{QoS}$.

3.3.4. CAC em função do CDV

Este método foi proposto por Skliros em [66], capítulo 5, e utiliza a função **UPC à nível de circuito virtual**, com o algoritmo GCRA em função da taxa da pico de célula PCR e o "jitter" (τ ou CDVT), de cada conexão, gerado pelas respectivas fontes. Ele se baseia no fato de que, se a soma dos tamanhos máximos de surtos das conexões individuais sob pior caso de tráfego (emissão de PCR e ocorrência de CDVT) não ultrapassar o tamanho M do "buffer", todas podem ser admitidas para tráfego. Apesar deste método subutilizar estatisticamente a rede ATM ele pode fornecer segurança, conforme for prioritário, para as redes que lidam com tráfego heterogêneo nos seus "buffers" de saída.

Conforme Figura 3.5, adotando-se a distribuição de cada fonte como genericamente geométrica-GGeo caracterizada por dois momentos [21], o inverso da taxa de pico de célula $\frac{1}{PCR}$ e a variância σ^2 (ver, na Figura 3.4).
sendo que:

$$\sigma_i^2 = (MBS_i - 1)\left[1 - (PCR_i/C)\right]^2 \tag{3.37}$$

e o MBS_i, o tamanho máximo de surto da conexão i, sob as piores condições de $CDVT_i$ e PCR_i destas conexões e é dado pela equação 1.10:

$$MBS_i = 1 + \lfloor CDVT_i/[(1/PCR_i) + (1/C)] \rfloor \tag{3.38}$$

procede-se então a soma dos tamanhos máximos de surtos, resultando no tamanho

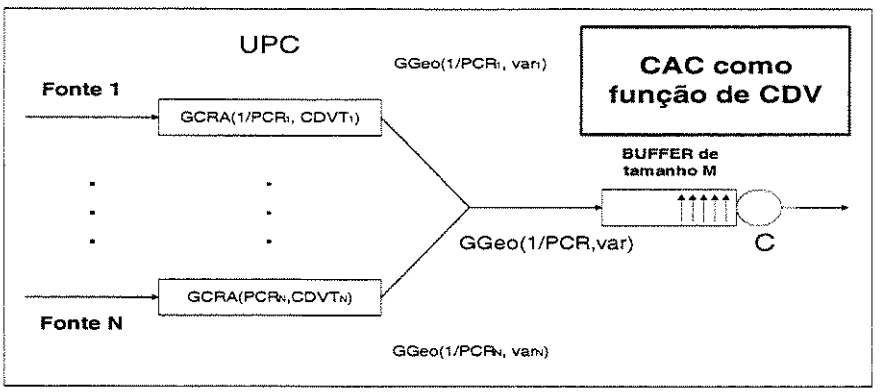

Figura 3.5: CAC como função do CDV (CDVT)

máximo do surto no "buffer" de tamanho M:

$$MBS = \sum_{i=1}^{N} MBS_i \tag{3.39}$$

Se uma conexão (N+1) solicita admissão, calcula-se $MBS(N+1) = \sum_{i=1}^{N+1} MBS_i$, **se** $MBS(N+1) < M$, **a conexão é aceita, caso contrário, é rejeitada!**

3.3.5. Aproximação para tráfego "pesado";

K. Sohrabi propôs este método [4] estudando o comportamento assintótico da "calda "da distribuição do comprimento de uma fila em um "buffer" teoricamente de capacidade infinita. Aperfeiçoou este estudo em [78], com uma abordagem de **espaço de estados**, utilizando metodologia da matriz geométrica (cuja teoria está em [26] e [79], com aplicações para MMPP e MMBP em [35]), .

O estudo, baseado na condição de estado estacionário da "calda" da distribuição da fila, considera os tempos de serviço constantes (tempo de célula ATM) e o processo de chegada descrito por uma matriz de probabilidade $P(z)$ (veja tráfego MMPP e MMBP em modelos de tráfego, neste trabalho) que possui a menor raiz z^* delimitada pelo círculo de raio unitário, do determinante desta matriz.

O método gerencia diretamente o "buffer" e as decisões de admissão (CAC) são baseadas na questão do comprimento da fila q ou na probabilidade de $q \geq i$ ou $P(q \geq x_{(CLP)_{QoS}}) = p$, sendo $x_{(CLP)_{QoS}} = x(\epsilon)$, o limite de comprimento de fila a partir do qual o $CLP \geq (CLP)_{QoS} = \epsilon$ lembrando que a rede ATM deve ter como condição $P(q > x(\epsilon)) \leq \epsilon$ para manter os requisitos de QoS.

Os autores concluem que, para x tendendo a infinito (ou muito grande):

$$P\{q > x(\epsilon)\} = \alpha(1/z^*) \tag{3.40}$$

onde α é uma constante desconhecida, a ser estimada. A questão então é calcular z^* e estimar α. Seja a fonte ON/OFF considerada com parâmetros:

R_i a taxa de pico de célula da conexão i;
m_i a taxa média de célula da conexão i;
b_i o comprimento médio de surto da conexão i;
ρ_i o fator de utilização da fonte i, ou seja: $\rho_i = \frac{m_i}{R_i}$;

A constante α é estimada como um fator de utilização da fila com N conexões multiplexadas, ou seja:

$$\alpha = \sum_{i=1}^{N} \rho_i = \sum_{i=1}^{N} \frac{m_i}{R_i} \tag{3.41}$$

e os autores, assumindo que esses períodos ON/OFF são exponencialmente distribuídos, calculam z^*

$$z^* = 1 + \frac{1-\alpha}{\sum_{I=1}^{N} m_i \left(1-\rho_i\right)^2 b_i} \tag{3.42}$$

e para o caso de distribuições arbitrárias de períodos ON/OFF, Sohrabi utiliza a expressão:

$$z^* = 1 + \frac{1-\alpha}{\sum_{I=1}^{N} m_i \left(1-\rho_i\right)^2 b_i [c_i^2(ON) + c_i^2(OFF)]} \tag{3.43}$$

onde $c_i^2(ON)$ e $c_i^2(OFF)$ são coeficientes quadráticos das variações dos períodos ON e OFF, respectivamente. Assim o comportamento da "calda" da distribuição pode ser descrito aproximadamente por:

$$P\{q > x(\epsilon)\} = \gamma \left(1/z^*\right)^{x(\epsilon)} \tag{3.44}$$

onde γ é a intensidade de tráfego (o autor sugere $0,8 \leq \gamma \leq 1$ como fator de segurança, generalizando a utilização como "tráfego pesado") e aproxima o comprimento de fila crítico como sendo o tamanho do "buffer" a ser utilizado em dimensionamento, assim $x_{(CLP)_{QoS}} = M$ ou:

$$P\{q > M\} = \gamma \left(1/z^*\right)^M \tag{3.45}$$

Concluindo, a conexão N+1 que requer admissão, será aceita se $z^*(N+1)$, calculado nas situações apropriadas de tráfego, resultar em:

$$\ln[\gamma \left(1/z^*\right)^M] \leq \ln \epsilon \tag{3.46}$$

Se isto não ocorrer, a chamada será rejeitada.

3.3.6. Alocação com tráfego fractal na entrada

Como já mencionamos em Modelos de Tráfego, os processos auto-similares tem se destacado como caracterização realística do comportamento estatístico de tráfego em

redes ATM e o esquema de CAC que se segue, em duas versões, alocação estatística e determinística de "buffer", foi desenvolvido por Giordano e outros [61] e incorpora estas características ditas "fractais" notadas principalmente no tráfego multimídia de dados [28].

O esquema explora o fato das redes ATM serem orientadas à conexão ou seja, estabelecidos caminho e circuito virtuais, estes configuram a ligação até o final da chamada, por isto é dado ênfase na obtenção do ganho estatístico, assim a admissão de chamada e a formação desta ligação seria baseada em julgamento mais realístico com um conjunto de descritores de tráfego e parâmetros QoS obtidos à partir do comportamento auto-similar do tráfego heterogêneo.

É esta a primeira questão a se resolver: Como definir os padrões QoS de acordo com as características auto-similares destas fontes ATM, tais como CLP e CTD ?

Compondo o CTD (vide Figura1.16), está o CDV, que por sua vez tem como um dos fatores principais o atraso de fila, geralmente nos "buffers" de saída dos comutadores ATM (configuração típica). Este fator é a chave para entendermos o comportamento de tais parâmetros sob diversos tipos de tráfego sendo que o assunto foi pesquisado em [80], [81] e [82], mas Tsybakov em [44] abordou plenamente a questão, enfocando os efeitos do tráfego fractal nestas filas.

Voltando à seção redes ATM neste trabalho, a Figura 1.11 mostrou a estrutura interna de um comutador ATM com "bufferização" na saída. Após a comutação, as células são armazenadas nestes "buffers" que, no esquema ora apresentado são de tamanho limitado, tráfego de entrada de modelo estocástico auto-similar ("ruído fracionário Gaussiano"-fGn) e, na saída, servidores com serviço determinístico.

Para alocar largura de banda (BW), dado um certo CDV e CLP, em [61] define-se duas possibilidades de tratamento para o tráfego auto-similar na entrada: "buffers" virtuais determinísticos ou "buffer" estatístico (vide Figura. 3.6):

O primeiro (Figura 3.4a) aloca, para cada conexão estabelecida e seu correspondente caminho virtual (VP), um "buffer" ou um tamanho fixo de fila que representa uma parte (de posição fixa) deste "buffer", sendo tudo identificado logicamente. Assim, com N VP's passa-se a ter N "buffers" virtuais e seus correspondentes N servidores virtuais permitindo que dado N fontes constituindo um tráfego heterogêneo, possa-se identificar N requisitos QoS diferentes dedicados a cada VP facilitando o controle de tráfego no nó ATM. Mas, como anteriormente vimos, a grande desvantagem do tratamento determinístico é a sub-utilização dos recursos da rede ATM apesar das facilidades de controle.

Para remediar, pode-se adotar o comportamento estatístico (Figura 3.4b), que considera o "buffer" de saída como de fila única contendo tráfego superposto proveniente da multiplexação das N fontes ATM. Os requisitos QoS seriam somente da composição deste tráfego heterogêneo que é denominado "sistema de fila fractal" quando as N fontes forem independentes e apresentarem certo grau de auto-similaridade ($H > 0,5$ - vide parâmetro H, de Hurst, em Modelos de Tráfego, neste trabalho).

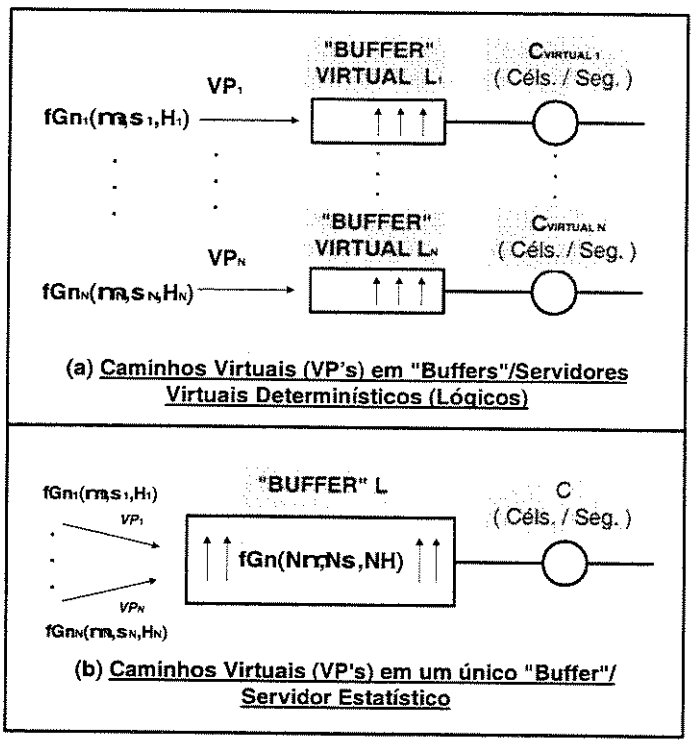

Figura 3.6: Estratégias de Tratamento de Tráfego Auto-Similar

Assumindo a generalização para tráfego auto-similar, I. Norros em [55] e [47] propõe a abordagem de **aproximação por difusão sob tráfego pesado** para se calcular o número de chegadas $A(t)$ no intervalo $(0,t]$ por processo de Poisson:

$$A(t) \approx \mu t + \sqrt{\mu} W(t) \tag{3.47}$$

onde; μ é a taxa média de entrada, $W(t)$ é o processo de movimento Browniano padrão $(H = 0,5)$. E para ruído Gaussiano fracionário-fGn, Norros decorre que:

$$A(t) \approx \mu t + \sqrt{a\mu} Z(t) \tag{3.48}$$

onde: $Z(t)$ é o processo Browniano fracionário com parâmetros $(0, |t|^{2H})$ sendo que $H \in [.5, 1]$, a é um coeficiente de variância $(a > 0)$

Admitindo as seguintes propriedades:

a) o incremento $X(n)$ do processo $A(t)$ no intervalo $((n-1)T, nT]$ é um fGn com:

$$X(n) = A(nt) - A((n-1)T) \tag{3.49}$$

$$E[X(nT)] = \mu T \tag{3.50}$$

$$var[X(nT)] = a\mu |t|^{2H} \tag{3.51}$$

b) Para $T = 1$ temos que:

$$var[X(nT)] = a\mu \tag{3.52}$$

c) Associando $V(t)$ como sendo o processo virtual de tempo de espera e A_s como o número de chegadas em estado estacionário, no que Norros chama de "armazenagem" Browniana fracionária (Ruído-fBn) e utilizando a fórmula de Reich [55], obtém que:

$$V(t) = \sup_{s \leqslant t} [A(t) - A_s - C(t-s)] \tag{3.53}$$

onde C é a capacidade do enlace

$V(T)$ representa o número de células na fila assumindo o "buffer" de tamanho infinito, obtém-se para um tamanho de fila crítico $x(\epsilon)$ (podendo até ser o tamanho do "buffer" M)

$$\epsilon = P\{V(t) > x(\epsilon)\} \tag{3.54}$$

sendo ϵ o limitante superior da probabilidade de perda de célula para uma fila de tamanho $x(\epsilon)$;

d) Da estacionaridade do Processo decorre que:

$$\begin{aligned} P\{V(t) > x(\epsilon)\} &= P\{V(0) > x(\epsilon)\} = \\ P\left\{\sup_{s \leqslant 0}[-A(s) + Cs] > x(\epsilon)\right\} &\geqslant \max_{t \geqslant 0}\{P[A(t) > Ct + x(\epsilon)\} \end{aligned} \tag{3.55}$$

e aplicando-se a aproximação de Weibull [48] para a distribuição complementar $Q(.)$, **assumida Gaussiana**, [55] chega ao limitante superior de $Q(x)$

$$\epsilon = e^{-\frac{(C-\mu)^{2H}}{2a\mu[(1-H)^{1-H}H^H]^2}x(\epsilon)^{2-2H}} \tag{3.56}$$

e, como no citado caso, trata-se de tráfego composto:

$$A'(t) = \sum_{i=1}^{N} A_i(t) \qquad \mu = \sum_{i=1}^{N} \mu_i \qquad a = \frac{\sum_{i=1}^{N} a_i \mu_i}{\sum_{i=1}^{N} \mu_i} \tag{3.57}$$

Quando os parâmetros H de cada conexão forem diferentes, ou seja, processos fGn independentes, o processo A(t) será:

$$A(t) = \sum_{i=1}^{N} A_i(t) \tag{3.58}$$

ou seja, um processo Gaussiano de incrementos estacionários.

Assim obtém-se que [61]:

$$P\{V(t) > x(\epsilon)\} \geqslant \max\left\{Q\left[\frac{x(\epsilon) + (C-\mu)t}{\sigma}\right]\right\} \tag{3.59}$$

onde:

$$\sigma^2 = \sum_{i=1}^{N} a_i \mu_i t^{2H_i} \qquad e \qquad \mu = \sum_{i=1}^{N} \mu_i \tag{3.60}$$

são respectivamente, a variância e a média do processo superposto.

Para maximizar Q(.), iguala-se a primeira. derivada a zero obtendo:

$$(C-\mu)\sum_{i=1}^{N} a_i \mu_i (1-H_i)^{2H_i} - x(\epsilon)\sum_{i=1}^{N} a_i \mu_i t^{2H_i - 1} = 0 \tag{3.61}$$

A a partir daqui, pode-se descrever as duas propostas de *CAC de tráfego fractal.*

Algoritmo de CAC fractal de comportamento de entrada determinístico

Conforme já descrito, na linha determinística particiona-se os "buffers" criando "buffers" virtuais lógicos e que, para cada fonte assume-se que a rede provenha C_i e K_i que garanta os requisitos QoS. Assim, de acordo com a Figura 3.7:
- Os descritores de tráfego μ_i, σ_i^2 e H_i são especificados para cada fonte i até N;
- Para uma conexão $(N+1)$ que solicita admissão especifica-se μ_{N+1}, σ_{N+1}^2 e H_{N+1}
- Entra-se com o valor de CLP_i requerido para garantir QoS para cada fonte i;
- Soma-se as taxas médias de células $\mu_{tot} = \sum_{i=1}^{N+1} \mu_i$, então, se;

μ_{tot} ultrapassa C, as conexões de menor prioridade são descartadas ou rejeita-se a chamada $(N+1)$, caso contrário, verifica-se C_i tal que;

- A rede ATM atribui um VP_i para cada fonte i com o correspondente C_i, incluindo a chamada $(N+1)$;

- Igualando $\epsilon_i = CLP_i$ calcula-se K_i ou o tamanho da partição do "buffer" para cada VP_i,

- Se a soma das partições $\sum_{i=1}^{N+1} K_i > K$, **as conexões de menor prioridade são descartadas ou rejeita-se a chamada N+1**, caso contrário;

- Avalia-se o menor $K_{\min}$ para o $VP_{\min}$ o maior $K_{\max}$ para o maior $VP_{\max}$, se $K_{\min} > K$, **as conexões de menor prioridade são descartadas ou rejeita-se a chamada N+1**;

- Caso contrário procede-se a realocação escalonada de 1 célula de "buffer" por segundo de banda, através do acréscimo de 1 célula de "buffer" /seg ao c_{max} e o decréscimo de 1 célula de "buffer" /seg ao c_{min} e procede-se novamente ao cálculo de

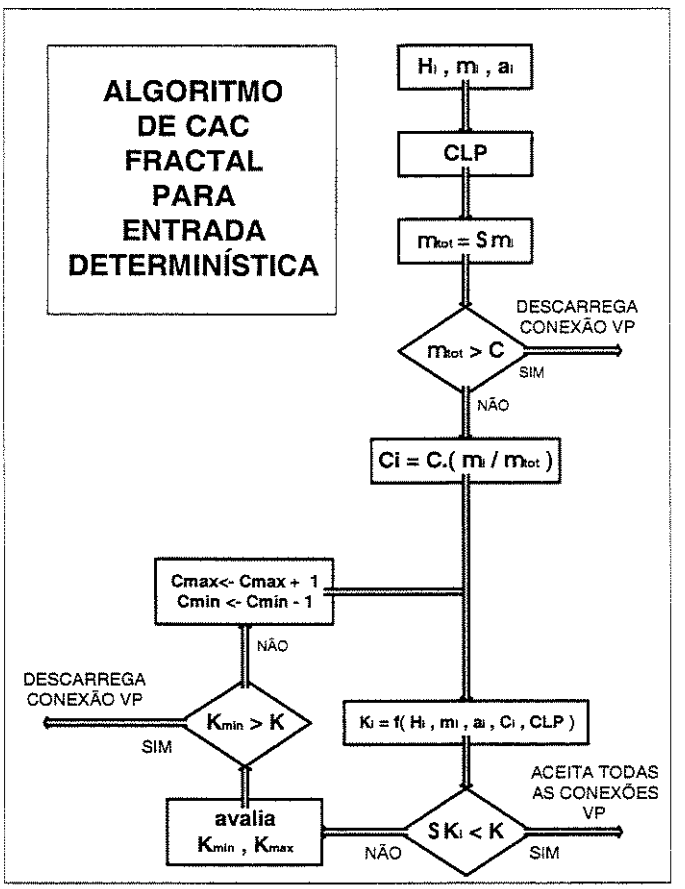

Figura 3.7: Algoritmo de CAC para Tráfego Fractal com entrada determinística

K_i e à verificação $\sum_{i=1}^{N+1} K_i < M$, sendo que para otimização de recursos na rede ATM, assume-se $K = M$;

- Após alguns ciclos de realocação, quando ocorrer $\sum_{i=1}^{N+1} K_i < M$ **aceita-se a chamada N+1 com as N conexões já em tráfego.**

Algoritmo de CAC fractal de comportamento de entrada estatístico:

De acordo com a Figura 3.8, quando a conexão $(N+1)$ solicita admissão:

- Determina-se os parâmetros de tráfego H_{N+1}, μ_{N+1}, a_{N+1} e, juntamente com as conexões que já estavam calcula-se $CLP(N+1)$:

$$\sigma^2 = \sum_{i=1}^{N+1} a_i \mu_i t^{2H_i} \ , \ \mu = \sum_{i=1}^{N+1} \mu_i \ \text{e} \max \left\{ Q \left[\frac{M + (C-\mu)t^*}{\sigma} \right] \right\} = CLP(N+1) \qquad (3.62)$$

onde M é o tamanho do "buffer" e t* é o valor ótimo de t que maximiza Q(.) de 3.57

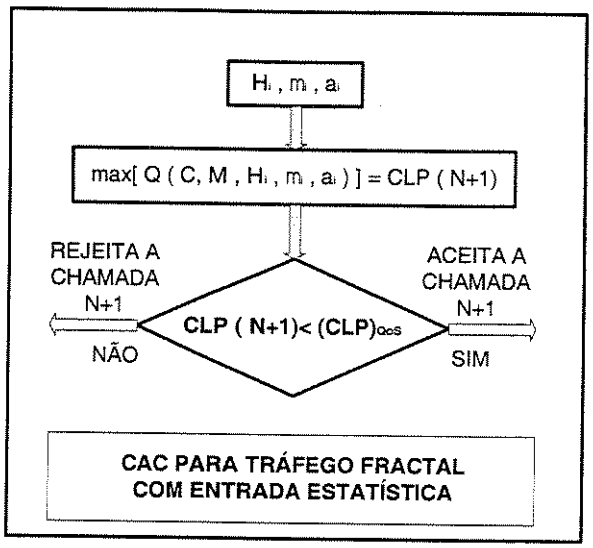

Figura 3.8: CAC fractal com entrada estatística

que, para N+1, fica:

$$(C-\mu)\sum_{i=1}^{N+1} a_i\mu_i(1-H_i)(t^*)^{2H_i} - M\sum_{i=1}^{N+1} a_i\mu_i(t^*)^{2H_i-1} = 0 \tag{3.63}$$

Obs.: Considerando N conexões iguais, de H_k, e $Q(x) = CLP(N+1) \rightarrow x = Q^{-1}(M, C, \mu, t, a) = \frac{M+(C-\mu)t^*}{\sigma}$, obtem-se o valor ótimo $t^* = \frac{M}{(C-Nm)}\frac{H_k}{1-H_k}$ da equação acima, e que:

$$x = \left(\frac{M}{1-H_k}\right)\frac{1}{\sqrt{Nma}}\left[\frac{C_2' - Nm}{M}\right]^{H_k}\left[\frac{(1-H_k)}{H_k}\right]^{H_k} \tag{3.64}$$

onde C_2' será o tráfego compartilhado com outro algoritmo $(C_1 + C_2 = 1)$ durante a simulação numérica com o Matlab no proximo capítulo.
Normalizando-se m, C_2' e $\sqrt{a} = \sqrt{\sigma^2} = \sigma_k$ em C, capacidade total do enlace, obtem-se respectivamente m_k, C_2, e σ_k para a seguinte expressão utilizada para programação (já com a numenclatura utilizada na programação):

$$X = \left[\frac{(1-H_k)^{(H_k-1)}}{H_k^{H_k}}\right]\left[\frac{(C_2 - Nc_3m_k)^{H_k}}{\sqrt{Nc_3m_k}}\right]\left[\frac{M^{(1-H_k)}C^{(H-3/2)}}{\sigma_k}\right] \tag{3.65}$$

onde Nc_3 é o número de conexões admitidas pelo algoritmo 3 que será comparado com outros dois primeiros algoritmos no capítulo 4.
então, **compara-se o valor CLP(N + 1) com $(CLP)_{QoS}$, requerido para manter a QoS do enlace de capacidade C: se for maior rejeita-se a conexão N+1, caso contrário, aceita-se.**

3.4. Sistemas CAC neuro-fuzzy;

3.4.1. Evolução tecnológica

Quando Saito [76] propôs seu método de CAC já mencionado anteriormente, Hiramatsu em [83] propôs (na mesma publicação IEEE JSAC de sept/91) o primeiro método de CAC envolvendo redes neurais ("back propagation") dando início a uma série de propostas subseqüentes tais como Habib [84], Faragó [85], LeMair [87], Cheng & Chang [90], Tsitsiklis [91], Chang & Hu [92] e Fan & Mars [93], sendo que nestes dois últimos empregou-se também técnicas de predição de tráfego.

Ao mesmo tempo surgiram esquemas envolvendo técnicas de lógica Fuzzy em Ramamurthy [94], Youssef & Habib [95], Uehara [96]; e mais recentemente, em Leung & Kan [98] iniciou-se a implementação de treinamento de redes neurais com o filtro de Kalman estendido, agilizando o aprendizado destas, e assim prossegue este extenso campo de pesquisa pois o completo e efetivo CAC ainda está por aparecer.

3.4.2. Algoritmo de CAC com redes neuro-fuzzy

Sem entrar em detalhes sobre o funcionamento intrínseco destas redes descreve-se a seguir o método proposto por Cheng & Chang [49] pois este mostra-se bem didático para o presente trabalho, conforme constataremos, e por isto exploramos este método sugerindo propostas de aprimoramento.

O esquema pode ser visto na Figura 3.9:

O bloco estimador de BW é uma rede fuzzy que recebe informações sobre a fonte que requer admissão, ou seja, taxa de bit de pico (Rp), taxa média de bit (Rm) e duração da taxa de bit de pico (Tp);

Juntamente com a informação inerente da BW total da rede ATM estima, como resultado, a fração desta BW total que a chamada $(N+1)$ requer, ou seja:

$$C_e = \frac{BW(N+1)}{BW_{total}} \tag{3.66}$$

Este resultado segue para o gerenciador de recursos da rede, que o compara com a fração normalizada C_a que resta de BW se retiradas as frações normalizadas das conexões em tráfego C_i, ou seja:

$$C_a = 1 - \sum_{i=0}^{N} C_i \qquad \text{onde } C_i = \frac{BW_i}{BW_{total}} \tag{3.67}$$

se

$$C_e \geq C_a \ ,$$

a conexão é impossível e esta informação segue para o controlador neural, que está

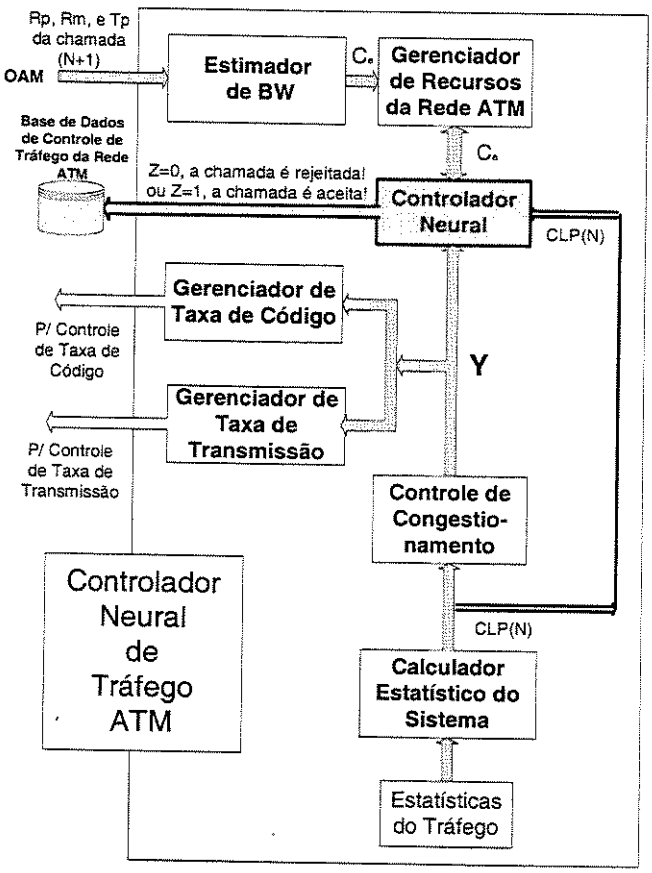

Figura 3.9: CAC com Redes Neuro-Fuzzy

previamente treinado para negar admissão nestas situações, caso contrário,se

$$C_e \leq C_a$$

a conexão é possível, o gerenciador de recursos libera C_a (apesar da chamada $N+1$ ocupar apenas $C_e\,(N+1)$ na rede ATM) e o controlador neural processa esta informação junto com as demais de sua entrada ou seja, o índice de congestionamento Y e o $CLP(N)$.

O índice de congestionamento Y é fornecido pelo calculador estatístico, com base nas informações estatísticas de tamanho da fila q_i, variação do tamanho da fila Δq_i e $(CLP)_i$ de todas as chamadas em tráfego até a chamada N. Este parâmetro pode ser definido, no caso de N fontes, como análogo ao ganho estatístico:

$$Y = G = \frac{N}{N_{\max}} \tag{3.68}$$

onde $N_{\max}$ é o número máximo de fontes a ocupar o "buffer" para que o $(CLP)_{QoS}$ do enlace seja respeitado e N o número de fontes em tráfego. Para uma fila multiplexada

$ND/D/1$, o fator de ocupação ρ do "buffer" com q células é dado por [34] como;

$$\rho = \frac{2qN}{2qN - [2q^2 + N\ln(CLP)]} \tag{3.69}$$

Dividindo por $(2qN)$ obtemos: $\frac{\rho}{2qN} = \frac{1}{1-[(q/N)+\frac{\ln(CLP)}{2q}]}$ e, como $2qN >> 1$, para que $\frac{\rho}{2qN} < 1$, é necessário que $[(q/N) + \frac{\ln(CLP)}{2q}] < 0$ ou, usando a definição 3.61:

$$\left(\frac{q}{YN_{\max}}\right) + \frac{\ln(CLP)}{2q} < 0 \text{ ou } Y(N) < -\frac{2q^2}{N_{\max}\ln CLP(N)}$$

Considerando a capacidade M do "buffer" e $(CLP)_{QoS}$ do enlace e que $\ln(CLP)_{QoS} < 0$ então:

$$Y_{\max} = \left\lfloor -\frac{2M^2}{N_{\max}\ln(CLP)_{QoS}} \right\rfloor > 0 \tag{3.70}$$

Por ex.: Para um "buffer" de 1000 células ATM, máximo (de projeto) de 1000 fontes homogêneas ("buffer" K dedicado ao serviço), e um $(CLP)_{QoS} = 10^{-9}$, teremos $Y = \lfloor 96,51 \rfloor = 96$;

Quando o $Y(N) \geq Y_{\max}$ o controlador neural produzirá $Z = 0$ não permitindo a admissão e se $Y(N) < Y_{\max}$ o controlador neural o associará com C_a e $CLP(N)$ para avaliar a admissão. Quando $Y(N)$ se aproximar de $Y_{\max}$ (um limite de 10%, por exemplo), os gerenciadores de taxa de código e de transmissão fazem com que estas taxas diminuam um pouco, através do artifício de uma "bufferização" de entrada (vide Figura.3.10), que permite controlar as taxas das fontes em tráfego, permitindo garantir a admissão da chamada $(N+1)$ em condições de fronteira. Além disto, para cada tipo de serviço, é dedicado um "buffer" K deixando grupos homogêneos em separado, facilitando o controle.

Assim, **a chamada $(N+1)$ será admitida pelo controlador neural** $(Z = 1)$ se:

$$C_e(N+1) < C_a \rightarrow C_a, \qquad Y(N) < Y_{\max} \text{ e } CLP(N) < (CLP)_{QoS} \tag{3.71}$$

3.4.3. Rede neural controladora

Em [90], utilizou-se uma rede neural "back propagation" com L camadas, sendo a primeiro nó de entrada com os três parâmetros principais (1 camada de entrada com três neurônios) controladores, com $2 \leq$ *camadas escondidas* $\leq (L-1)$ sendo que a última camada consta de apenas um neurônio com resposta em degrau.

Na primeira camada entram o status de congestionamento $Y(N)$, probabilidade de perda de células $CLP(N)$ e a banda alocada C_a. O objetivo da rede neural é produzir um resultado de decisão $Z\,(N+1)$ com o mínimo erro conforme a Figura 3.11.

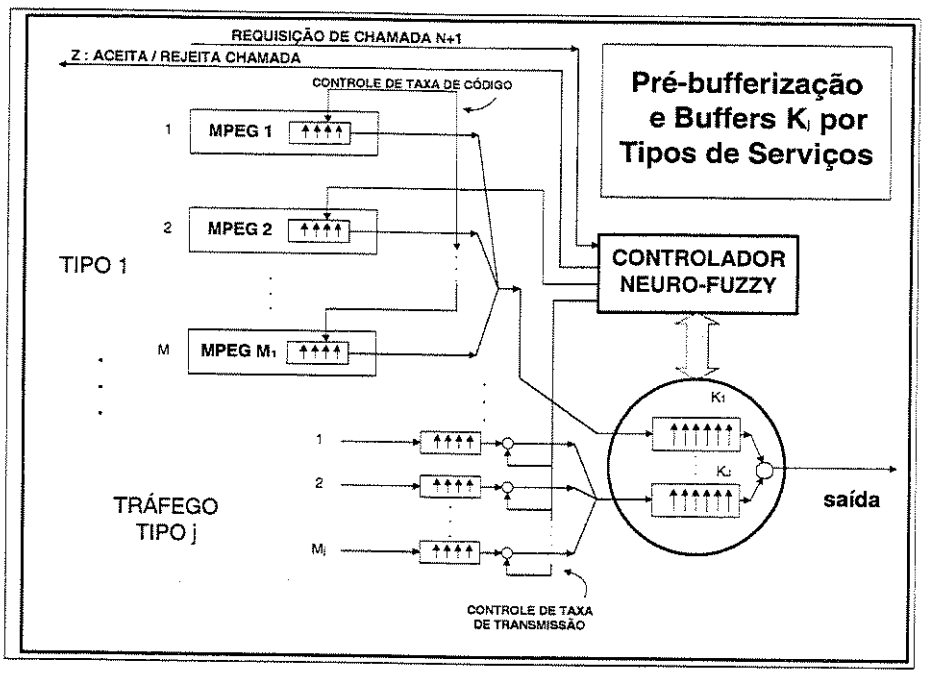

Figura 3.10: Esquema CAC Neuro-Fuzzy pré-bufferizado

3.4.4. Treinamento da rede neural

O esquema de treinamento segue o algoritmo de aprendizagem do "back propagation" que minimiza o erro de decisão com sucessivos ajustes dos pesos sinápticos de onde se produz um peso ótimo de acordo com um passo de aprendizagem conveniente para que a rede neural atue com estabilidade.

De acordo com a figura acima observamos que, a cada entrada dos parâmetros de controle, os pesos são ajustados produzindo uma saída Z, a tabela de treinamento produz entradas teóricas onde é coletado a probabilidade de perda de células, extraída sua média móvel e comparada com a especificação do serviço QoS que, com a função degrau U produz a saída teórica Z.

O sinal Z é comparado com o resultado $Z(N+1)$ e verificado o erro de decisão quando este erro tender a zero, a rede estará treinada.

3.4.5. Conclusão sobre CAC com redes neuro-fuzzy

A utilização de redes neurais em CAC's é viável, pois atende as especificações de QoS. Conforme Figura 3.9, podem ser obtidas melhorias com o artifício do pré-processamento e otimização do dimensionamento da própria rede neural (número de neurônios, pesos sinápticos, ... etc), além de dedicar cada "buffer" a tipos de serviços diferentes.

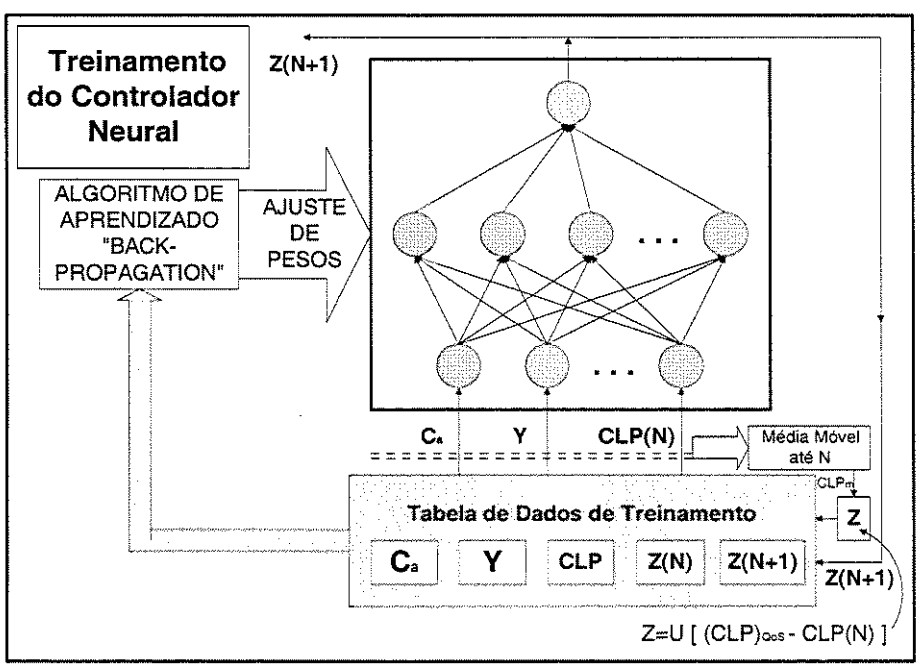

Figura 3.11: Treinamento do Controlador Neural

Capítulo 4

Simulações numéricas e resultados

4.1. Avaliação da atuação conjunta de CAC's

Como na maioria das redes ATM o tráfego é misto ou não-homogêneo, surge, a necessidade de se conhecer o comportamento dos mecanismos de admissão de chamadas destas redes sob este ambiente. Isto pode até fazer parte da fase de negociação para o estabelecimento da conexão ATM citado no capítulo anterior.

Perros & Elsayed em [60] propõe uma sistemática de avaliação, que consiste de se simular o ambiente de tráfego heterogêneo tomando as classes de tráfego duas a duas e verificando:

-as regiões de aceitação de chamada;

-influências das variações das características de tráfego das classes;

-influência do tamanho do "buffer" e

-comportamento do CLP sob variações das características de tráfego e do "buffer"

Para demonstrar esta sistemática promove-se a obtenção de resultados numéricos com a simulação em MATLAB Versão 5 de dois CAC's de **capacidade efetiva ou equivalente**, sendo um destinado ao tráfego não auto-similar ($H \approx 0.5$) e o outro, ao tráfego auto-similar ($H > 0.5$), atuando conjuntamente e compartilhando a capacidade do "buffer".

A escolha destes algoritmos se deve ao fato de que eles já foram comparados por Perros & Elsayed em [60] com os demais algoritmos mais utilizados nas redes ATM e **foi demonstrado que o algoritmo de capacidade efetiva apresenta melhor desempenho**.

4.2. Algoritmo de Capacidade Equivalente Norros-Tsybakov (tráfego com $H > 0.5$) versus Capacicade Equivalente ($H = 0.5$)

A configuração de análise é mostrada na Figura 4.1.

Os dois algoritmos foram descritos na subseção 3.3.2, capítulo 3, paginas 62-

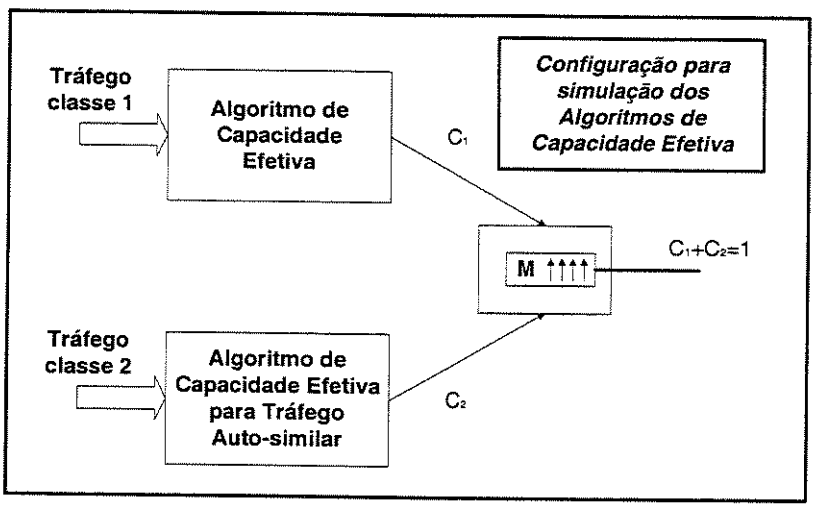

Figura 4.1: Configuração para atuação conjunta dos algoritmos de capacidade efetiva

64, deste trabalho, agora com a capacidade total do enlace de saída normalizada para $C = 1$ e conseqüentemente os demais parâmetros também normalizados em C obtem-se a equação 3.20

$$\begin{aligned} c_k &= m_k + f^{-1}(\epsilon) \times \gamma_s \\ f^{-1}(\epsilon) &= \left[H_i^{H_i} (1 - H_i)^{(1-H_i)} \sqrt[2]{-2ln\epsilon} \right] \\ \gamma_s &= v_k^{\frac{1}{H_K}} m_k^{\frac{1}{2H_k}} \left[\frac{C}{(424 \times M)} \right]^{\frac{1-H_k}{H_k}} \end{aligned} \tag{4.1}$$

O tráfego classe 1 é do tipo ON-OFF e assume os seguintes valores normalizados:

Probabilidade da fonte estar em ON:	$\rho = 0.4$ (canal de voz)
Taxa máxima de bit:	$R_i = 0.025$
Taxa média de bit	$m_i = 0.01$
Tamanho médio do surto:	$b_i = 40$

O tráfego classe 2 é do tipo auto-similar com os seguintes parâmetros:

Taxa média de bits:	$m_K = 0.01$
Coeficiente de variância:	σ^2 (vide observaçãp)
Parâmetro de Hurst:	$0.5 < H_K < 1$

Obs.: Os coeficientes de variância (nos programas MATLAB do apêndice 1 foi utilizado o desvio padrão σ normalizado em C tornando-se σ_k-sigmk na programação) foram extraídos de arquivos de tráfego gerados com H de 0.5 a 0.9, simulados com velocidade de 1.55Mbps (1% de C=155Mbps) no simulador SimATM, normalizado em

C e assim utilizado, visando com isso, futuramente aproximar os valores obtidos das avaliações com o SimATM, com a utilização destes mesmos arquivos de tráfego.

A partir daí, com os programas MATLAB fornecidos no apêndice 1 deste trabalho, obteve-se os resultados a seguir

4.2.1. Regiões de aceitação de chamadas

As regiões de aceitação de chamadas são obtidas pelos pontos de tráfego classe 1 e 2 que resultam em $C_1 + C_2 = 1$, ou seja o número de conexões em cada classe que, atuando em conjunto, resultam na ocupação total da capacidade do enlace de saída.

As Figuras 4.2 a 4.5 mostram as regiões de aceitação obtidas por simulação numérica, mantendo $CLP = 10^{-4}$, o tamanho do "buffer" $M = 100$ células constante e variando-se apenas o parâmetro H de 0.5, 0.7, 0.8 e 0.9, respectivamente, já que o valor de H para 0.6 quase não apresenta variação em relação ao H=0.5.

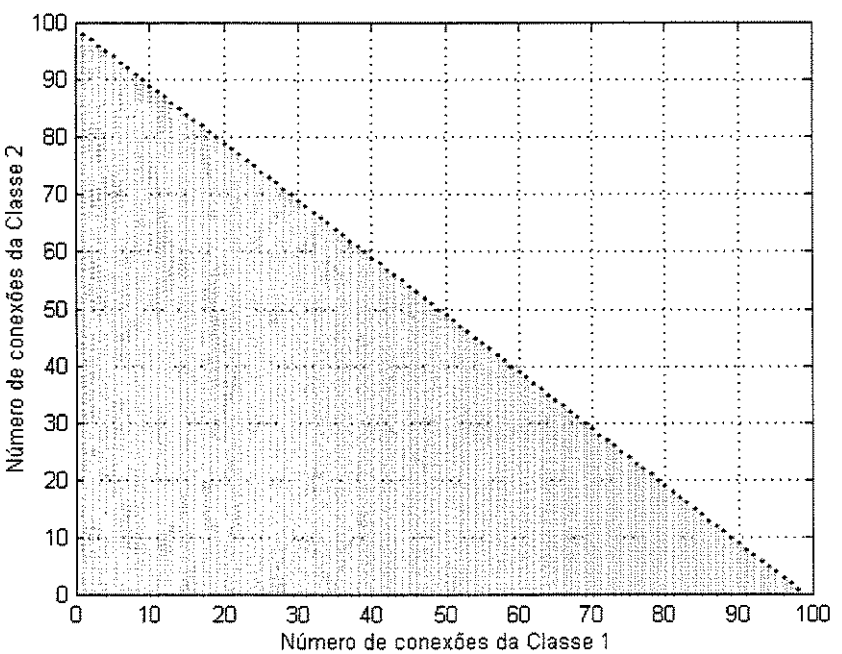

Figura 4.2: Região de aceitação do algoritmo de Norros-Tsybakov (classe2) versus algoritmo de capacidade equivalente (classe1) para H=0.5, CLP=10^{-4} e M=100 células

Na Figura 4.2 nota-se que, com $H \approx 0.5$, o tráfego classe 2 comporta-se da mesma forma que o tráfego classe1, com o mesmo número de conexões aceitas (98 conexões em ambas as classes, fornecendo um ganho estatístico de:

$$G_e = \frac{\text{número de conexões aceitas}}{\left(\frac{1}{R_i}\right)} = \frac{98}{\left(\frac{1}{0.025}\right)} = 2.45$$

Convém observar que o número máximo de conexões aceitas será $\left(\frac{1}{m_i}\right) = \left(\frac{1}{0.01}\right) =$ 100 conexões, pois este montante está limitada à média normalizada das conexões, que são iguais.

A partir da Figura 4.3, o número de conexões de tráfego classe 2 começa a ser afetado pelo aumento de H até à situação extrema de aceitação de apenas 18 conexões quando H=0.9 enquanto o comportamento para o tráfego não auto-similar (classe1) não se altera, **levando-se a concluir que se, em uma situação real estiver ocorrendo a situação máxima e acontece uma variação positiva de H, ocorrerão perda de conexões de natureza auto-similar.**

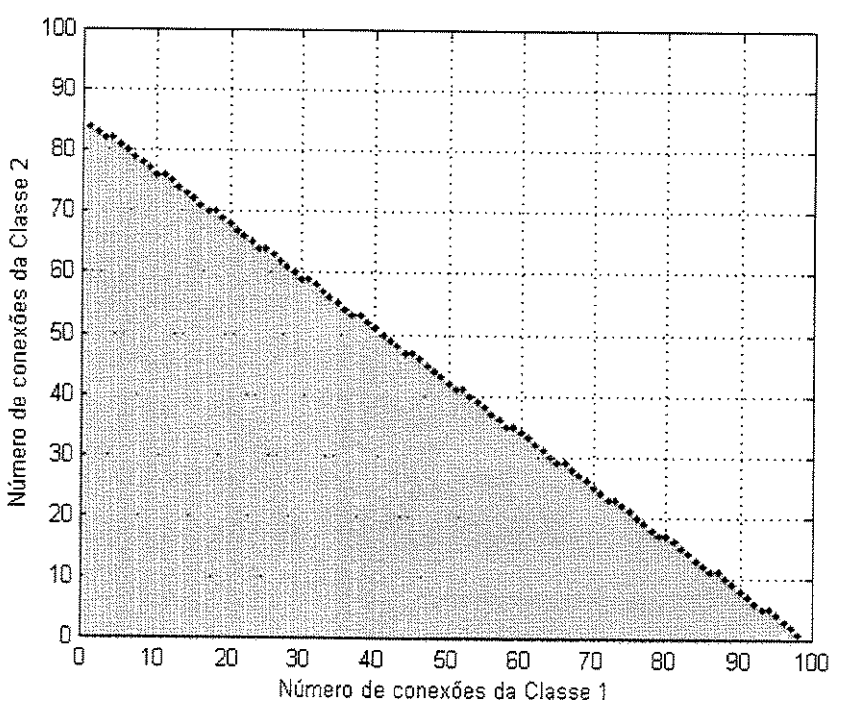

Figura 4.3: Região de aceitação do algoritmo de Norros-Tsybakov (classe2) versus algoritmo de capacidade equivalente (classe1) para H=0.7, CLP=10^{-4} e M=100 células

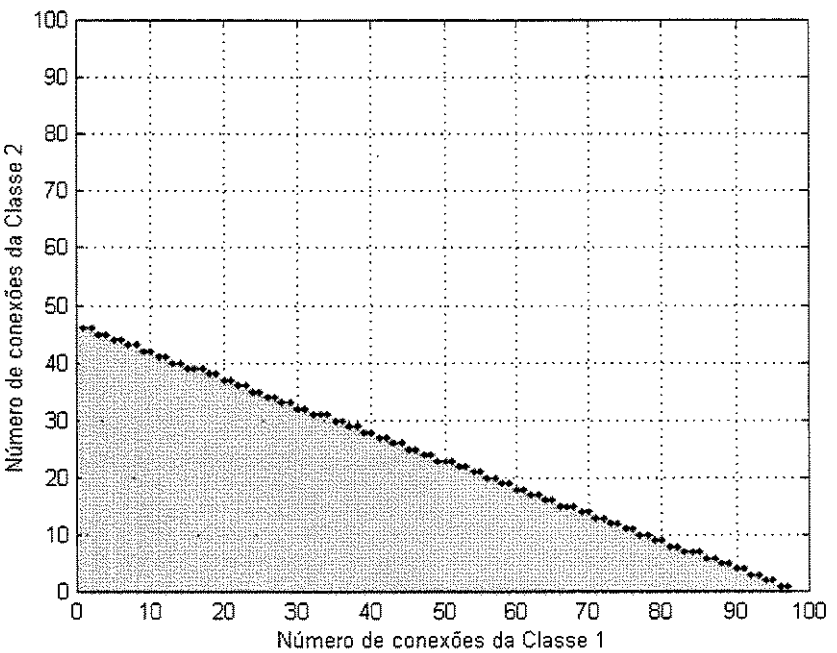

Figura 4.4: Região de aceitação para H=0,8, CLP=10^{-4} e M=100 células

4.2.2. Influência da auto-similaridade

Isolando-se a classe 1 (não fractal), pode-se observar o comportamento contínuo das conexões de natureza auto-similar (classe 2) com relação a H, na Figura 4.6. Nota-se um decaimento mais rápido a partir de $H = 0.7$ até $H \approx 1$

Pode-se observar, portanto, **que a auto-similaridade consome largura de banda, promovendo limitações na rede ATM** que fazem diferença durante sua operação.

4.2.3. Influência do tamanho do "buffer"

Nas Figuras 4.7 a 4.11, verifica-se a influência do tamanho do "buffer" na admissão de chamadas em escala semi-log para o tráfego classe 2 para $H \approx 0.5$, $H = 0.7$, $H = 0.8$ e $H = 0.9$, respectivamente. Na Figura 4.7 o mesmo número máximo de conexões (não há tráfego da classe1 competindo) permanece constante mesmo que o buffer aumente até 10^5 células. Isto porque, para este valor de H, não ocorre o efeito de latência no "buffer", verificado por Tsybakov e Georganas em [44], [46] e [56].

Este efeito começa a ser observado a partir de H=0.7, atingindo um grau crítico em $H = 0.9$.

O efeito "escada" pode ser notado e é devido ao fato do algoritmo promover a escolha de tratamento como aproximação Gaussiana ou cálculo de banda efetiva, depen-

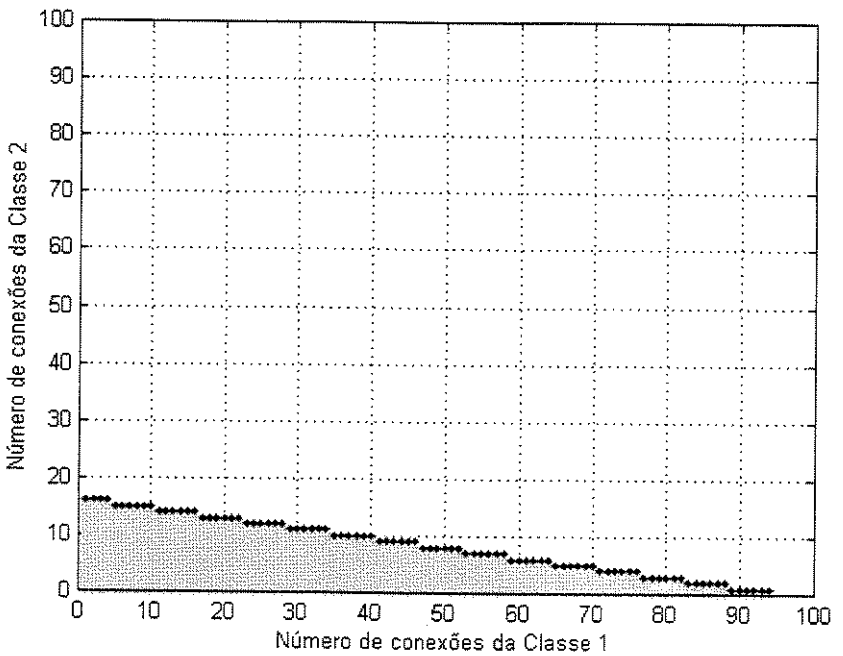

Figura 4.5: Região de aceitação para H=0,9, CLP=10^{-4} e M=100 células

dendo de qual forneça a menor capacidade efetiva.

Quando o valor do "buffer" aumenta, o valor mínimo de capacidade efetiva passa a ser fornecido por aproximação gaussiana e o algoritmo passa a não ter dependência do tamanho de "buffer", permanecendo constante o último valor calculado pelo cálculo de banda efetiva durante toda a parcela do processo de aproximação Gaussiana.

Nas regiões aparentemente mais lineares também ocorre este processo mas representado por segmentos menores, dando apens a aparência de região linear.

Observa-se, pelos resultados, **que o tamanho do "buffer", em conjunto com a característica de auto-similaridade H, devem ser considerados de extrema relevância pois são fatores decisivos para o desempenho da rede ATM.**

4.2.4. Influências da auto-similaridade e do tamanho do "buffer" no CLP

Por último, observa-se pela Figura 4.12 (a curva superior é de buffer de 10 células, a seguinte inferior é de 100 células e assim por diante) que, mesmo que se empregue tamanhos de "buffers" significativos, todas as curvas de CLP convergem para uma taxa de erro alta. Isto pode ser confirmado pela Figura 4.13, que é a ampliação da região de convergência.

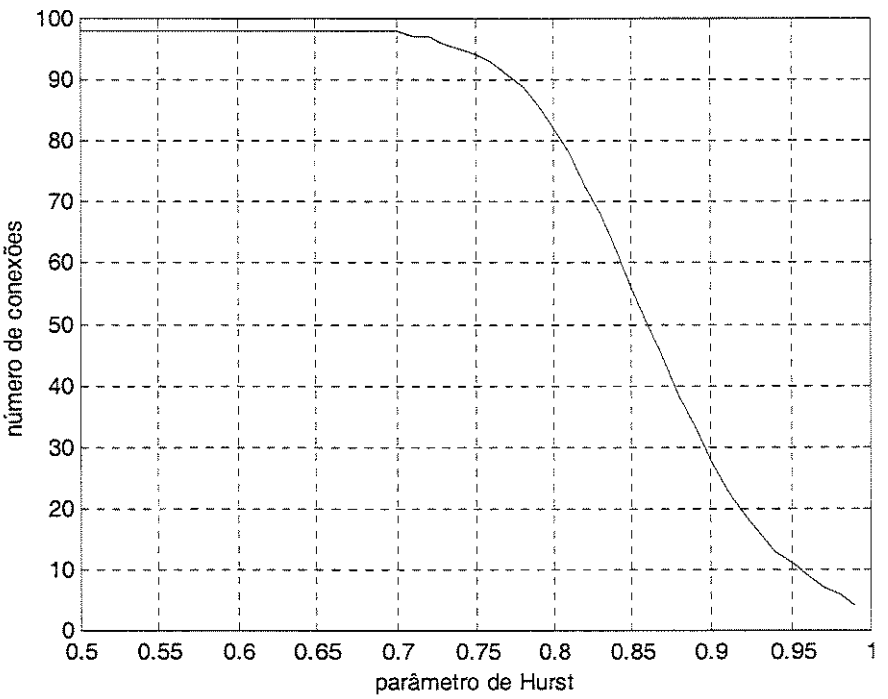

Figura 4.6: Influência do parâmetro H no número de conexões aceitas pelo algoritmo de Norros-Tsybakov

4.3. Algoritmo de Pisa versus Algoritmo de Norros-Tsybakov (ambos com tráfego de $H > 0.5$)

O algoritmo de Pisa foi apresentado nas páginas 79-80 deste trabalho e leva este nome devido a ele ter sido concebido por professores da Universidade de Pisa (ver Giordano e outros [61]).

Tomando a equação 3.65:

$$X = \left[\frac{(1-H_k)^{(H_k-1)}}{H_k^{H_k}}\right]\left[\frac{(C_2 - Nc_3m_k)^{H_k}}{\sqrt{Nc_3m_k}}\right]\left[\frac{(424\times M)^{(1-H_k)}\,C^{(H-3/2)}}{\sigma_k}\right] \tag{4.2}$$

para N conexões iguais e fazendo:

$\mu_i = m_i$ é taxa média de bit normalizada em relação a capacidade do enlace C

$C = C_2$ é a capacidade necessária normalizada em relação a C tomada do resultado do algoritmo de capacidade equivalente para tráfego auto-similar

$a_i = \sigma_i^2 = \sigma_k^2$ é a variância normalizada em relação a C ou $\left(\frac{\sigma}{C}\right)^2$ ou $\sigma_k = \frac{\sqrt[2]{a_i}}{C}$

$H_i = H_k$ é o parâmetro de Hurst

M é a capacidade do "buffer" em células

os quais, implementados no MATLAB conforme figura 4.14, fornece o número de conexões N baseado na limitação de Q(x) e na largura de faixa disponível C_2.

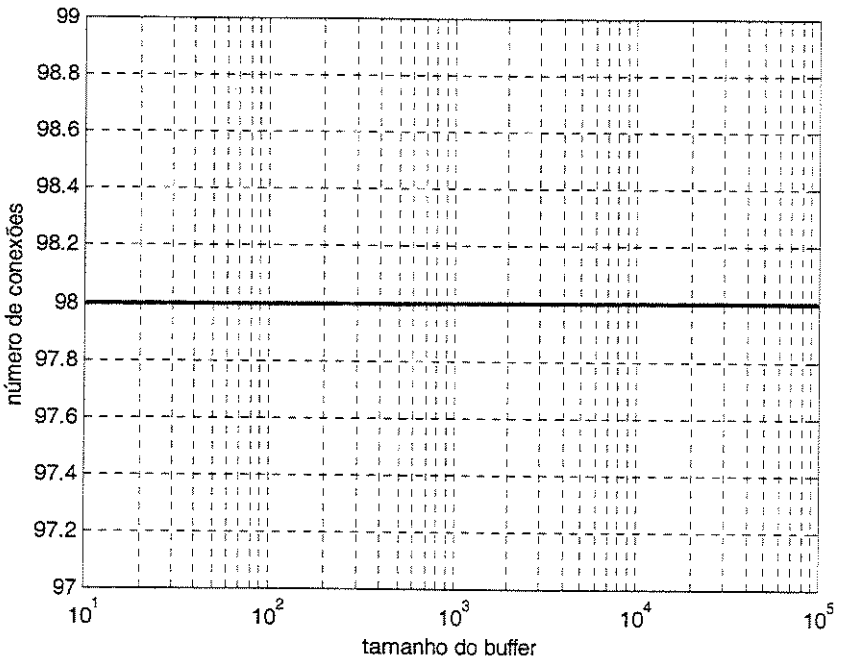

Figura 4.7: Influência do tamanho do buffer na aceitação de chamadas pelo algoritmo de Norros-Tsybakov com H=0.5, CLP=10^{-4} e M=100 células.

Os resultados são mostrados nas figuras 4.15 a 4.18 a seguir:

Regiões de admissão, influência de H e do "buffer"

Conforme as Figuras 4.15 a 4.18 as conexões admitidas pelo algoritmo de Pisa são em número bem maior que no caso da aplicação do algoritmo de capacidade equivalente para tráfego auto-similar de Norros-Tsybakov, assumindo o mesmo comportamento para H=0.5, até o valor de H=0.8 quando o algoritmo de Norros-Tsybakov passa a admitir mais conexões.

Em relação ao tamanho do buffer, quando este é pequeno, o algoritmo de Pisa comparado ao de Norros-Tsybakov (figuras 4.7 a 4.11 e figuras 4.20 a 4.23) admite mais conexões mas quando quando o buffer e H aumentam o algoritmo de Norros-Tsybakov admite maior númro de conexões.

As figuras 4,12 e 4.13 demonstram que, **quando o parâmetro de Hurst tende a 1, existe uma convergência de CLP que vale tanto para o Algoritmo de Norros-Tsybakov quanto para o Algoritmo de PISA, pois os dois algoritmos diferem apenas no modo em que calculam seus limites de conexões, não alterando sua função erro** e assim esta convergência passa a denotar uma característica da rede ATM, com esta configuração de utilização.

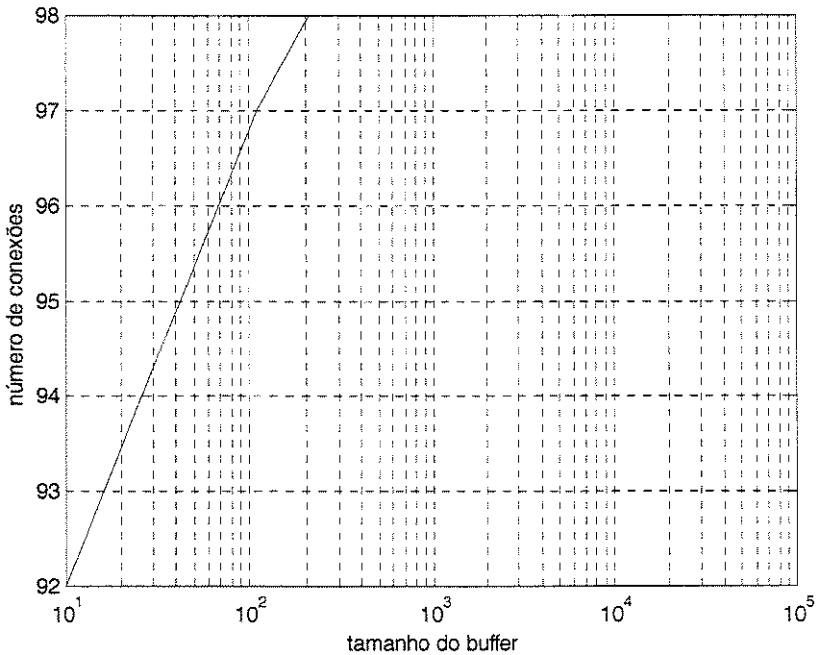

Figura 4.8: Influência do tamanho do buffer na aceitação de chamadas pelo algoritmo de Norros-Tsybakov com H=0.6, CLP=10^{-4} e M=100 células.

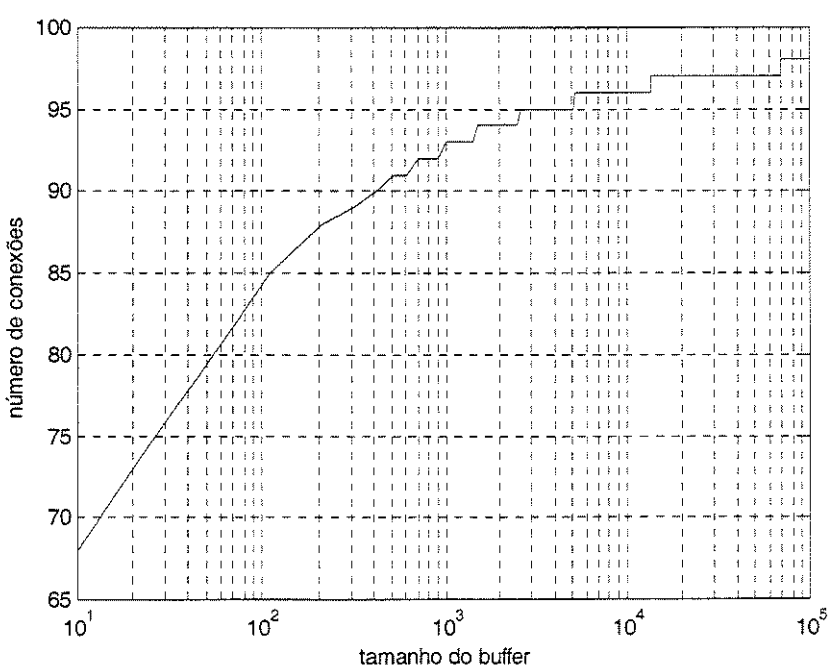

Figura 4.9: Influência do tamanho do buffer na aceitação de chamadas pelo algoritmo de Norros-Tsybakov com H=0.7, CLP=10^{-4} e M=100 células.

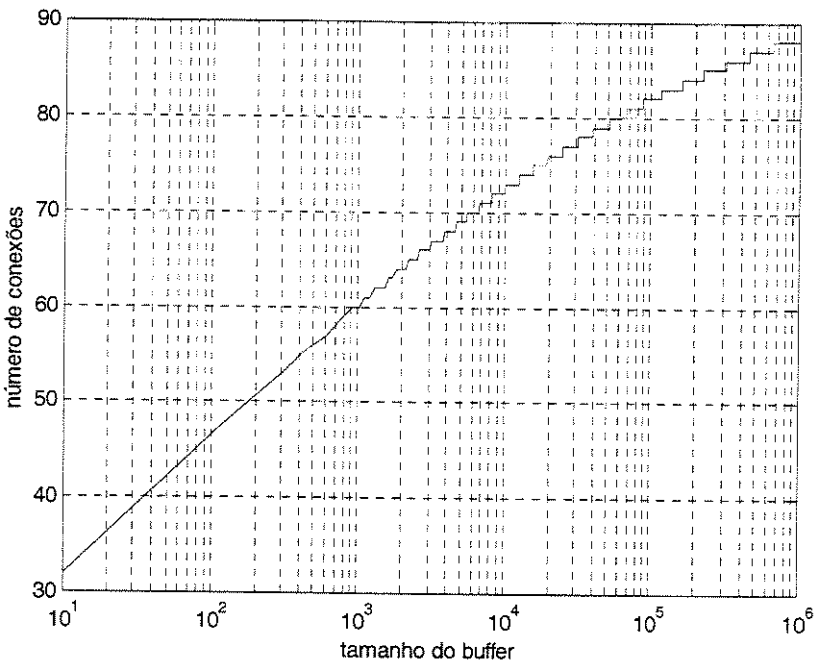

Figura 4.10: Influência do tamanho do buffer na aceitação de chamadas pelo algoritmo de Norros-Tsybakov com H=0.8, CLP=10^{-4} e M=100 células.

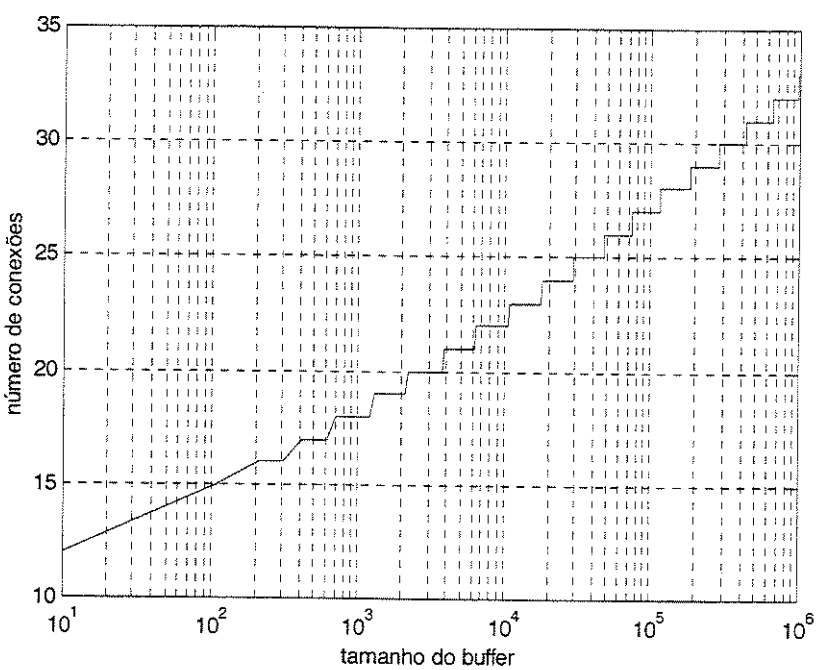

Figura 4.11: Influência do tamanho do buffer na aceitação de chamadas pelo algoritmo de Norros-Tsybakov com H=0.9, CLP=10^{-4} e M=100 células.

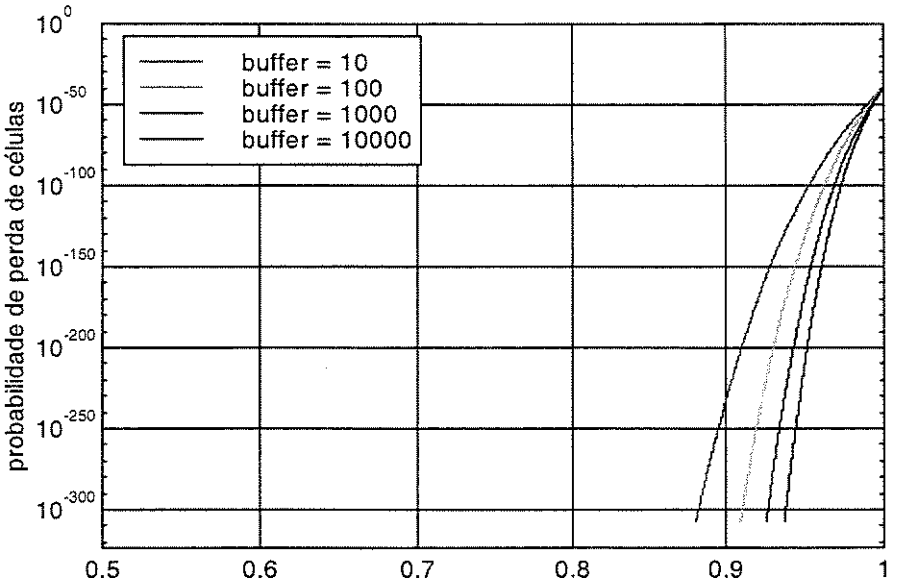

Figura 4.12: Influências do tamanho do buffer e de H no CLP

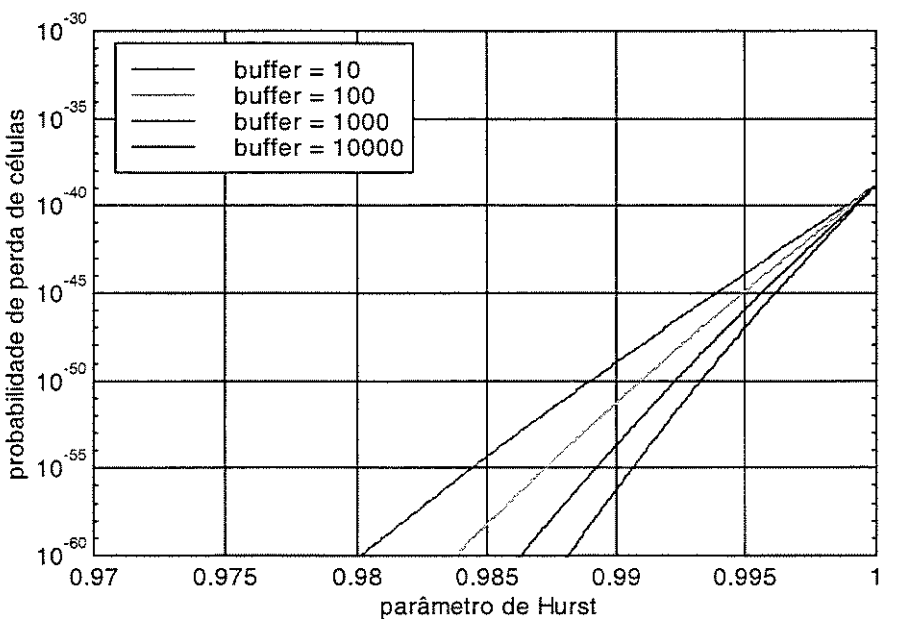

Figura 4.13: Ampliação da Região de convergência da figura 4.16

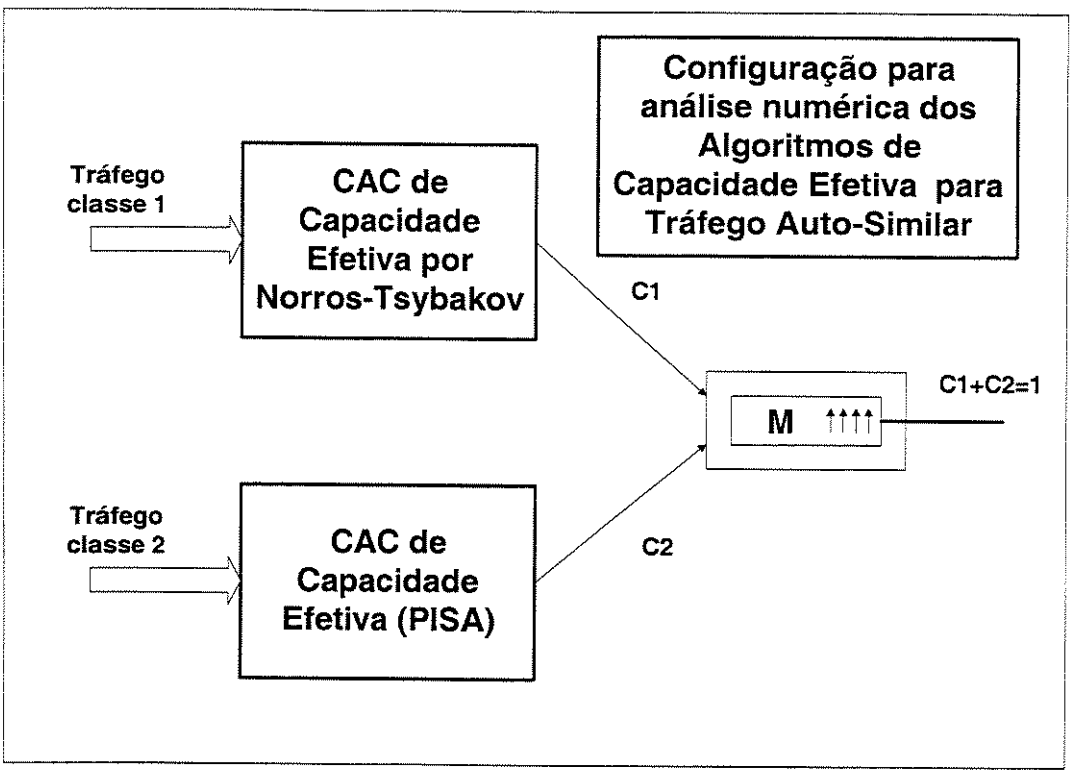

Figura 4.14: Configuração para atuação conjunta dos Algoritmos de Norros-Tsybakov e PISA

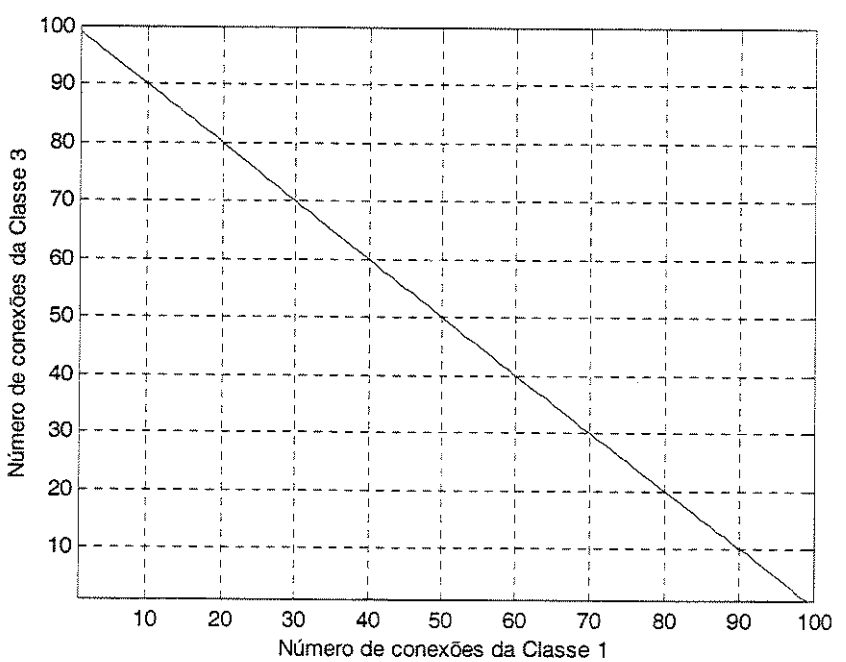

Figura 4.15: Região de aceitação com algoritmo de Pisa (classe 3) em ralação ao algoritmo de Norros-Tsybakov (classe 2) com H=0.5 , CLP=10^{-4} e buffer de M=100 células

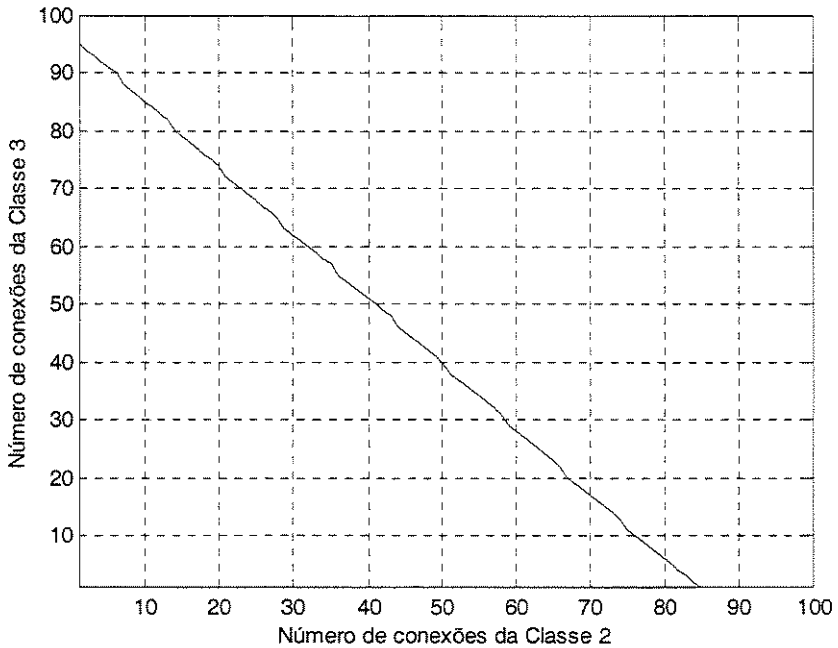

Figura 4.16: Região de aceitação com algoritmo de Pisa (classe 3) em ralação ao algoritmo de Norros-Tsybakov (classe 2) com H=0.7 , CLP=10^{-4} e buffer de M=100 células

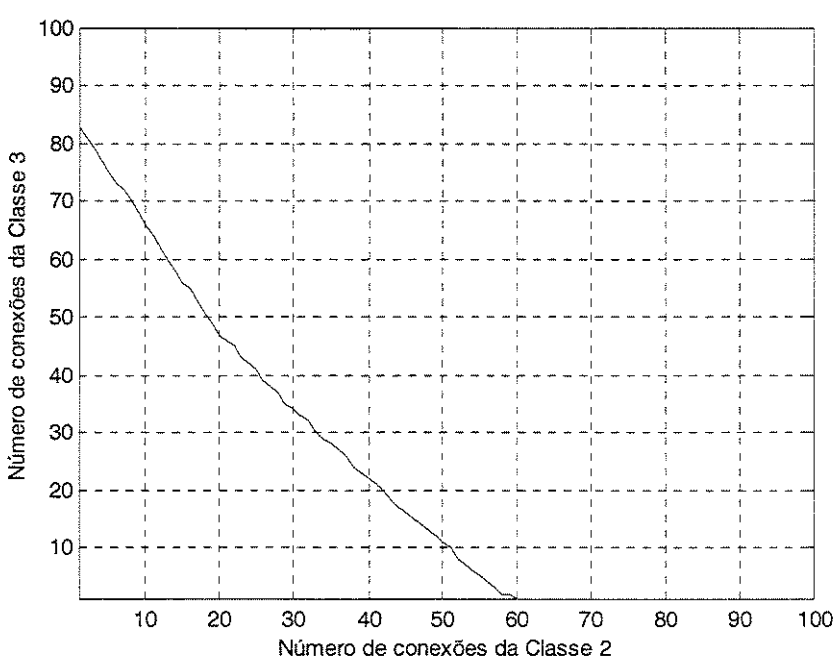

Figura 4.17: Região de aceitação com algoritmo de Pisa (classe 3) em ralação ao algoritmo de Norros-Tsybakov (classe 2) com H=0.8 , CLP=10^{-4} e buffer de M=100 células

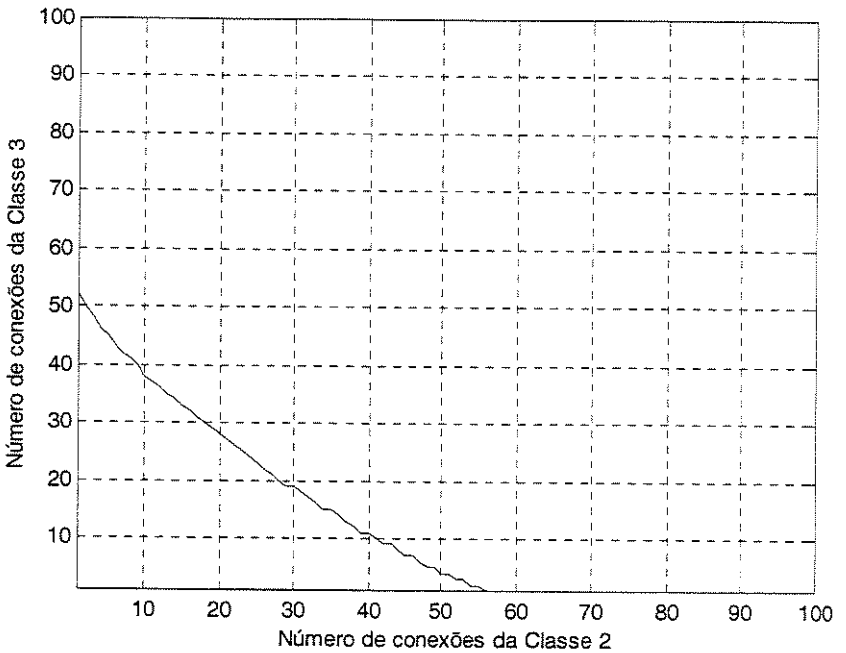

Figura 4.18: Região de aceitação com algoritmo de Pisa (classe 3) em ralação ao algoritmo de Norros-Tsybakov (classe 2) com H=0.9 , CLP=10^{-4} e buffer de M=100 células

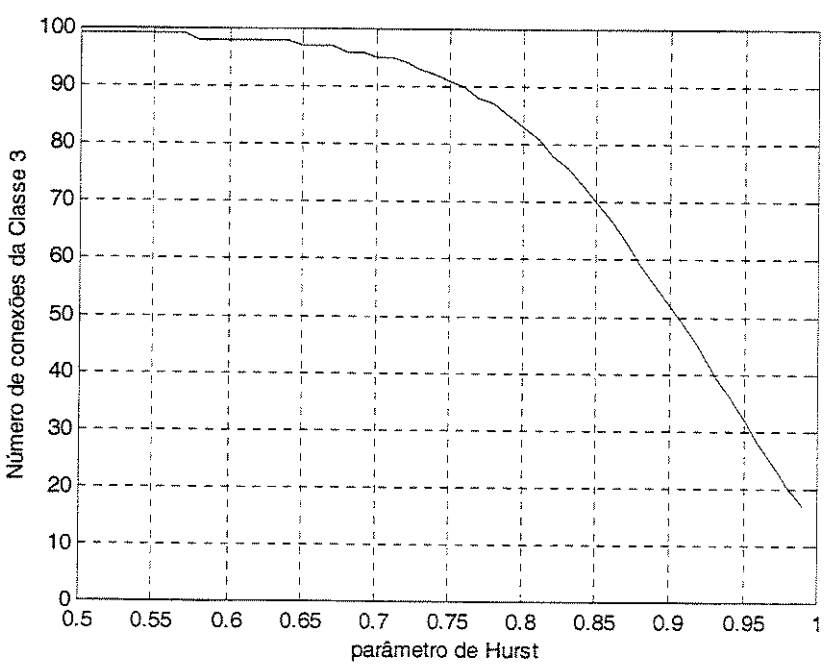

Figura 4.19: Influência de H na aceitação de chamadas com Algoritmo de Pisa(da classe 3) e CLP=10^{-4} e M=100

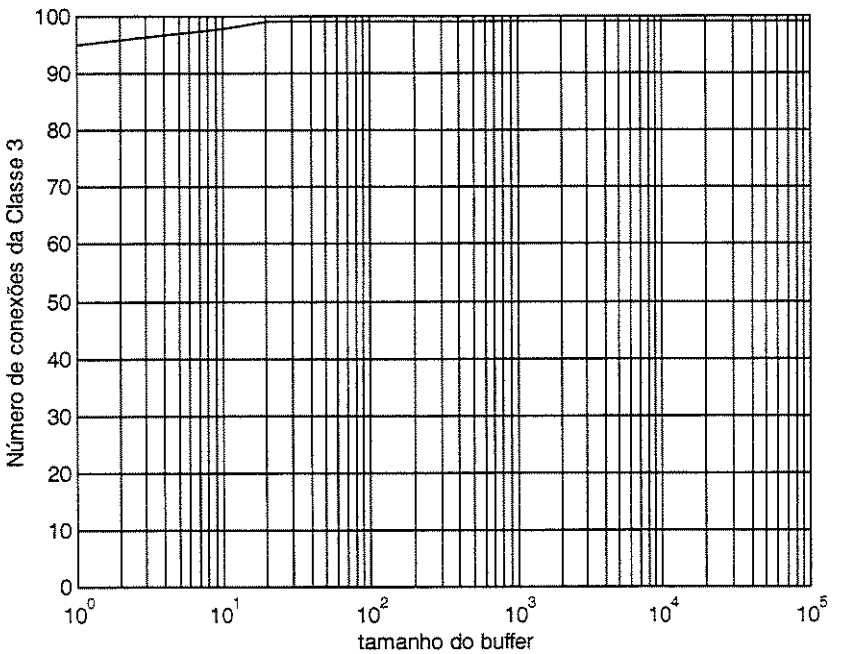

Figura 4.20: Número de conexões aceitas pelo algoritmo de Pisa em função do tamanho do buffer com H=0.5, CLP=10^{-4} e M=100 células.

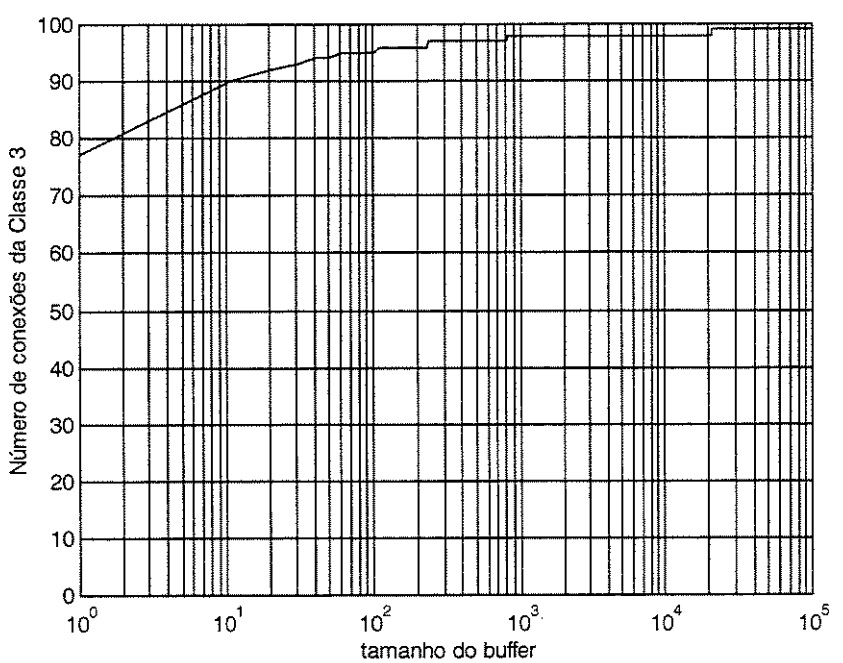

Figura 4.21: Número de conexões aceitas pelo algoritmo de Pisa em função do tamanho do buffer com H=0.7, CLP=10^{-4} e M=100 células.

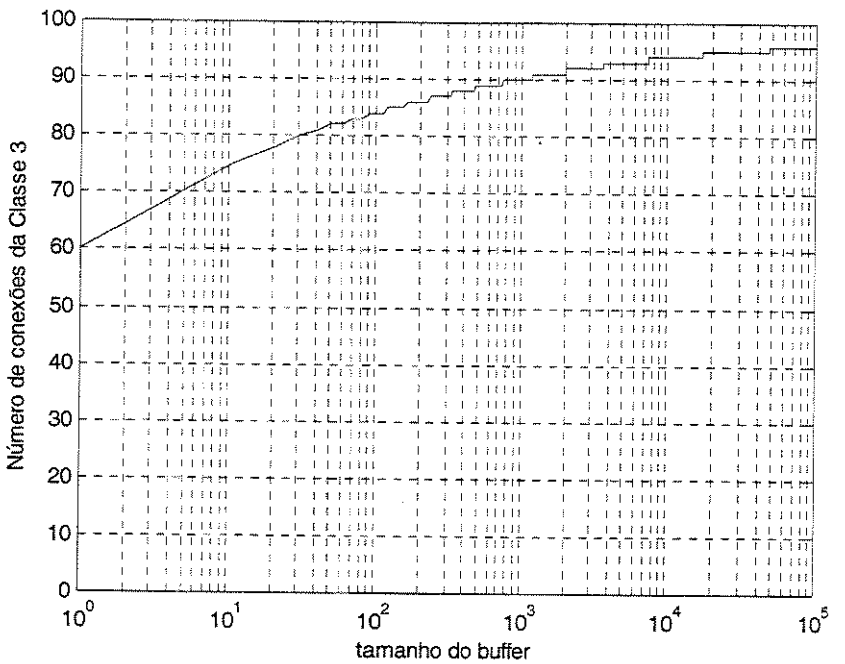

Figura 4.22: Número de conexões aceitas pelo algoritmo de Pisa em função do tamanho do buffer com H=0.8, CLP=10^{-4} e M=100 células.

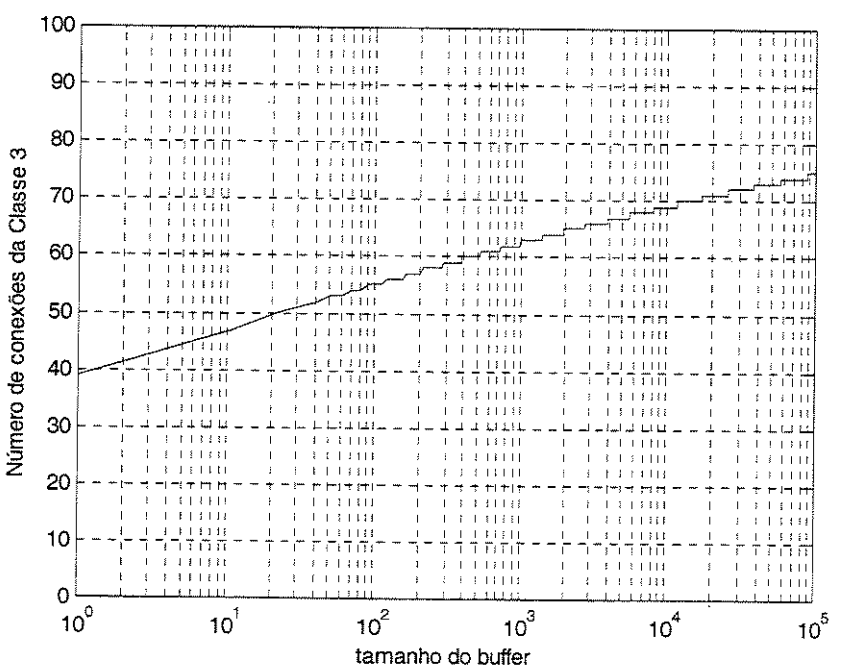

Figura 4.23: Número de conexões aceitas pelo algoritmo de Pisa em função do tamanho do buffer com H=0.8, CLP=10^{-4} e M=100 células.

Capítulo 5

Conclusão

Este trabalho procurou atingir sua meta de demonstrar a metodologia de validação dos algoritmos de Controle de Admissão de Chamadas (CAC) em redes ATM em ambiente de tráfego heterogêneo avaliando, como exemplo, mais positivamente o algoritmo de capacidade efetiva para tráfego auto-similar por Norros-Tsybakov que o de PISA ressaltando-se que o segundo se mostra mais indicado para "buffers" pequenos enquanto que o algoritmo de Norros-Tsybakov se mostrou mais indicado para "buffers" grandes onde um maior número de conexões foram admitidas, com o crescimento de H.

Conforme resultado destes estudos numéricos realizados com o MATLAB Versão 5, nos esquemas CAC apresentados, ficou constatado também que aqueles de tratamento estatístico oneram menos a rede ATM, com ganho estatístico superior à unidade, sendo portanto, mais recomendados.

Como desdobramento, este trabalho conclui ainda que, para o atual ambiente das redes ATM, é necessário um controle preditivo para o tráfego auto-similar com alocação de largura de faixa para que não ocorram perdas de conexões.

Outra conclusão adicional que pode-se extrair do apêndice 1 deste trabalho, é que, no contexto da tecnologia ATM atual, falta uma proposta que seja flexível em aplicações de diferentes tipos de tráfego, que seja adaptativa quando em operação (com relação ao parâmetro H, por exemplo), e que ao mesmo tempo seja simples em processamento, para não sobrecarregar o gerenciamento da rede ATM. *Este assunto será o principal objeto de futuro estudo do autor.*

Cabe salientar, em tempo, que os critérios de simulação variam de artigo para artigo em todos os artigos estudados para este trabalho, distorcendo as comparações de autor para autor e de método para método. Para se obter uma comparação unificada seria necessário a formulação de um método próprio de simulação ou a utilização de um único simulador para refazer estes estudos. Por ser uma tarefa extensa, este estudo torna-se também objeto de trabalho futuro deste autor.

Bibliografia

Bibliografia

[1] ITU-T Traffic Control Congestion in B-ISDN rec.I.371 Bruxelas: ITU,1995.

[2] STALLINGS, W. Data and Computer Commmunications. Quinta Edição. New Jersey: Prentice Hall, 1997.

[3] PRYCKER, Martin de Asyncronous Transfer Mode. Terceira Edição. UK: Prentice Hall International, 1995.

[4] ONVURAL, Ralf ATM - Performance Issues. Segunda Edição. Norwood, MA: Artech House, Inc.,1995.

[5] COST 242 - Final Report of Action (Broadband Network Teletraffic). SPRINGER VERLAG, 1996.

[6] STALLINGS, W. Local & Metropolitan Area Networks.Quinta Edição.New Jersey: Prentice Hall, 1997.

[7] BRISA/EMBRATEL. Arquitetura de Redes de Computadores-OSI e TCP/IP. São Paulo: MAKRON Books, 1994.

[8] SOARES, L. F. G. ; LEMOS, G. e COLCHER, S. Redes de Computadores. segunda Edição. Rio de janeiro: Editora CAMPUS, 1995.

[9] TANENBAUM, A. S. Computer Networks. Terceira Edição. New Jersey: Prentice Hall, 1996.

[10] CHEN, T. M. e LIU, S. S. ATM Switching Systems. Norwood, MA: Artech House, Inc.,1995..

[11] COOVER, E. R. ATM Switches. Norwood, MA: Artech House, Inc.,1997.

[12] IEEE Journal of Selected Areas on Comm.. Advances in ATM for Switching Design. IEEE Press. New York,June 1997

[13] PATTAVINA, A. Switching Theory: Architetures and Performance in Broadband ATM Networks. UK: John Wiley & Sons Ltd, 1998

[14] LEDUC, J. -P. Digital Moving Pictures-Coding and Transmission on ATM Networks Amsterdam, The Netherlands: Elsevier Science B. V.,1994.

[15] ATM-FORUM Traffic Management Specification Version 4.0. New York: ATM Forum, 1996.

[16] SAHA, D.; MUKHERJEE, S. e TRIPATHI, S. K. Multirate Scheduling of VBR Video Traffic in ATM Networks. IEEE Journal of Selected Areas on Communications, pp 1132-1147. aug., 1997.

[17] BONOMI, F. e FENDICK, K. The Rate-based Flow Control Framework for the ABR ATM Service. IEEE Network, pp. 25-39, March/April, 1995.

[18] HONG, D. P. Hong e SUDA, T. Performance of ATM Available Bit Rate for Bursty TCP Sources and Interfering Traffic. Computer Networks, pp. 7-18, feb., 1999.

[19] ONVURAL, Ralf e CHERUKURI, Rao. Signaling in ATM Networks. Norwood, MA: Artech House, Inc.,1997.

[20] BLACK, Uyless ATM-Volume II. New Jersey: Prentice Hall, 1998.

[21] ROBERTAZZI, T. G. Computer networks and Systems. Segunda Edição. New York: Springer-Verlag, 1994

[22] STALLINGS, W. High-Speed Networks, TCP/IP and ATM Design. New Jersey: Prentice Hall, 1996.

[23] GERSHT, A. e LEE, K. J. A Congestion Control Framework for ATM Networks. IEEE Journal of Selected Areas on Communications, pp.1119-1130, sept.,1991.

[24] KLEINROCK, L. Queueing Systems Volume I. UK: John Wiley & Sons Ltd, 1975

[25] KLEINROCK, L. Queueing Systems Volume II. UK: John Wiley & Sons Ltd, 1976

[26] DSHALALOW, J. H. Advances in Queueing. New York: CRC Press, 1995

[27] GROSS, D. e HARRIS, C. M. Fundamentals of Queueing Theory. UK: John Wiley & Sons Ltd, 1998

[28] LELAND, W. E. Leland; TACQQU, M. S. ; WILLINGER, W. e WILSON, D. V. On The Self-Similar Nature of Ethernet Traffic (Extended Version). IEEE/ACM Transactions on Networking, pp. 1-15, vol. 2, no. 1, feb.1994.

[29] PAXSON, V. e FLOYD, S. Wide-area Traffic: The Failure of poisson Modeling. In:. ACM SIGCOMM'94, London, august 1994.

[30] MANDELBROT, B. The Fractal Geometry of Nature. New York: Freeman, 1983.

[31] FROST, V. S. e MELAMED, B. Traffic Modeling For Telecommunications Networks. IEEE Communications Magazine, pp. 70-81, mar./1994.

[32] ADAS, A. Traffic models in Broadband Networks. Georgia Institute of Technology Georgia:1998

[33] HUI, J. Y. Resourse Allocation for Broadband Networks, IEEE Journal of Selected Areas on Communications, pp 1221-1233, dec. /88.

[34] PITTS, J. M.e SCHORMANS, J. A. Introduction to ATM Design and Performance UK: John Wiley & Sons Ltd, 1996

[35] SCHWARTZ, Mischa Broadband Integrated Networks. New Jersey: Prentice Hall, 1996.

[36] ANICK, D. ; MITRA, D. e SONDHI, M. M. Stochastic Theory of a Data-handling System with Multiple Sources. Bell System technical Journal, pp. 1871-1894, vol. 61, n. 8. Oct./ 1982.

[37] LAU, W. -C. e LI, S.-Q. Statistical Multiplexing and Buffer Sharing in Multimedia High-Speed Networks: A frequency-Domain Perspective. IEEE/ACM transaction on Networking, pp. 382-396, june/1997.

[38] PAPOULIS, A. Probability, Random Variables, and Stochastic Processes. Terceira Edição. New-York: McGraw-Hill,1991

[39] GARRET, M. W. e WILLINGER, W. Analysis, Modeling and Generation of Self-Similar VBR Vídeo Traffic. In:. ACM SIGCOMM'94, pp. 269-280. London, august 1994.

[40] WORNELL, G. W. Signal Processing with Fractals A Wavelet-Based Approach. New Jersey: Prentice Hall, 1996.

[41] HURST, H.; BLACK, R. e SIMAIKA, Y. Long-Term Storage: An Experimental Study. London: Constable, 1965.

[42] ARANTES, Dalton S. e OLIVEIRA, Albanita G. D. de Modelos Auto-Similares para Tráfego de Taxa de Bit Variável em Redes ATM (Tese de Mestrado - UNICAMP - 97). Orientador: Prof. Dr. Dalton S. Arantes./ Aluna: Albanita G. D. de Oliveira.

[43] GIORDANO, S. ; PANNOCCHIA, R. e F. Russo. Estimation of Hurst Parameter: Analysis of The Burstiness of Self-Similar traffic Models. Univ. of Pisa-ITALY.

[44] TSYBACOV, B. e GEORGANAS, Nicolas D. . On Self-Similar Traffic in ATM Queues: Definitions, Overflow Probability Bound, and Cell Delay Distribution. IEEE Journal of Selected Areas on Communications. june, 1997..

[45] TSYBAKOV, B. e GEORGANAS, Nicolas D..Self-Similar Processes in Communications Networks. IEEE Transactions. on Information. Theory. sept./ 98.

[46] TSYBAKOV, B. e GEORGANAS, Nicolas D. Overflow Probability in an ATM Queue With Self-Similar Input Traffic. Russian Ac. of Science-RUSSIA e Univ. of Otaa- CANADA.

[47] GRIPENBERG, G. e NORROS, I.. On The Prediction of Fractional Brownian. IEEE Transactions. on Information. Theory. sept./ 96.

[48] GARCIA, A. L.. Probability and Random processes for Electrical Engineering. New YorkAddison Wesley,1994.

[49] BERAN, J., SHERMAN, R., TAQQU, M. S. e WILLINGER,W.. Long-Range Dependence in Variable-Bit-Rate Video Traffic.IEEE Trans. on Comm., feb./mar./ap., 1995.

[50] CSORGO, S. e MIELNICZUK, J.. Density Estimation Under·Long-Range Dependence. The Annals of Statistics. 1995, Vol. 23, número 3.

[51] CSORGO, S. e MIELNICZUK, J Nonparametric Regression Under Long-Range Dependent Normal Errors. The Annals of Statistics. 1995, Vol. 23, número 3.

[52] GROSSGLAUSER, M. e BOLOT, J. -C.. On The Relevance of Long-Range Dependence in Network Traffic. SIGCOMM'96. Outubro,1996.

[53] RYU, B. K. e ELWALID, A. The Importance of Long Range Dependence of VBR Vídeo Traffic in ATM Traffic Engineering: Myths and Realities. SIGCOMM'96.Outubro,1996.

[54] ABRY, P., VEITCH, D. e FLANDRIN, P.. Long-Range Dependence Revisiting Aggregation with Wavelets. Journ. of Time Séries Analysis. may, 1998.

[55] NORROS, I.. On the Use of Fractional Brownian Motion in the Theory of Connectionlesss Networks. IEEE Journal of Selected Areas on Comm.. aug.,1995.

[56] LIKHANOV, N., TSYBAKOV, B. e GEORGANAS, N. D.. Analysis of an ATM Buffer With Self-Similar ("Fractal") Input Traffic. Russian Ac. of Science-RUSSIA e Univ. of Otawa-CANADA.1998

[57] CROVELLA, M. E. e BESTAVROS, A.. Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes. IEEE/ACM Trans. on Networking, dec.,1997.

[58] DAHLHAUS, R.. Efficient Parameter Estimation for Self-Similar Processes. The Annals of Statistics, dec. 1989.

[59] HEYDE, C. C. e YANG, Y.. On Defining Long-Range Dependence. Journal of Applied Probability, dec., 1997.

[60] PERROS, H. G. Perros e ELSAYED, K. M. . Call Admission Control Schemes: A Review. IEEE Commun. Magazine, november 1996.

[61] GIORDANO, S.; PAGANO, M.; PANNOCHIA, R e RUSSO, F.. A New Call Admission Control Scheme Based on The Self Similar Nature of Multimédia Traffic. Department of Information Engineering-University of Pisa 1997.

[62] GUÉRIN, R. ; AHMADI, H. e NAGHSHINEH, M.. Equivalent Capacity and Application to Bandwidth Allocation in High-Speed Networks. IEEE Journal of Selected Areas on Comm., sept., 1991.

[63] GOLESTANI, S. J. "Congestion-free communication in broadband packet networks," IEEE Journal of Selected Areas on Comm., vol. 39, 1991.

[64] FABER, T. e LANDWEBER, L.. "Dynamic Time Windows: packet Admission Control with Feedback," Proc. SIGCOMM'92, pag. 124-35.

[65] ÖZVEREN, C. M. , SIMCOE, R. e VARGHESE, G. . Reliable and Efficient Hop-by-Hop Flow Control. IEEE Journal of Selected Areas on Comm., may, 1995.

[66] KOUVATSOS, Demetres D. Performance Modelling and Evaluation of ATM Networks - Vol. I (IFIP). CHAPMAN & HALL, 1995

[67] TURNER, J. S. , "Bandwidth Management in ATM Networks Using Fast Buffer Reservation,"Proc. Australian Broadband Switching and Services Synp., melbourne, Australia, july,1992.

[68] CHEN, T.; LIU, S., SALAMAM, V., PROCANIK, M. J. e KAVOUSPUR, D. . Monitoring and Control of ATM Networks Using Special Cells. IEEE Network. sept., oct.,/ 1996.

[69] ELWALID, A.,MITRA D., e WENTWORTH, R. H. . A New Apr for Allocating Buffers and Bandwidth to Heterog. Regulated Traffic in an ATM Node IEEE Journal of Selected Areas on Comm., aug., 1995.

[70] VECIANA, G. de; KESIDIS, G. e WALRAND, J.. Resource Management in Wide-Area ATM Networks Using Effective Bandwidths. IEEE Journal of Selected Areas on Comm., aug., 1995.

[71] CHANG, C.-S. e THOMAS, J. A . Effective Bandwidth in High-Speed Digital Networks. IEEE Journal of Selected Areas on Comm., august, 1995.

[72] GALASSI, G. ; RIGOLIO, G., e FRATTA, L.. "ATM: Bandwidth Assignment and Bandwidth Enfourcement Policies,"GLOBECOM'89, 1989.

[73] BOULEAU, N. e LÉPINGLE, D.. Numérical Méthods for Stochastic Processes. JOHN WILEY,1994.

[74] SIMONIAN, A. e GUILBERT, J.. Large Deviations Approximation for Fluid Queues Fed by a Large Number of On/Off Sources. IEEE Journal of Selected Areas on Comm.. Aug. 1995.

[75] WEISS, A.. An Introduction to Large Deviations for Communication Networks. IEEE Journal of Selected Areas on Comm.. aug. 1995.

[76] SAITO, H. e SHIOMOTO, K.. Dynamic Call Admission Control in ATM Networks.IEEE Journal of Selected Areas on Comm., september 1991.

[77] SAITO, H.. Call Admission Control in an ATM network Using Upper Bound on Cell Loss Probability. IEEE Trans on Comm.. Sept., 1992.

[78] AKAR, N., OGUZ, N. C. e SOHRABY, K.. Matrix-Geometric Solutions of M/G/1-Type Markov Chains: A Unifying Generalized State-Space Approach. IEEE Journal of Selected Areas on Comm.. Jun.,1998.

[79] NEUTS, M.. Structured Stochastic Matrices Of M/G/1 Type And Their Applications. Marcel Decker inc., New york and Basel, 1989.

[80] GOLESTANI, S. J.. Network Delay Analysis of a Class of Fair Queueing Algorithms. IEEE Journal of Selected Areas on Comm.. aug. 1995.

[81] KUMAR, P. R.. A Tutorial on Some New Methods for Performance Evaluation of Queueing Networks. IEEE Journal of Selected Areas on Comm., aug. 1995.

[82] JELENKOVIC, P. R.; LAZAR, A. A. e SEMRET, N.. The Efect of Multiple Time Scales and Subexponentiality in MPEG Video Streams on Queueing Behavior. IEEE Journal of Selected Areas on Comm.. aug. 1997.

[83] HIRAMATSU, A.. Integration of ATM Call Admission Control and Link Capacity Control of Distributed Neural Networks. IEEE Journal of Selected Areas on Comm..Sept., 1991.

[84] HABIB, I. W. Applications of Neurocomputing in Traffic Mannagement of ATM Networks. Procedings of IEEE, october 96.

[85] FARAGÓ, A.;BIRÓ, J.; HENK, T. e BODA, M. Analog Neural Optimization for ATM Resource Management. IEEE Journal of Selected Areas on Comm., feb.1997.

[86] MASUGI, M.. A Neural Network Aproach to Cell Loss Rate Estimation for Call Admission Control in ATM Networks. IEICE Trans. on Commun., mar.1997.

[87] MAIR, M. H. W. -Le e SHAE, Z. -Y. Videoconferencing over Packet-Based Networks. IEEE Journal of Selected Areas on Comm.. aug. 1997.

[88] PITSILLIDES, A. e LAMBERT, J. Adaptive Congestion Control in ATM Networks: Quality of Service and High Utilisation. Computer Communications, vol 20, 1997, pp. 1239-1258. ELSEVIER

[89] SRINIDHI, S. M.; THESLING, W. H. e KONANGI, V. K. An Adaptive Scheme for Admission Control in ATM Networks. Computer Networks and ISDN Systems, vol 29, 1997, pp. 569-582. ELSEVIER

[90] CHENG, R. -G. e CHANG, C, -J. Neural Network Connection-Admission Control for ATM Networks. IEE Proc.-Commun., april 97.

[91] MARBACH, P. e TSITSIKLIS, J. N. A Neuro-Dynamic Programming Approach to Admission Control in ATM Networks; The Single Link Case. Laboratory for Information and Decision Systems, MIT-1997

[92] CHANG, P.-R. e HU, J.-T.. Optimal Nonlinear Adaptive Prediction and Modeling of MPEG Video in ATM Networks Using Pipelined Recurrent Neural Networks. IEEE Journal of Selected Areas on Comm.. aug. 1997.

[93] FAN, Z. e MARS, P. Access Flow Control Scheme for ATM Networks Using Neural-Network-Based Traffic Prediction. IEE Proc.-Commun.. oct. 1997.

[94] PITSILLIDES, A.; SEKERCIOGLU, Y. A. e RAMAMURTHY, G. Effective Control of Traffic Flow in ATM Networks Using FERM. IEEE Journal of Selected Areas on Comm., feb.1997.

[95] YOUSSEF, S.A.; HABIB, I. W. e SAADAWI, T. N. A Neurocomputing Controller for Bandwidth Allocation in ATM Networks. IEEE Journal of Selected Areas on Comm., feb. 1997.

[96] UEHARA, K. e HIROTA, K. Fuzzy Connection Admission Control for ATM Based on Possibility Distribution of Cell Loss Ratio. IEEE Journal of Selected Areas on Comm., feb. 1997.

[97] BENSAOU, B; LAM, S. T. C.; CHU, H. -W. e TSANG, D. H. K. Estimation of the Cell Loss ratio in ATM Networks with a Fuzzy System and Application to Measurement-Based Call Admission Control. IEEE Transaction on networking, vol. 5, no. 4, pp. 572-584.

[98] SUM, J.; LEUMG, C.-S. e KAN, W. -K. On The Kalman Filtering Method in Neural-network Training and Pruning. IEEE Trans. on Neural Net.. jan. 1999.

[99] GIROUX, N.e GANTI, S. Quality of Service in ATM Networks: State-of-the-Art Traffic Management, 1999. PRENTICE HALL

[100] ATM Forum Private Network-Network Interface Specification. PNNI Specification. ATM Forum, Kyoto, Japan, nov. 28-dec. 03, pp.1-46, 1994.

[101] COURCOUBETIS, C.; KESIDIS, G.; RIDDER, A. e WALRAND, J. Admission Control and Routing in ATM Networks Using Inferences From Measured Buffer Occupance. IEEE Transactions on Communications, vol 43, nos. 2/3/4 (feb./mar./Ap. 1995), pp. 1778-1784.

[102] CIDON, I.; GOPAL, I. S. e SEGALL, A. Connection Establishment in High-Speed Networks. IEEE Transactions on Networking, pp. 469-481, august/1993.

[103] JORDAN, S. e JIANG, H. Connection Establishment in High-Speed Networks. IEEE Journal on Selected Areas in Communications, pp. 1150-1161, september/1995.

[104] DZIONG, Z. ATM Network Resource Management. McGraw Hill, New York, 1997.

[105] ITU-T Recomendation. Q.2931. ITU, Bruxelas, 1993.

[106] WU, C. -S.; JIAU, J. -C.; CHEN, K. -J. e CHOY, M. Resource Control and Management for Real-Time Call Setup in ATM Networks. Proceedings IEEE, ICC'97, pp.1739-1743, march/1997.

[107] NIEHAUS, D.; BATTOU, A.; McFARLAND, A.; DECINA, B.; DARDY, H.; SIRKAY, V. e EDWARDS, B. Performance Benchmarking of Signaling in ATM Networks.(www.ukansas.edu).

[108] WU, C. -S.; JIAU, J. -C.; CHEN, K. -J. e CHOY, M. Minimizing Call Setup Delay in ATM Networks via Optimal Processing Capacity Allocation. IEEE Communications Letters, april/1998, pp. 110-112.

[109] GELENBE, E.; PUJOLLE, G. e NELSON, J. C. C. Introduction to Queueing Networks. New York, John Wley & Sons, 1987.

[110] PERROS, H. G. e ALTIOK, T. - Editores. Queueing Networks with Blocking. Amsterdam, North holand, 1989.

[111] HAMMER, P. L. - Editor. Queueing Networks with Blocking. Annals of Operations Research, vol. 79, march/1998, pp. 143-207.

[112] WU, J. -L. C. e HU, Y. -C. The Estimation of Signaling Delay in ATM Networks. IEEE Communication Letters, november/1997, pp. 172-174.

Apêndice A

Estabelecimento da conexão ATM e a fase "Setup".

Em paralelo aos aspectos tecnológicos de operação e manutenção das redes ATM estão atrelados os aspectos empresariais e comerciais, pois oriunda dos primeiros está a qualidade de serviço que precisa ser tecnicamente preservada para o cliente, e dos segundos, a integridade da rede ATM cuja falta pode comprometer o atendimento do provedor aos seus usuários de forma mais generalizada.

Quando a rede é corporativa e monolítica ou seja, destinada apenas ao suporte dos processos internos locais de uma empresa, a solução destes problemas assume conotação amena, pois basta evitar o congestionamento para que a comunicação se estabeleça por uma média mínima "aceitável" entre os funcionários desta empresa local, que as relações provedor-clientes estarão satisfeitas. Mas quando a rede passa a ser uma sub-rede ATM, mesmo corporativa, as questões se complicam e o tráfego gerado pelos clientes, bem como os mecanismos de admissão e policiamento empregados pela rede global como um todo passam a fazer diferença.

Na Figura 4.1 [100], observa-se como a disposição e a topologia destas redes global e locais que, em vários níveis (domínios) diferentes (neste caso três domínios), podem influenciar no roteamento e operação da rede ATM, forçando com que os níveis elevados de rede tenham gerência sobre os níveis inferiores.

Como se observa, nas redes ATM do primeiro domínio os elementos A, B, ... são formadas de sub-redes ATM no segundo domínio, que são os elementos A1, A2,....;B1, B2.....,..., e assim por diante, e no terceiro domínio são mostradas as sub-redes A1.1, B1.2,...,E1.1...etc; sendo que cada rede em seu domínio possui um servidor de interligação com outras redes ("líder de rede") e de acordo com a interligação fim-a-fim a rede determina por estes pontos a melhor rota como um todo, bem como os outros aspectos de gerência.

Por exemplo, na Figura 4.1 deseja-se manter conexão entre A2.3 e E3.4. O nível mais elevado achou a rota mais viável A-C-E. No segundo nível, os "líderes de rede" A1-C1-E1 são ligados, conectando-se automaticamente os "líderes de rede" do terceiro

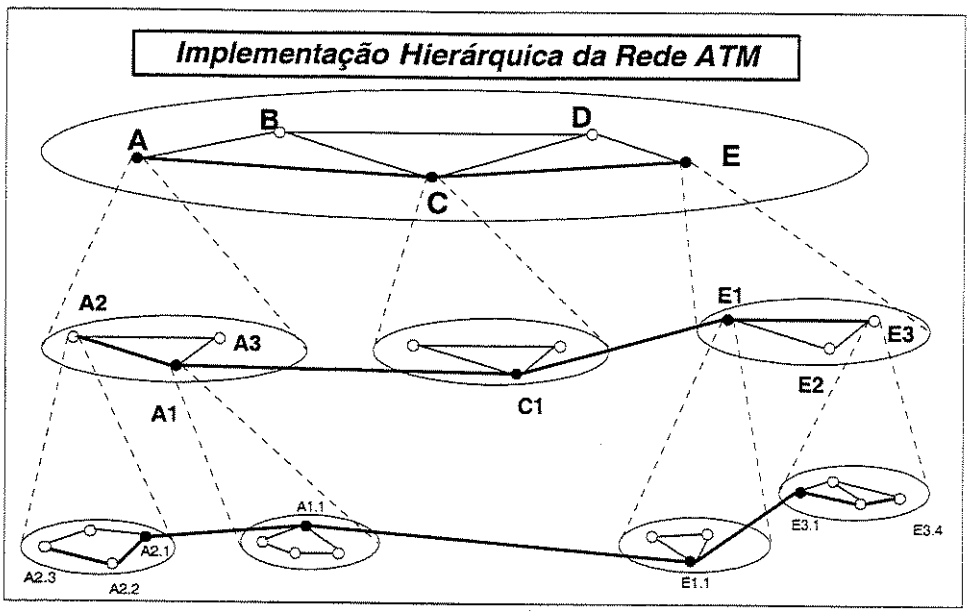

Figura A.1: Implementação Hierárquica da Rede ATM (PNNI)

nível A1.1-C1.1-E1.1. (C1.1 não aparece na Figura). Como A2.3 e E3.4 pertencem às sub-redes A2 e E3 no segundo nível, respectivamente, neste é determinada a conexão A1-A2 e E1-E3 através da ligação dos "líderes de rede" A1.1-A2.1 e E1.1-E3.1. Finalmente, os ponteiros conectam A2.3-A2.1-A1.1-C1.1-E1.1-E3.1 realizando a rota.

O ATM FORUM, especificações PNNI [100] (consultar para maiores detalhes), determina as rotas desta forma usando designações de listas de trânsito (DTL's - Designated Transit Lists) de tal forma que, para o exemplo dado :

DTL1: [A1.1, A2.1, A2.3, C1.1, E1.1, E3.1], pointer 3
DTL2: [A1, A2, C1, E1, E2], pointer 2
DTL3; [A, C, E], pointer 1

Cada DTL possui um número de elementos interligados n_{DTL} e, no caso do exemplo $n_{DTL_1} = 6$, $n_{DTL_2} = 5$ e $n_{DTL_3} = 3$ e os ponteiros ("pointers") indicam o nível de interligação por domínios, indicando a prioridade de interpretação lógica pela ordenação numérica.

Atualmente as estruturas de transporte de informação que interligam redes ATM de locais diferentes são geralmente não-ATM (sistemas rádio digital PCM, SDH,...etc) mas quando uma rede ATM de maiores dimensões recebe muitas chamadas requerendo admissão, o problema é o mesmo, sendo esta interligação tratada como "tranparente" ao processo.

Se a rede ATM de domínio superior é o próprio "backbone" das inferiores (corporativa ou não) assumindo grandes proporções topológicas e altas taxas de dados, as questões de tráfego, mecanismos de admissão, policiamento do tráfego e também agora o tempo de "setup" passam a ser decisivos [104]. Assim, para manter-se ao mesmo tempo

a integridade da rede e o bom atendimento ao usuário são necessárias garantias legais de suporte ao contexto tecnológico.

Generalizando, para uma rede ATM não corporativa o estabelecimento da conexão ATM tem como estágio prévio obrigatório a formalização do Contrato de Prestação de Serviços (CPS) entre o provedor da rede ATM (público ou privado) e o usuário (cliente) desta rede e, que com os problemas enfocados, as negociações anteriores ao CPS esclarecendo as limitações, capacidades de atuação, direitos e deveres de ambas as partes assumem importância fundamental para o futuro bom funcionamento da rede ATM.

A.1. Negociações e contrato

Nesta fase, como bem exposto em Jordan e outros em [103], o provedor deve deixar claro que as regras são diferentes e separadas para as partes envolvidas, sendo que, enquanto o primeiro é o agente policiador de tráfego e a interface obrigatória com outras redes ATM envolvidas, possuindo até parâmetros QoS não negociados com os clientes como CER, SECBR e CMR (vide QoS); o segundo (cliente), deve especificar as características de tráfego, os parâmetros QoS negociáveis de acordo com os serviços envolvidos, tais como CLR, CDVT, SCR, MBS, MCR (vide QoS) e também o atraso fim-a-fim, se estes serviços forem sensíveis ao atraso [15].

Junto ao usuário (estando este ciente ou não), o provedor deve buscar uma completa caracterização das fontes i que irão entrar em tráfego. Além dos parâmetros QoS já citados e da prioridade de células para determinado tráfego, deve-se também obter a taxa de pico de célula-PCR_i, taxa média de célula-mCR_i, variâncias σ_i^2 e a taxa de surtos b_{s_i} para células (vide ítem 3.1, igual para bits), ou seja:

$$b_{s_i} = \frac{PCR_i}{mCR_i} \tag{A.1}$$

sendo estes parâmetros captados de modelos de tráfego convenientes (MMPP, IPP, etc,.vide modelos de tráfego) de acordo com os serviços requeridos, curvas de taxas de surto de acordo com variação de tráfego (serviços) com horários de operação do usuário [103].

Em seqüência a estes dados, a caracterização passa a incluir a largura de banda efetiva-$(BW_e)_i$ para as fontes i a serem habilitadas e possíveis variações desta, de acordo com o comportamento operacional do usuário.

Por parte do provedor, este deve demonstrar a capacidade de absorção destas fontes e garantir para o usuário a manutenção dos parâmetros QoS negociados que reflita a confiabilidade acordada de prestação de serviços (99,9% ou 99,99% do tempo de operação, conforme o caso).

Pode ser mostrada a arquitetura de gerenciamento de recursos-RM e as estratégias de controle de acesso de conexão-CAC, bem como as **regiões de aceitação do QoS** contratado de acordo com as classes de tráfego do cliente (maiores detalhes em

Dziong [104]).

As curvas de ganho estatístico (vide pag. 77, eq. 3.61), levando em conta os fontes i contratadas do cliente em situações de baixo e de alto tráfego nos nós-acesso, devem ser calculadas e juntamente com outros fatores já citados, devem ser comparadas com simulações realizadas pelo provedor (no mercado já existem muitos simuladores disponíveis).

Após esta fase, pode ser firmado o contrato e efetuado o cadastramento do cliente na rede ATM pela configuração de endereços e características em software de base de dados da rede ATM, estipulando um período de funcionamento experimental com condições atenuantes (até recisórias para ambas as partes), onde se observará a ação do policiamento do tráfego pelo provedor e a constatação real dos parâmetros e curvas obtidas teoricamente e por simulação, após o que, o cliente e o provedor estarão mutuamente habilitados a estabelecer conexão.

A.2. A fase "setup"

A conexão é estabelecida então, sob gerenciamento do provedor (o policiamento é realizado após a admissão da chamada) e do cliente. Conforme a Figura 4.2, após o cadastramento do cliente na rede ATM, este passa a fazer parte da tabela da base de dados local e global da rede (foi configurado) [106], tendo acesso à sinalização e, em conseqüência, estando também sujeito aos controles local e global de recursos [19].

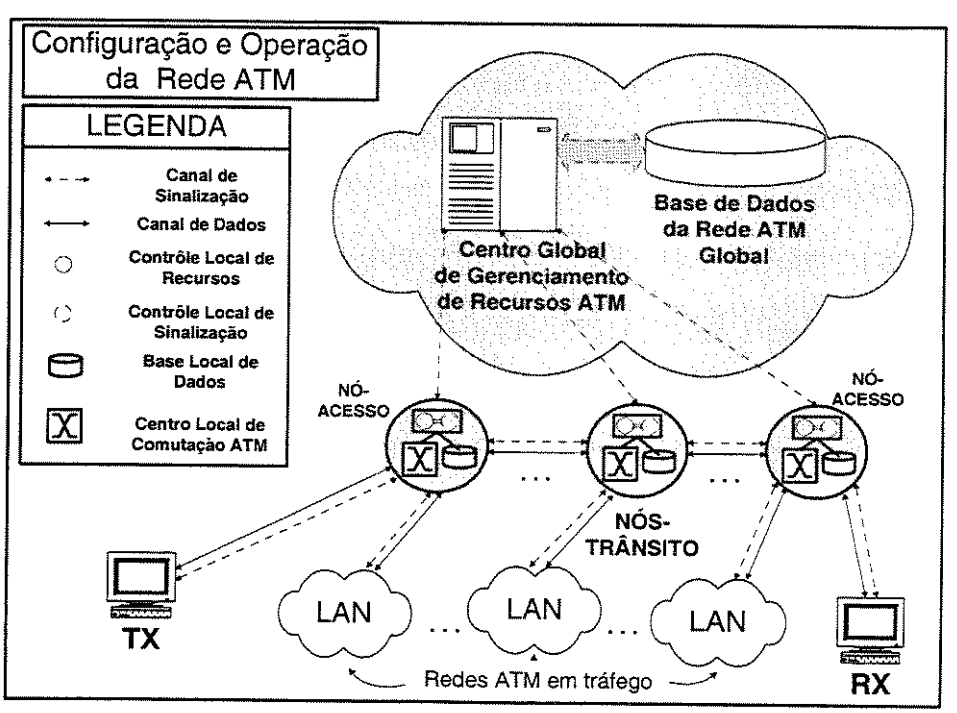

Figura A.2: Operação e Configuração de uma rede ATM

Como se observa, os canais para sinalização são separados dos canais de dados, estando coerentes com a separação de camadas funcionais ATM em camadas de tráfego e de controle (vide Figura 1.13). O acesso de um terminal TX buscando conectividade com o terminal RX se inicia com o compartilhamento de recursos no primeiro nó-acesso com uma rede ATM local já em tráfego. O controle global se encarregará de transportar o pedido de conexão até o outro nó-fim RX e, simultaneamente, gerenciará o compartilhamento de recursos nos nós-trânsito, que também se encontram com outras redes ATM locais já em tráfego.

Os nós de acesso e de trânsito, conforme a Figura 4.2, dispõem de uma base de dados local, de um centro de comutação ATM (vide Figuras 1.11 e 1.12, para se fazer distinção do comutador ATM), cujos comutadores são gerenciados por agentes locais de controle de recursos e de sinalização e supervisionados pela rede global (plano de gerência dos planos da Figura 1.13).

Para entender melhor como ocorre a fase do "setup" é necessário proceder-se a uma revisão sintética sobre a sinalização envolvida no processo, como a seguir.

A.3. A dinâmica do "setup"

Conforme a norma de sinalização do ITU-T Q.2931 [105], o processo de sinalização para o "setup" pode ser descrito com a ajuda da Figura 4.3, sendo que um protocolo mais detalhado pode ser examinado em Cidon e outros [102] e na própria norma citada. Na parte superior da Figura 4.3, vê-se os nós de acesso e trânsito mostrando somente os planos de controle com seus componentes CC (call control/controle de chamada-camada de controle de chamada) e AS (access signaling layer 3/acesso de sinalização-terceira camada) [105].

Na parte inferior da Figura 4.3, encontra-se descrito a própria dinâmica do "setup", pois a partir de comandos (primitivas) de CC, a camada AS gera uma mensagem de "setup" na UNI de TX para o nó-acesso da rede ATM. Esta mensagem de "setup" (1) contém informações que identificam TX e RX, bem como as características desejadas de conexão (tipo de serviço,tráfego, QoS,.. etc). O intervalo T_1 entre a geração de mensagem pela UNI e sua recepção no primeiro nó-trânsito é denominado tempo de transmissão da UNI ou **uni_tx** e é o mesmo no sentido inverso, do RX para o nó-trânsito.

Assim que o primeiro nó (o de acesso) recebe a mensagem de "setup", a rede emite um sinal de retorno a TX, CALL PROC (2) (call proceeding/procedimento de chamada-informa que há pedido de conexão de chamada em curso) com campos VCI/ identificador de circuito virtual e VPI, identificador de caminho virtual, da célula de mensagem de "setup" (vide Figura 1.5-célula ATM) atualizados em caráter provisório, enquanto que, simultaneamente, localiza RX pela base de dados global da rede ATM.

Ainda no primeiro nó de acesso e a partir da tabela de roteamento de suas bases de dados local e global, seleciona-se o melhor caminho virtual (PV) para RX,

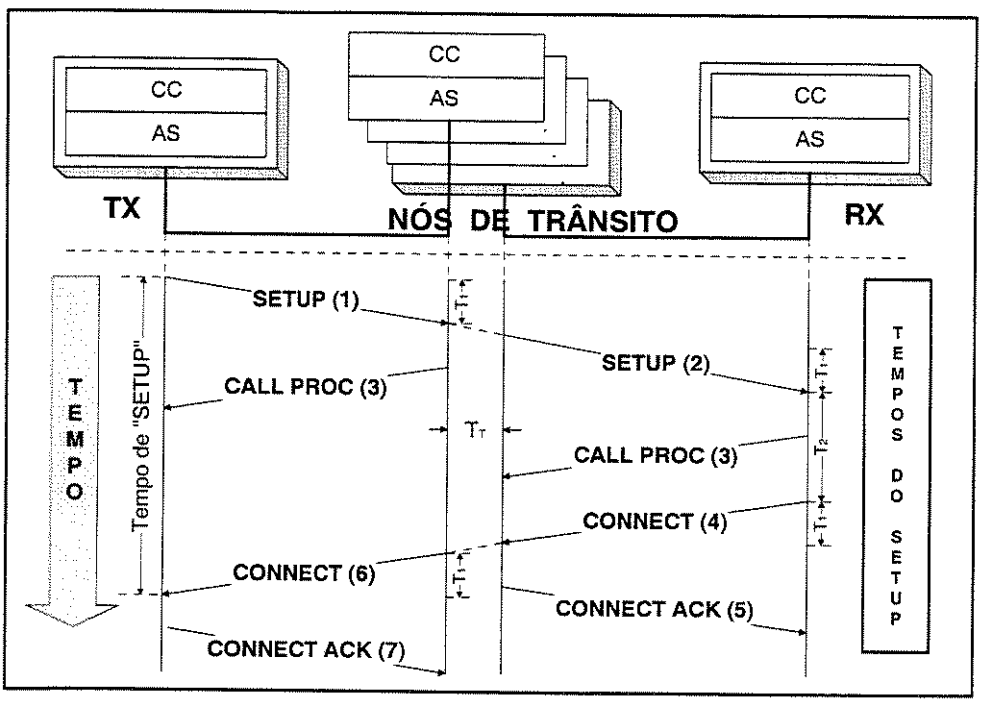

Figura A.3: Sinalização e Tempos de "SETUP" ATM

após o que o nó-acesso envia a mensagem de "setup" por este caminho para todos os nós-trânsito (Roteadores ATM ou NNI's) até RX. O tempo de seleção de rota e o tempo para a mensagem de "setup" passar por todos os nós-trânsito é T_T (denominado de"**network_time**").

Quando esta mensagem de "setup" chega a RX (2) transcorreu o tempo

$$(2 \times T_1 + T_T)$$

e, a partir do nó-acesso para Rx, os campos VCI e VPI da célula de mensagem de "setup" são fixados, já chegando a RX definidos de forma permanente.

Em RX, a mensagem de "setup" também acarreta uma resposta de CALL PROC (3) de retorno ao nó-acesso de RX, informando que há pedido de conexão de chamada em curso e que a UNI de RX necessita de mais tempo para processar sua admissão. Com as informações da mensagem de "setup", RX pode admitir (ou não) a chamada em um tempo T_2 (denominado de **uni_resp**). Se a chamada for admitida, RX envia uma mensagem CONNECT (4), informando a aceitação (caso contrário será enviada a mensagem REJECT, após várias mensagens de CRANKBACK, conforme visto adiante).

O nó-acesso de RX, recebendo o CONNECT, responde a RX com um CONNECT ACK (5) (o que ocorre em todos os nós da rede), informando que a mensagem de aceitação está em curso pela rede e promovendo a atualização das bases de dados locais dos nós da rede ATM (pode também haver falha de CV na rede e, apesar do CONNECT ter sido enviado, a conexão somente será completada se esta chegar a TX).

Do nó-acesso a TX, a mensagem CONNECT (6) é emitida pela rede, chegando a TX e completando o "setup"! Mesmo assim, uma mensagem CONNECT ACK (7) é retornada de TX ao nó-acesso informando a aceitação da chamada a rede. Quando todos os nós da rede ATM receberem o CONNECT ACK a base de dados global da rede ATM é atualizada. A partir daí, inicia-se a transferência de dados de TX à RX e vice-versa.

O tempo total de "setup" T_s será, de forma genérica:

$$T_s = 4 \times T_1 + 2 \times T_T + T_2 \tag{A.2}$$

Quando a rede ATM assume proporções macrodimensionais, este tempo T_s adquire grande importância pois no seu transcorrer outras conexões podem simultaneamente requerer admissão, competindo com recursos e provocando assim um sub- aproveitamento da rede (podendo ou não até cancelar as melhorias obtidas da alocação estatística de BW em relação à determinística). Neste enfoque, como os tempos T_1 (transmissão) não podem ser alterados, resta os tempos T_T (seleção de rotas, admissão nos nós-trânsito e transmissão) e T_2 (processamento de admissão da chamada-CAC em RX) para serem otimizados, visando a diminuição de T_s.

A norma Q.2931 [105], na tentativa de disciplinar a questão, especifica "TIMERS" colocados estrategicamente na rede ATM para limitar estes tempos e não prejudicar as demais conexões que solicitam admissão em outros pontos da rede. Niehaus e outros em [107] comparam o tempo de "setup" T_s com várias topologias e fabricantes diferentes de redes ATM, constatando-se uma faixa de variação de 10ms a 900ms.

Procurando dar mais atenção ao assunto, analisa-se a questão detalhando e identificando os agentes ativos desta dinâmica do "setup".

A.4. Um modelo de rede de filas para o "setup".

Wu e outros em [106], identificam sete agentes ativos que promovem o processo de"setup", que são:

1-UCH (UNI Call Handler): Processador de chamada da UNI, formado por entidades das camadas CC e AS mostradas na Figura 4.3. Assim que CC solicita conexão a AS, o UHC cria um "objeto" de solicitação de chamada, que grava as informações sobre o cliente, atributos de chamada e requerimentos de serviço, gerando as células especiais de "setup" para o nó-acesso. O UCH promove a ativação do QM (3) na UNI de TX e o QT (5) para todo caminho virtual escolhido por PS (4).

2-NCH (NNI Call Handler): Processador de chamada das NNI's ou nós-trânsito, podendo também processar solicitações de chamadas originadas da própria NNI (neste caso a NCH, da mesma forma que o UCH, promove a ativação do QM (3) na NNI do nó original e o QT (5) para todo caminho virtual escolhido por PS (4)). Mas sua principal função é transformar o "objeto" de solicitação de chamada criado pela UHC em "objeto" de solicitação de chamada **em trânsito**, com a ativação do PS (4) por um período de tempo.

3-QM (QoS Mapper): Mapeador de QoS, é ativado pelo UHC (NHC) na UNI (NNI), conforme o originador da chamada, tem como função achar uma classe de serviço (da tabela de classes de serviços das bases de dados local e global) que se enquadre com o tipo de chamada, parâmetros QoS e descritores de tráfego da solicitação requerida. Se a solicitação não se enquadrar em nenhuma classe de serviços o QM retorna uma mensagem de "REJECT" ao TX, rejeitando a solicitação.

4-PS (Path Selector): Seletor de caminho virtual-VP, é ativado no primeiro nó-acesso, na transição do UHC para o NHC, tem a função de escolher um caminho fim-a-fim a partir das tabelas de rotas baseados nos endereços de fonte-destino e policiamento prévio dos caminhos, de acordo com a classe de serviço definida por QM (3). Opera por tentativas e erros, pois um caminho, ao ser "escolhido", não foi ao todo previamente testado no momento da escolha e podem ocorrer falhas no trajeto até RX. Se ocorrer falha de VP, a partir do nó em que esta ocorreu, é retornada uma mensagem de "CRANKBACK de PS" (falha de VP) ao PS que tenta um novo caminho alternativo. Se o número de tentativas exceder um limite prévio estabelecido, a solicitação é rejeitada com o retorno de uma mensagem "REJECT" à UNI de TX [105].

5-QT (QoS Tester): Testador de QoS, é ativado na UNI de TX e tem como função testar o caminho virtual (PV), escolhido por PS até RX, de acordo com a classe de serviços enquadrada por QM na solicitação da conexão. Se, em algum ponto da rede o teste resultar negativo, este promove uma mensagem de retorno ("CRANKBACK de PV") ao PS, que escolhe outro caminho virtual (PV). Resultando positivo, em cada nó-trânsito é gerada uma mensagem de CALL PROC (vide Figura 4.3) ao nó anterior que, como visto, avisa que a solicitação de conexão está em andamento e, com isto, deixa reservado os recursos do caminho já testado.

6-CAC (Call Admission Controler): Controlador de admissão de chamadas, é ativado quando a mensagem de "setup" chega à UNI de RX. De acordo com as informações nela contidas, o CAC admite ou não a conexão (vide capítulo 3, deste trabalho). Este elemento ativo está presente também nos nós-trânsito na forma de CAC nas NNI's ou de CAC nos roteadores (vide Figura 3.3) da rede ATM. Caso a conexão não seja admitida, destes pontos é enviada uma mensagem de retorno ao PS, "CRANKBACK de BW", que indica que não foi possível a alocação da largura de banda (BW) requerida pela conexão solicitada. O PS então escolhe outro PV até o seu limite de tentativas. Se este limite for excedido, a solicitação de conexão é rejeitada por um "REJECT" a partir do PS. Se a conexão for admitida na UNI de RX, a mensagem "CONNECT" é gerada em retorno, fixando os recursos previamente reservados pelos sucessivos "CALLs PROC's" ao longo de PV. A partir daí, como visto, as bases de dados são atualizadas e quando o UCH (NHC) emite o "CONNECT ACK" a tranferência de dados é iniciada.

7-BC (BandWidth Calculator): Calculador de BW é um "ajuste fino" do CAC e é ativado quando a carga do enlace de saída ultrapassar um limite de crítico previamente estabelecido e a classe de tráfego é do serviço VBR, ou seja, um recurso para se evitar que todo o processo seja reiniciado a partir de PS, quando o CAC não

admitir a conexão devido a este estado crítico da rede ATM [106]. Ele recalcula a BW requerida sob condições para o serviço VBR, em função do taxa de surtos de tráfego com alocação estatística precisa. Caso a conexão não seja admitida em BC, é enviada uma mensagem de retorno ao PS, "CRANKBACK de BW", a qual indica que não foi possível a alocação da Largura de Banda (BW) requerida. Então, todo o processo descrito em 6 é executado para reprocessamento da solicitação da conexão.

Com base na identificação destes elementos ativos do "setup", Wu e outros em [108] agruparam estas funções (com exceção de QM, presente somente na UNI de TX na direção considerada de envio do "setup", os demais estão presentes em todos os nós) e os processos de espera (várias conexões com solicitações simultâneas no mesmo PV) distribuidas na rede ATM como se fossem equivalentes a servidores e filas associadas, em vez de procederem a descrição usual em diagrama de blocos do processo. O resultado é uma rede de filas aberta que está representada na Figura 4.4.

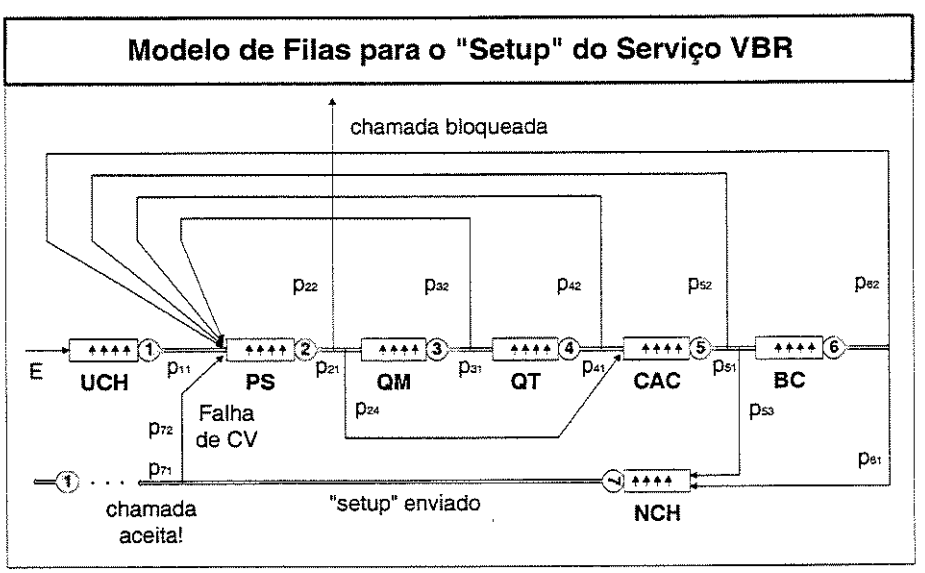

Figura A.4: Modelo de Filas para o Serviço VBR

A rede de filas inicia-se com UCH e termina com NCH, podendo esta representar apenas a interface TX/nó acesso, bem como a conexão TX/RX como um todo (concentrando-se virtualmente os agentes ativos e filas reais distribuídos na rede), e podendo até RX ser o acesso (conexão) a outra rede ATM. Convém lembrar que isto constitui uma interpretação de Wu e outros [106] sobre a norma de sinalização Q.2931 [100].

Identifica-se a rede de filas como uma rede BCMP (Baskett-Chandy-Muntz-Palácios) de classe 3 devido ao conjunto de serviços atendidos $K = (VBR, CBR, UBR)$, e que foi estudada por Gelenbe e Pujole em [109], consistindo de sete processadores, formando o conjunto de nós $I = (UCH, QM, PS, QT, CAC, BC, NCH)$ com filas consideradas inicialmente infinitas (para efeitos de simplificação de análise).

Como para o serviço VBR faz-se uso de todos os nós processadores a rede de filas que melhor o representa é a própria Figura 4.4.

No caso de serviço CBR o nó BC não é utilizado e sua rede de filas é mostrada na Figura 4.5 a), enquanto que o serviço UBR faz uso somente do UCH, PS, CAC e NCH (taxa de bit não é especificada, vide Figura 4.5 b).

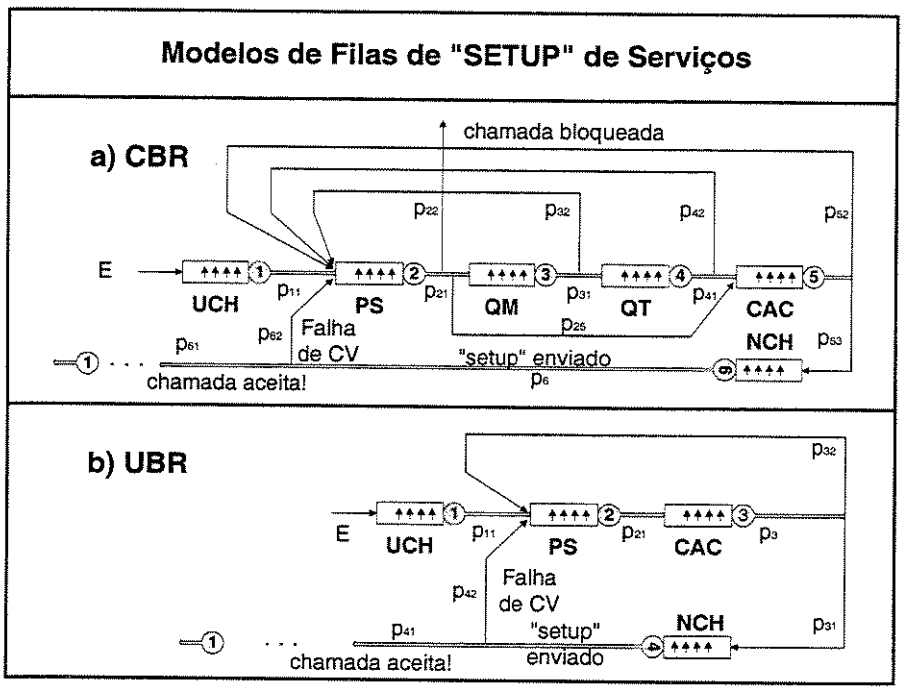

Figura A.5: Modelos de Filas para os Serviços a) CBR e b) UBR

Todos os processos nas Figuras 4.4 e 4.5 já foram descritos anteriormente fazendo-se apenas as adaptações necessárias a cada tipo de serviço e observa-se que, a saída de cada servidor NCH, os "setups enviados" podem ser as mensagens CONNECT/REJECT, se tratar-se de processo na UNI de RX (após o CAC final em RX) do processo visto como um todo (processo único) ou simplesmente SETUP (solicitação de "setup" encaminhado para um estágio posterior, se tratar-se de processo nos nós-acesso e trânsito (estágios intermediários).

A.5. O tempo de "setup" na rede ATM

Wu e outros em [106] e [108] procuraram obter o tempo de "setup" na rede ATM primária ou de domínio mais baixo (sem implementação hierárquica ou seja, como na Figura 4.1, considerando-se somente A1, formada por A1.1...A1.3, sem considerar A2, A, B, etc) em função do processamento da CPU (instruções/segundo) dos nós, utilizando o modelo mais genérico da Figura 4.4 para o serviço VBR. É assumida a distribuição de chegadas como sendo Poissoniana com taxa média E^{κ} (chamadas/seg) e o processamento

médio requerido de h_i^k (instruções/chamada) para a chamada de classe $k \in K$, durante o processo $i \in I$; o roteamento é considerado probabilístico e é denominada a probabilidade p_{ij}, sendo que $(i,j) \in I$, como a probabilidade de que a chamada, estando no processo i, vá para o processo j.

Com a rede BCMP assim caracterizada pode-se utilizar o teorema BCMP que prove uma distribuição "Coxiana" (de Cox, detalhes em Gelenbe e Pujole [109] e Dshalalow [26]) que, em suma, consiste em se considerar para uma distribuição geral de tempos de serviço, uma distribuição exponencial do processamento entre os estágios (nós). O teorema BCMP afirma que a distribuição de cada tipo de chamada em progresso e o atraso médio de processamento associado a um processo arbitrário pode ser derivado de uma solução "forma-produto" [26] [110] [111].

Sintetizando, a partir destas considerações Wu e outros em [106] e [108] obtiveram, sob condições de otimização, que o tempo necessário $T_{setup\ mínimo}$ para se processar a chegada de uma requisição de "setup" é:

$$T_{setup\ mínimo} = \frac{1}{E}\left[\frac{\left(\sum_i \sqrt[2]{H_i}\right)^2}{P - \sum_i H_i}\right] \tag{A.3}$$

onde: E é taxa de chegada de requisições em UCH, ou seja, $E = \sum_{k \in K} E^k$ requisições/segundo;

P é a soma das capacidades de processamento das CPU's P_i dos nós acesso e trânsito ou seja,

$$P = \sum_{i \in I} P_i \text{ (instruções/segundo).}$$

H_i^k é o número médio de instruções (requeridos pelo "setup" da classe de serviço $k \in K$) no nó i quando deste é requerido um processamento médio de h_i^k (instruções/chamada da classe de serviço $k \in K$) e chega a taxa média de chamadas λ_i^k (chamadas/segundo da classe de serviço $k \in K$).

$$H_i^k = h_i^k \lambda_i^k \text{ (instruções/segundo)}$$

Verifica-se que existe uma condição de estabilidade $P - \sum_i H_i > 0$ ou seja $P > \sum_i H_i$, o que indica que:

$$\sum_{i \in I} P_i > \sum_{i \in I} H_i \tag{A.4}$$

ou seja, para que a rede ATM **seja estável**, a soma das capacidades das CPU's nos nós acesso e trânsito tem que ser maior que a soma das velocidades das instruções nestes nós.

Obs.: 1-Para uma rede ATM com serviço VBR em tráfego (7 nós processadores, Figura 4.4), com processamentos idênticos nos nós $\left(\sum_{i\in I} H_i = 7H_i\right)$ e capacidades de CPU também idênticas $\left(\sum_{i\in I} P_i = 7P_i\right)$ e, fazendo-se $P_i = \alpha H_i$ com $\alpha > 1$, da equação 4.3 tem-se:

$$T_{setup\ mínimo\ VBR} = \frac{1}{E}\frac{7}{(\alpha - 1)}$$

e nestes termos, por extensão:

$$T_{setup\ mínimo\ CBR} = \frac{1}{E}\frac{6}{(\alpha - 1)}$$

e

$$T_{setup\ mínimo\ UBR} = \frac{1}{E}\frac{4}{(\alpha - 1)}$$

observa-se que se $\alpha > 1$ for mantido constante será necessário um tempo de "setup" menor para processar mais chamadas. Se α se aproximar muito de 1(a capacidade das CPU's dos nós se aproximar das taxas de instruções a processar) será necessário um tempo bem maior para processar o mesmo número de requisições

2-Wu e Hu em [112] verificaram que os nós TX e de trânsito ajustam-se a um Modelo de Erlang tipo 2 (detalhes em Leon-Garcia [48], Gross e Harris [27] e Dshalalow [26]), pois apresentam duas fases de operação com a troca de primitivas entre CC e AS, enquanto o nó RX se comporta como um modelo de Erlang tipo 3 pois apresenta uma terceira fase com o envio do "CONNECT" ou "REJECT" e assim desenvolveram um estudo dos tempos de "setup" baseado na padronização de topologia da rede ATM em malha para todos os domínios. Como este artigo trata de situações muito particulares, apenas registra-se e indica-se o estudo para a consulta.

Mas deste artigo e do ATM FORUM, especificações PNNI [100] extrai-se a expansão do tempo de "setup" para os outros domínios citados e mostrados na Figura 4.1, tal que o este tempo agora refletindo o tempo total de setup T_{setup} da rede global, para uma dada conexão, será:

$$T_{setup} = T_{request} + T_{connect/reject}$$

onde

$$T_{request} = n_{DTL_x} n_{DTL_{x-1}} \ldots n_{DTL_2} T_{setup\_mínimo\_request} =$$

$$= \left(\prod_{x=2}^{X} n_{DTL_x}\right) T_{setup\_mínimo\_request};$$

$$T_{connect/reject} = n_{DTL_y} n_{DTL_{y-1}} ... n_{DTL_2} T_{setup\_mínimo\_connect/reject} =$$

$$= \left(\prod_{y=2}^{Y} n_{DTL_y} \right) T_{setup\_mínimo\_connect/reject;}$$

onde: X é número de ordem do domínio mais elevado no sentido "REQUEST" (ascendente na rede), contado a partir do domínio mais baixo ($x = 1$) ou seja, o trajeto de $x = 1$ (domínio mais baixo) até X (domínio mais elevado) corresponde ao caminho percorrido pela mensagem SETUP requerendo admissão;

n_{DTL_x} é o número de elementos de trânsito no domínio x;

$T_{setup\_mínimo\_request}$ é o tempo de "setup" considerado no domínio mais baixo ($x = 1$) calculado conforme equação 4.3;

Y é número de ordem do domínio mais elevado no sentido da resposta "CONNECT" ou "REJECT" (descendente na rede), contado a partir do domínio mais baixo ($y = 1$), ou seja, o trajeto de $y = 1$ (domínio mais baixo) até Y (domínio mais elevado) corresponde ao caminho percorrido pelas mensagens CONNECT/REJECT, respondendo ao SETUP;

n_{DTL_y} é o número de elementos de trânsito no domínio y

$T_{setup\_mínimo\_connect/reject;}$ é o tempo de processamento da resposta CONNECT ou REJECT considerado em apenas no domínio mais baixo ($y = 1$), conforme equação 4.3.

Obs.: Sendo o trajeto das mensagens SETUP e CONNECT/REJECT opostos ao domínio mais baixo para o SETUP ($x = 1$) será o mais elevado para o CONNECT/REJECT (Y) e vice-versa.

Considerando-se as topologias para processamento de SETUP, CONNECT e REJECT iguais:

$$\left(T_{setup\_mínimo\_request} = T_{setup\_mínimo\_connect/reject} = T_{setup\ mínimo}\right)$$

pode-se escrever:

$$T_{setup} = (n_{DTL_1} + n_{DTL_X}) \left(\prod_{x=2}^{X-1} n_{DTL_x} \right) T_{setup\ mínimo} \qquad (A.5)$$

onde $T_{setup\ mínimo}$ é dado pela fórmula 4.3 empregada, ou no domínio mais baixo ou no mais elevado da rede ATM.

A.6. Conclusão: "setup"-um problema de otimização em aberto.

Assim como o tempo de "setup" se mostra tão importante quanto a qualidade da admissão da chamada com relação ao tráfego e os requisitos QoS, da mesma forma

que ocorreu o surgimento de muitas alternativas de CAC (capítulo 3) recentemente, poderão surgir mais estudos de impacto destes mecanismos no processamento e tempo de conexão, também levando em conta as características de tráfego auto-similar. O fato de se relacionar a quantidade de instruções nos nós com a capacidade da CPU reafirma a necessidade de preocupação com a quantidade de instruções lógicas no projeto dos Algoritmos de CAC's e de Roteamento.

O presente trabalho já mostrou as formas de CAC atualmente em voga (capítulo 3) que podem obter otimização em termos de adequação aos diferentes modelos de tráfego e requisitos QoS, mas a complexidade de processamento (número de instruções por chamada, tempo por instrução e número de instruções às CPU's) foi ignorada, deixando uma lacuna de pesquisa que, como já discutido, pode ser responsável por sérios comprometimentos à qualidade da rede ATM.

Verifica-se, portando, que o problema de otimização é bem amplo e envolve qualidade dos algoritmos de roteamento, de CAC e a rapidez com que estes atuam refletindo no tempo de "setup".

Apêndice B

Programas MATLAB para o estudo numérico do capítulo 4

A seguir fornecemos os programas em MATLAB versão 5 para a obtenção das figuras do Capítulo 4, *by Henrique A.Mielli Camargo* - **R. A.972410 UNICAMP**, outubro/99

1) Gráficos das figuras 4.2 à 4.5

Região de Aceitação de Chamadas de uma classe de tráfego fractal em função de uma classe de tráfego ATM não fractal.

```
% Região de Admissão do Algoritmo de Capacidade Equivalente por
% Norros/Tsybakov versus Algoritmo de Capacidade Equivalente
% dados constantes
eps = 1e-4 ;
ppfa = 0.4 ;
mi = 0.01 ;
M = 100 ;
% dados
Hk = 0.5;
mk = 0.01 ;
sigmk = 0.01511 ;
gama=(((Hk^Hk)*((1-Hk)^(1-Hk))*sqrt(-2*log(eps)))^(1/Hk))*((155)^((1-Hk)/Hk));
Ck=mk+(gama*(sigmk^(1/Hk))*((424*M)^((Hk-1)/Hk)))*(mk^(1/(2*Hk)));
alfa = sqrt((-2*log(eps)-log(2*pi)));
beta=alfa*(1-ppfa)*(155) ;
yi = (beta*mi)-424*M ;
Ci = (yi+sqrt((yi^2)+(1696*M*ppfa*beta*mi)))/(2*beta*ppfa) ;
```

```
Ncon1T = 3000 ; % Número de conexões da classe 1
vNcon1 = [] ;
vNcon2 = [] ;
vetC1 = zeros(1,Ncon1T);
vetC2 = zeros(1,Ncon1T) ;
for Ncon1=1:Ncon1T
vNcon1 = [vNcon1 Ncon1] ;
Ceq1 = Ncon1*Ci ;
Ceq2 = Ncon1*(mk*(1+alfa*sqrt((1-ppfa)/ppfa)));
C1 = min([Ceq1 Ceq2]);
vetC1(Ncon1) = C1 ;
op = 0 ;
Ncon2 = 0 ;
Ntes = 0 ;
Ctes = 0 ;
while op<1
Ceqk1 = Ncon2*Ck ;
Ceqk2 = Ncon2*(mk*(1+alfa*sqrt((1-ppfa)/ppfa)));
C2 = min([Ceqk1 Ceqk2]) ;
if C2 + C1 < 1
Ntes = Ncon2 ;
Ctes = C2 ;
Ncon2 = Ncon2 + 1 ;
else
Ncon2 = Ntes ;
C2 = Ctes ;
vetC2(Ncon1) = C2 ;
break;
end
end
vNcon2 = [vNcon2 Ncon2] ;
end
LL = 100 ;
for Ncon1=1:Ncon1T
delta = (vNcon2(Ncon1)/LL) ;
if delta > 0
vxaux = Ncon1*ones(1,LL) ;
vyaux = 0:delta:vNcon2(Ncon1);
vyaux = vyaux(1:LL) ;
plot(vxaux,vyaux,'c.'); hold on ;
plot(Ncon1,vNcon2(Ncon1),'b.');
```

```
end
end
% plot(vNcon1,vNcon2,'b.');
v = axis ;
axis([0 M 0 M]);
set(gcf,'Color',[1 1 1]);
grid on ;
xlabel('Número de conexões da Classe 1');
ylabel('Número de conexões da Classe 2');
```

2) Figura 4.6

Influência de H no número de conexões aceitas da classe 2 com M e CLP constantes.

```
% Número das Conexões Fractais versus Parâmetro de Hurst
eps = 1e-4 ;
h1 = [0.5+eps 0.6 0.7 0.8 0.9 ];
sk1 = [0.01411 0.01743 0.02131 0.02621 0.03368 ] ;
[P,S] = polyfit(h1,sk1,7) ;
% dados constantes
eps = 1e-4 ;
ppfa = 0.4 ;
Ri = 0.025 ;
bi = 40 ;
mi = 0.01 ;
M = 100000 ;
% dados
Hk = 0.8 ;
mk = 0.01 ;
sigmk = 0.02621 ;
Ncon1T = 3000 ; % Número de conexões da classe 1
vNcon1 = [] ;
vNcon2 = [] ;
vHk = [] ;
vetC1 = zeros(1,Ncon1T);
vetC2 = zeros(1,Ncon1T) ;
Ncon1 = 1;
for Hk=(0.5+eps):0.01:(1-eps)
vHk = [vHk Hk] ;
gama=(((Hk^Hk)*((1-Hk)^(1-Hk))*sqrt(-2*log(eps)))^(1/Hk))*((155)^((1-Hk)/Hk));
Ck=mk+(gama*(sigmk^(1/Hk))*((424*M)^((Hk-1)/Hk)))*(mk^(1/(2*Hk)));
alfa = sqrt((-2*log(eps)-log(2*pi)));
beta=alfa*(1-ppfa)*(155) ;
yi = (beta*mi)-424*M ;
Ci = (yi+sqrt((yi^2)+(1696*M*ppfa*beta*mi)))/(2*beta*ppfa) ;
vNcon1 = [vNcon1 Ncon1] ;
Ceq1 = Ncon1*Ci ;
med = Ncon1*mi ;
Ceq2 = med + alfa*sqrt(Ncon1*mi*(Ri-mi)) ;
C1 = min([Ceq1 Ceq2]);
op = 0 ;
```

```
Ncon2 = 0 ;
Ntes = 0 ;
Ctes = 0 ;
sigmk = polyval(P,Hk) ;
while op <1
Ceqk1 = Ncon2*Ck ;
std2 = Ncon2*sigmk ;
med2 = Ncon2*mk ;
Ceqk2 = med2 + sqrt(std2*alfa) ;
C2 = min([Ceqk1 Ceqk2]) ;
if C2 + C1 < 1
Ntes = Ncon2 ;
Ctes = C2 ;
Ncon2 = Ncon2 + 1 ;
else
Ncon2 = Ntes ;
C2 = Ctes ;
vetC2(Ncon1) = C2 ;
break;
end
end
vNcon2 = [vNcon2 Ncon2] ;
end
plot(vHk,vNcon2);
set(gcf,'Color',[1 1 1]);
xlabel('parâmetro de Hurst');
ylabel('número de conexões');
grid on ;
```

3) Figuras 4.7 a 4.11

Influência do tamanho do "buffer" no número de conexões aceitas da classe 2 com H e CLP constantes

```
% dados constantes
eps = 1e-4 ;
ppfa = 0.4 ;
Ri = 0.025 ;
bi = 40 ;
mi = 0.01 ;
M = 100 ;
% dados
Hk = 1-eps ;
mk = 0.01 ;
sigmk = 0.090 ;
Ncon1T = 3000 ; % Número de conexões da classe 1
vNcon1 = [] ;
vNcon2 = [] ;
vM = [] ;
vetC1 = zeros(1,Ncon1T);
vetC2 = zeros(1,Ncon1T) ;
Ncon1 = 1;
for M =10:100:1000000
vM = [vM M] ;
gama=(((Hk^Hk)*((1-Hk)^(1-Hk))*sqrt(-2*log(eps)))^(1/Hk))*((155)^((1-Hk)/Hk));
Ck=mk+(gama*(sigmk^(1/Hk))*((424*M)^((Hk-1)/Hk)))*(mk^(1/(2*Hk)));
alfa = sqrt((-2*log(eps)-log(2*pi)));
beta=alfa*(1-ppfa)*(155) ;
yi = (beta*mi)-424*M ;
Ci = (yi+sqrt((yi^2)+(1696*M*ppfa*beta*mi)))/(2*beta*ppfa) ;
vNcon1 = [vNcon1 Ncon1] ;
Ceq1 = Ncon1*Ci ;
med = Ncon1*mi ;
Ceq2 = med + alfa*sqrt(Ncon1*mi*(Ri-mi)) ;
C1 = min([Ceq1 Ceq2]);
op = 0 ;
Ncon2 = 0 ;
Ntes = 0 ;
Ctes = 0 ;
while op <1
```

```
Ceqk1 = Ncon2*Ck ;
std2 = Ncon2*sigmk ;
med2 = Ncon2*mk ;
Ceqk2 = med2 + sqrt(std2*alfa) ;
C2 = min([Ceqk1 Ceqk2]) ;
if C2 + C1 < 1
Ntes = Ncon2 ;
Ctes = C2 ;
Ncon2 = Ncon2 + 1 ;
else
Ncon2 = Ntes ;
C2 = Ctes ;
vetC2(Ncon1) = C2 ;
break;
end
end
vNcon2 = [vNcon2 Ncon2] ;
end
semilogx(vM,vNcon2);
set(gcf,'Color',[1 1 1]);
xlabel('tamanho do buffer');
ylabel('número de conexões');
grid on ;
```

4) Figuras 4.12 e 4.13

Influência do tamanho do "buffer" e do parâmetro de Hurst na Probabilidade de perda de células, Henrique A. Mielli Camargo

```
%Probabilidade de Perda de Célula versus Parâmetro de Hurst e
%Tamanho de "buffer"
close all
clear all
Ck = 0.0163 ;
mk = 0.01 ;
sigmk = 0.0114 ;
M = 10 ;
eps = 1e-4 ;
delta=1e-4 ;
h1 = [0.5+eps 0.6 0.7 0.8 0.9 ];
sk1 = [0.01411 0.01743 0.02131 0.02621 0.03368 ] ;
[P,S] = polyfit(h1,sk1,7) ;
vcor = 'rgbkycm' ;
figure ;
vM = [] ;
M = 1
for Conta = 1:1:4
vHk = [] ;
vEps = [] ;
M = M*10 ;
vM = [vM M] ;
for Hk = (0.5+delta):delta:(1-delta)
sigmk = polyval(P,Hk) ;
A = (Ck-mk)/((sigmk^(1/Hk))*(M^((Hk-1)/Hk))*(mk^(1/(2*Hk))))^Hk ;
Eps = exp(-(((Hk^(-Hk))*((1-Hk)^(Hk-1))*A)^2)/2) ;
vHk = [vHk Hk] ;
vEps = [vEps Eps] ;
end
max(vEps)
semilogy(vHk,vEps,vcor(Conta)); hold on ;
drawnow ;
end
legend('buffer = 10','buffer = 100','buffer = 1000','buffer = 10000',2);
set(gcf,'Color',[1 1 1]);
v = axis ;
```

```
axis([0.98 1 1e-40 1e-38]);
grid on ;
ylabel('probabilidade de perda de células');
xlabel('parâmetro de Hurst');
```

5) Figuras 4.15 a 4.18

Região de aceitação de chamadas ATM número de conexões por algoritmo de Pisa versus número de conexões classe 1 (por algoritmo de Norros-Tsybakov)

```
% Região de Admissão das Conexões admitidas pelo Algoritmo de Pisa
% versus Algoritmo de Capacidade Equivalente para tráfego auto-similar
% por Norros/Tsybakov
close all
clear all
% dados constantes
eps = 1e-4 ;
ppfa = 0.4 ;
Ri = 0.025 ;
bi = 40 ;
mi = 0.01 ;
M = 100 ;
% dados
Hk = 0.9 ;
mk = 0.01 ;
sigmk = 0.03368 ;
X = 3.719 ;
gama=(((Hk^Hk)*((1-Hk)^(1-Hk))*sqrt(-2*log(eps)))^(1/Hk))*((155)^((1-Hk)/Hk));
Ck=mk+(gama*(sigmk^(1/Hk))*((424*M)^((Hk-1)/Hk)))*(mk^(1/(2*Hk)));
alfa = sqrt((-2*log(eps)-log(2*pi)));
beta=alfa*(1-ppfa)*(155) ;
yi = (beta*mi)-424*M ;
Ci = (yi+sqrt((yi^2)+(1696*M*ppfa*beta*mi)))/(2*beta*ppfa) ;
Ncon1T = M ; % Número de conexões da classe 1
vNcon1 = [] ;
vNcon3 = [] ;
vetC1 = zeros(1,Ncon1T);
vetC2 = zeros(1,Ncon1T) ;
for Ncon1=1:Ncon1T
vNcon1 = [vNcon1 Ncon1] ;
Ceq1 = Ncon1*Ck ;
med = Ncon1*mi ;
Ceq2 = med + alfa*sqrt(Ncon1*mi*(Ri-mi)) ;
C1 = min([Ceq1 Ceq2 1-eps]);
vetC1(Ncon1) = C1 ;
Ncon3 = PisaExp((1-C1),Hk,mk,sigmk,M,X);
```

```
vNcon3 = [vNcon3 Ncon3] ;
end
figure
plot(vNcon1,vNcon3);
axis([1 M 1 M]);
set(gcf,'Color',[1 1 1]);
grid on ;
xlabel('Número de conexões da Classe 2');
ylabel('Número de conexões da Classe 3');
```

6) Figura 4.19

```
% Número de Conexões admitidas pelo Algoritmo de PISA versus
% Parâmetro de Hurst
close all
clear all
eps = 1e-4 ;
h1 = [0.5+eps 0.6 0.7 0.8 0.9 ];
sk1 = [0.01411 0.01743 0.02131 0.02621 0.03368 ] ;
[P,S] = polyfit(h1,sk1,7) ;
% dados constantes
eps = 1e-4 ;
ppfa = 0.4 ;
Ri = 0.025 ;
bi = 40 ;
mi = 0.01 ;
M = 100 ;
% dados
Hk = 0.9 ;
mk = 0.01 ;
sigmk = 0.03368 ;
Ncon1T = 3000 ; % Número de conexões da classe 1
vNcon1 = [] ;
vNcon2 = [] ;
vNcon3 = [] ;
vHk = [] ;
vetC1 = zeros(1,Ncon1T);
vetC2 = zeros(1,Ncon1T) ;
Ncon1 = 1;
for Hk=(0.5+eps):0.01:(1-eps)
vHk = [vHk Hk] ;
gama=(((Hk^Hk)*((1-Hk)^(1-Hk))*sqrt(-2*log(eps)))^(1/Hk))*((155)^((1-Hk)/Hk));
Ck=mk+(gama*(sigmk^(1/Hk))*((424*M)^((Hk-1)/Hk)))*(mk^(1/(2*Hk)));
alfa = sqrt((-2*log(eps)-log(2*pi)));
beta=alfa*(1-ppfa)*(155) ;
yi = (beta*mi)-424*M ;
Ci = (yi+sqrt((yi^2)+(1696*M*ppfa*beta*mi)))/(2*beta*ppfa) ;
vNcon1 = [vNcon1 Ncon1] ;
Ceq1 = Ncon1*Ck ;
med = Ncon1*mi ;
Ceq2 = med + alfa*sqrt(Ncon1*mi*(Ri-mi)) ;
```

```
C1 = min([Ceq1 Ceq2]);
op = 0 ;
Ncon2 = 0 ;
Ntes = 0 ;
Ctes = 0 ;
sigmk = polyval(P,Hk) ;
while op <1
Ceqk1 = Ncon2*Ck ;
std2 = Ncon2*sigmk ;
med2 = Ncon2*mk ;
Ceqk2 = med2 + sqrt(std2*alfa) ;
C2 = min([Ceqk1 Ceqk2]) ;
if C2 + C1 < 1
Ntes = Ncon2 ;
Ctes = C2 ;
Ncon2 = Ncon2 + 1 ;
else
Ncon2 = Ntes ;
C2 = Ctes ;
vetC2(Ncon1) = C2 ;
break;
end
end
Ncon3 = PisaExp((1-C1),Hk,mk,sigmk,M,3.719);
vNcon3 = [vNcon3 Ncon3] ;
vNcon2 = [vNcon2 Ncon2] ;
end
plot(vHk,vNcon2);
set(gcf,'Color',[1 1 1]);
xlabel('parâmetro de Hurst');
ylabel('número de conexões');
grid on ;
figure
plot(vHk,vNcon3);
set(gcf,'Color',[1 1 1]);
v = axis ;
axis([v(1) v(2) 0 M]);
grid on ;
xlabel('parâmetro de Hurst');
ylabel('Número de conexões da Classe 3');
```

7) Figura 4.20 a fig. 4.23

```
% Número de Conexões admitidas pelo Algoritmo de PISA versus
% tamanho do Buffer
close all
clear all
% dados constantes
eps = 1e-4 ;
ppfa = 0.4 ;
Ri = 0.025 ;
bi = 40 ;
mi = 0.01 ;
M = 100 ;
% dados
Hk = 0.9 ;
mk = 0.01 ;
sigmk = 0.03368 ;
Ncon1T = 3000 ; % Número de conexões da classe 1
vNcon1 = [] ;
vNcon2 = [] ;
vNcon3 = [] ;
vM = [] ;
vetC1 = zeros(1,Ncon1T);
vetC2 = zeros(1,Ncon1T) ;
Ncon1 = 1;
for M =1:10:100000
vM = [vM M] ;
gama=(((Hk^Hk)*((1-Hk)^(1-Hk))*sqrt(-2*log(eps)))^(1/Hk))*((155)^((1-Hk)/Hk));
Ck=mk+(gama*(sigmk^(1/Hk))*((424*M)^((Hk-1)/Hk)))*(mk^(1/(2*Hk)));
alfa = sqrt((-2*log(eps)-log(2*pi)));
beta=alfa*(1-ppfa)*(155) ;
yi = (beta*mi)-424*M ;
Ci = (yi+sqrt((yi^2)+(1696*M*ppfa*beta*mi)))/(2*beta*ppfa) ;
vNcon1 = [vNcon1 Ncon1] ;
Ceq1 = Ncon1*Ci ;
med = Ncon1*mi ;
Ceq2 = med + alfa*sqrt(Ncon1*mi*(Ri-mi)) ;
C1 = min([Ceq1 Ceq2]);
op = 0 ;
Ncon2 = 0 ;
Ntes = 0 ;
```

```
Ctes = 0 ;
while op <1
Ceqk1 = Ncon2*Ck ;
std2 = Ncon2*sigmk ;
med2 = Ncon2*mk ;
Ceqk2 = med2 + sqrt(std2*alfa) ;
C2 = min([Ceqk1 Ceqk2]) ;
if C2 + C1 < 1
Ntes = Ncon2 ;
Ctes = C2 ;
Ncon2 = Ncon2 + 1 ;
else
Ncon2 = Ntes ;
C2 = Ctes ;
vetC2(Ncon1) = C2 ;
break;
end
end
Ncon3 = PisaExp((1-C1),Hk,mk,sigmk,M,3.719);
vNcon3 = [vNcon3 Ncon3] ;
vNcon2 = [vNcon2 Ncon2] ;
end
semilogx(vM,vNcon2);
set(gcf,'Color',[1 1 1]);
xlabel('tamanho do buffer');
ylabel('número de conexõs');
grid on ;
figure
semilogx(vM,vNcon3);
set(gcf,'Color',[1 1 1]);
v = axis ;
axis([v(1) v(2) 0 100]);
grid on ;
xlabel('tamanho do buffer');
ylabel('Número de conexões da Classe 3');
```

Apêndice C

Glossário dos acrônimos utilizados.

AAL - ATM Adaptation Layer, camada de Adaptação ATM. Camada do modelo ATM em que são especificadas as funções relativas à adaptação de mensagens que trafegam entre o programa aplicativo em execução e a camada ATM e se subdivide em subcamada de segmentação e recomposição e subcamada de convergência (vide figura 1.13). Em suma, AAL define como os dados são encapsulados dentro das células e como são reconstituídos no destino conforme a classe de serviço (mais detalhes na figura 1.14).

ABR - Avaliable Bit Rate, Taxa de Bit Disponível. É a capacidade de transferência visando evitar congestionamento por um controle preventivo implementado, ou seja, um controle reativo por meio do qual a taxa em que o usuário pode transmitir é dinamicamente ajustada pela rede ATM.

ABT - ATM Block Transfer, Transferência de Blocos ATM. Neste serviço padronizado pelo ITU-T. o próprio usuário é habilitado a definir e controlar uma estrutura de bloco em seu fluxo de dados. O parâmetro negociado para cada bloco é o PCR, pela taxa média extraída do gerenciador de recursos-RM.

AH - Application Header, cabeçalho de aplicação. Cabeçalho que um segmento de dados de aplicação do usuário recebe na camada de aplicação do modelo OSI da ISO para ser encapsulado pela camada de apresentação.

ARPA - Advanced Research Project Agency. Projeto iniciado em 1968 e que perdurou durante a década de 70. Financiado pelo Departamento de defesa dos EUA visando integrar as redes de computadores das universidades americanas na época e que fez surgir o conceito de comutação de pacotes, protocolo de transferência de arquivos e o protocolo de terminal virtual.

ATC - ATM Tranfer Capabilities, transferências de capacidades ATM. Após a camada ATM ter dado formação à célula ATM, esta é liberada à camada física, podendo constituir fontes com diversos tipos de tráfego **por serviço ATM** ou ATC's

ATM FORUM - Organização internacional independente composta por fabricantes, vendedores, provedores de serviços e usuários da tecnologia ATM. Seu papel é complementar ao dos organismos formais de padronização. Gera especificações cujo principal objetivo é promover a interoperabilidade de produtos e serviços.

bit CLP Bit de Prioridade de Perda de Célula - Campo da célula ATM (vide Figura 1.5) ocupado por 1 bit, assume valor"0" para células prioritárias e valor "1" para células de menor prioridade.

BW - Band Width, largura de banda. Em redes ATM designa a capacidade dos enlaces que conectam os elementos da rede entre si, em bits por segundo ou células por segundo.

CAC - Connection Admission Control ou Call Acceptance Control, controle de admissão de conexão ou de chamada. Função da camada de rede ATM no plano de controle que é executada durante a fase de "setup" de uma conexão, que procura avaliar se a rede tem condições de suportar a QoS requerida e se a aceitação da nova conexão não terá impacto sobre as conexões já em andamento.

CBR - Constant Bit Rate, taxa de bit constante. Requer que uma taxa fixa de dados seja mantida pelo provedor ATM (estipulado previamente no contrato e controlado através da ação da camada de controle ATM à camada ATM, propriamente dita). O ITU-T a denomina de DBR (Deterministic Bit Rate,taxa de bit determinística).

CDV - Cell Delay Variation, variação do atraso de célula. É a parte variável do CTD e é resultante do gerenciamento do fluxo de células pela "bufferização" nos comutadores e multiplexers. É devido ao fluxo de uma mesma conexão e a superposição do fluxo de várias conexões em andamento (vide páginas 18 a 21). Durante a fase "setup" pode ser negociado em diferentes valores, dependendo das conexões em andamento, mas a variação máxima destes valores, o CDVT, tem que ser negociado antes do "setup", pois será utilizado para o policiamento do tráfego na rede ATM.

CDVT - Cell Delay Variation Tolerance, tolerância do CDV. Variação máxima permitida para o CDV e que deve ser declarada e negociada previamente antes do "setup".

CER - Cell Error Rate, taxa de erro de células. Limite imposto pelo provedor. Razão entre células recebidas erradas numa transmissão longa e o total de células recebidas nesta transmissão (vide página 24).

CLP - Cell Loss Probability, probabilidade de perda de células. Neste trabalho tomado com significado análogo ao CLR, para facilitação do tratamento para CLP(QoS), porém com sentido probabilístico ou estimado para uma transmissão longa futura. Difere-se aqui do CLP comumente tratado por Cell Loss Priority, prioridade de perda de célula, e que chama-se aqui de bit CLP (vide bit CLP).

CLP(QoS) - Cell Loss Probability (Quality of Service), probabilidade de perda de célula (qualidade de serviço). Um dos parâmetros definidos pelo conjunto de parâmetros de uma conexão chamado de qualidade de serviço requerido ou QoS requerida e que é negociado durante a fase de "setup".

CLR - Cell Loss Rate, taxa de perda de células. É a razão entre o número medido de células perdidas numa transmissão longa e o total medido de células transmitidas nesta transmissão.

CMR - Cell Mininsertion Rate, taxa de células mini-inseridas. Limite imposto pelo provedor da rede ATM. É a razão, no tempo, de células inseridas em outras conexões por erro de cabeçalho da célula.

CTD - Cell Tranfer Delay, atraso de transferência de células. é a soma do atraso fixo e o CDV de células de uma conexão ATM (vide página 16).

VC - Virtual Circuit, Circuito Virtual (CV). Via lógica definida pela fase "setup" de uma conexão ATM e que escoará o fluxo de pacotes ATM durante toda uma conexão.

DBR - Deterministic Bit Rate, taxa de bit determinística. Vide CBR.

DD - Depacketization Delay, atraso de desempacotamento. É um atraso que é introduzido no destino para garantir que uma conexão ATM, que represente um serviço em tempo real, como conversão em pacotes ATM de sinais de áudio e vídeo, não apresente superposição de células. É uma das causas do CDV (vide página 18).

DSI - Digital Speech Interpolation, interpolação digital de voz. Técnica de multiplexação digital que promove a ocupação de intervalos de silêncio de conversação por canais ativos. É o TASI digital, vide TASI.

EBW - Estatistical Bandwidth, largura de banda estatística. Capacidade de transmissão de um enlace que considera as lacunas aleatórias das conexões em surtos ou jatos. É utilizada na multiplexação estatísticas de conexões ATM (vide figura 3.1).

FBR - Fast Buffer Reservation, reserva rápida de "buffer". Técnica determinística de admissão de chamadas ATM que promove a alocação de banda por reserva de partição de "buffer" a partir do auxílio de células de sinalização com informações de requisitos QoS (vide páginas 57 a 62).

FSD - Fixed Switch Delay, atraso fixo de comutação. É o atraso da célula ATM devido aos componentes físicos do comutador ATM. Compõe o atraso fixo da célula ATM que é determinístico (vide página 1.17).

FTP - File Transfer Protocol, protocolo de transferência de arquivos. É o protocolo responsável pela transferência de arquivos entre sistemas dentro da INTERNET,

compatibilizando diferenças entre aplicações dos equipamentos envolvidos, além de garantir a integridade dos dados transferidos, que podem ser bancos de dados, imagens, programas,etc.

GCRA - É um algoritmo de policiamento de tráfego que confere todas as células para ver se há conformidade com os parâmetros requeridos de QoS para um circuito virtual. Tem dois parâmetros de referência: $PCR(Rp)$ e τ $(CDVT)$ (vide página 30).

HEC - Header Error Control, controle de erro de cabeçalho. Campo do cabeçalho da célula ATM (vide figura 1.5) que tem a dupla função de controlar os erros de cabeçalho e de delineamento da célula ATM.

IAB - Internet Activities Board, quadro para atividades da INTERNET. Equipe criada em seqüencia ao projeto ARPA para desenvolvimento e pesquisas para a evolução da INTERNET.

IBP - Interrupted Bernoulli Process, processo de Bernoulli interrompido. É um processo estocástico utilizado para modelagem de tráfego de dados, um caso especial do processo de Markov modulado Bernoulli (MMBP, vide página 46) ou um MMBP cuja correlação entre os tempos de interchegadas é zero.

IBT - Intrisic Burst Tolerance, tolerância intrínseca de surto. Um dos parâmetros diretores do GCRA juntamente com o PCR e SCR (vide QoS) e que está relacionado com o tamanho máximo do surto.

IFP - Interrupted Fluid Processes, processo de fluido interrompido.

INTERNET - Rede de comunicação de dados de alcance mundial, baseada em protocolos da família TCP/IP.

ISO - International Organization for Standardization, organização internacional de padronização. Órgão independente internacional de padronização.

ITU-T - International Telecommunication Union-Telecommunication Standardization Sector, União Internacional de telecomunicação-Setor de padronização em telecomunicações. Órgão internacional de padronização na área de telecomunicações: é um comitê da ITU, agência mantida pela ONU. Antigo CCITT.

LAN - Local Area Network, rede local. Rede de computadores que cobre uma área geográfica relativamente reduzida (normalmente um único andar ou o interior de um único edifício).

MAN - Metropolitan Area Network, rede metropolitana. Rede de computadores que cobre uma área geográfica maior que a da LAN e menor que a da WAN (uma cidade, por exemplo). Envolve distâncias da ordem de dezenas de quilômetros.

MBS - Maximum Burst Size, tamanho máximo de surto. É o tamanho máximo de um surto de células ATM que podem ser transmitidas no serviço VBR, durante a ocorrência do PCR e que depende do SCR e do CDVT (vide equação 1.10).

MCR - Minimum Cell Rate, taxa mínima de células. É especificado para o serviço ABR. Ele define o mínimo compromisso requerido à rede pela fonte ABR em taxa de células. A quantidade $[(PCR) - (MCR)]$ representa um componente elástico do fluxo de dados para que a rede forneça a margem de segurança que esta capacidade terá ao compartilhar vários fluxos de fontes ABR. Este valor pode variar entre zero e PCR e pode assumir valores diferentes para direções diferentes de VCC's.

NH - Network Header, cabeçalho de rede. Cabeçalho que um segmento de dados proveniente da camada de transporte do modelo OSI da ISO recebe quando é admitido na camada de rede.

MMBP - Markov Modulated Bernoulli Process, processo de Markov modulado Bernoulli. É o processo de Markov com o tempo discretizado em intervalos fixos que, por conveniência, será o tempo t_s de serviço ATM e será denominado "slot de tempo".

NNI - Network-Network Irterfaces, interface rede-rede. Em uma rede ATM, define a interface entre comutadores ATM.

NPC - Network Parameter Control, parâmetro de controle da rede, é o mesmo que UPC mas implementado nas NNI's.

OAM - Operations And Maintenance, operação e manutenção. Células especiais que constantemente circulam por canais virtuais permanentes e exclusivos da rede ATM e que possuem a função de monitoração da performance da rede, detecção de defeitos e falhas, proteção do sistema e localização de falhas.

OSI - Open Systems Interconnection, interconexão de sistemas abertos. Interconexão de sistemas que obedecem às condições que qualificam um sistema aberto.

PCR - Peack Cell Rate, taxa de pico de célula. Um dos requisitos QoS e que indica a taxa máxima de células em uma conexão.

PD - Packetization Delay, atraso de empacotamento. Atraso devido a formação da célula ATM na UNI.

PEF - Processo Estatisticamente Fractal. Processo estocástico com características estatísticas como média, variância e autocorrelação invariantes com a escala de tempo ou com parâmetro de Hurst maior que 0.5.

PH - Presentation Header, cabeçalho de apresentação. Cabeçalho que um bloco de dados da camada de aplicação recebe quando é admitido na camada de apresentação do modelo OSI da ISO.

QoS - Quality of Service, qualidade de serviço. Conjunto de parâmetros de desempenho da rede ATM (CER, SECBR, CLR ou CLP(QoS), CMR, CTD, CDV e CTD médio, vide página 21).

QD - Queueing Delay, atraso de fila. É o componente variável do CTD e é provocado pela ação do gerenciamento sobre o controle de congestionamento de fila nos "buffers" da rede ATM.

RDSI - Rede Digital de Serviços Integrados . Padrão internacional de rede de tecnologia inteiramente digital e que suporta a integração de serviços diversos de comunicação, como voz, dados e imagem.

RDSI-FE - RDSI de Faixa Estreita. O mesmo que RDSI, mas com velocidades menores, não suportando o serviço de vídeo com qualidade.

RDSI-FL - RDSI de Faixa Larga. O mesmo que RDSI, mas com velocidades maiores, suportando o serviço de vídeo com qualidade.

RM - Resource Management, gerenciador de recursos. Função de gerenciamento da camada de controle dos planos de gerência de rede ATM. Aloca recursos (largura de banda) a nível de caminho virtual de acordo com informações trazidas pelas células OAM. Estes recursos serão distribuídos dentre os vários circuitos virtuais que fluem por este caminho virtual pelos respectivos CAC's.

RSVP - Resource ReSerVation Protocol, protocolo de reserva de recursos. Protocolo que reserva recursos na rede, de acordo com requerimentos de tráfego.

SAP - Service Access Point, ponto de acesso de serviço. É o ponto de ligação entre entidades ímpares ou de camadas adjacentes funcionais do modelo de rede em foco (arquitetura OSI da ISO, TCP/IP ou ATM).

SBBP - Switched Batch Bernoulli Process, processo de Bernoulli comutado em lote. É a superposição de MMBP's. Neste Processo, o tempo em um SBBP é dividido em "slots" de igual comprimento, como convém às redes ATM. As chegadas durante um "slot" ocorre como um processo de lote de células, em vez de apenas uma célula com o tamanho do lote obedecendo a uma distribuição de acordo com o K-estado da Cadeia de Markov. (vide página 46)

SBR - Sustainable Bit Rate, taxa de bit sustentável. O mesmo que VBR (vide VBR).

SCR - Sustainable Cell Rate, taxa de célula sustentável. É definido como o inverso do tempo de interchegadas ($1/T_s$) de duas células consecutivas e é um dos parâmetros determinantes do MBS para o GCRA.

SD - Switch Delay, atraso de comutação. Atraso da célula ATM devido aos componentes físicos de um comutador ATM.

SDH - Syncronous Digital Hierarchy, hierarquia digital síncrona. Padrão de transferência de modo síncrono, desenvolvido originalmente para utilização em troncos telefônicos à base de fibras ópticas.

SECBR - Severely-Errored Cell Block Ratio, Razão de blocos de células severamente erradas. É a razão entre a constatação de certa quantidade de blocos de células severamente erradas num intervalo de tempo e o total de blocos de células ATM transmitidas no mesmo intervalo de tempo. Como um bloco de células é uma seqüência de N células transmitidas consecutivamente em uma dada conexão, um bloco de células severamente errado ocorre quando mais de M células erradas, células perdidas ou células mini-inseridas são observadas no bloco de células recebidas. É imposta pelo provedor ao usuário e realizado o devido policiamento.

TASI -Time-Assigned Speech Interpolation, interpolação de voz de tempo concedido. Método de multiplexação analógica para canais de voz em que é aproveitado o intervalo médio de silêncio de uma conversação para a transmissão de outros canais.

TCP/IP - os dois principais protocolos da arquitetura INTERNET. O TCP, Transmission Protocol, e o IP, Internet Protocol. O conjunto dos dois protocolos estabelecem a conexão fim-a-fim dos usuários.

TD - Transmission Delay, atraso de transmissão. Independe do modo de transferência utilizado para a informação e depende da distância fim-a-fim do meio de transmissão, somadas as partes (valor típico de 4 e 5 μseg por Km)

TDM - Time Division Multiplex, multiplex por divisão de tempo. Método de acesso onde vários usuários compartilham o mesmo meio de transmissão em tempos diferentes.

TELNET - Protocolo de acesso remoto pertencente à arquitetura TCP/IP o qual disponibiliza a conexão remota de uma estação de trabalho a um servidor.

TH - Transport Header, cabeçalho de tranporte. Cabeçalho que um bloco de dados da camada de sessão do modelo OSI da ISO recebe quando é admitido na camada de transporte.

UBR - Unspecified Bit Rate, taxa de bit não especificada. Aplica-se às conexões em que os parâmetros de tráfego não são declarados formalmente e, portanto, sem garantias

QoS. O senso comum é que os próprios usuários implementem controles reativos por ação de protocolos de altas camadas

UNI - User-Network Interface, interface usuário-rede. A UNI possui as seguintes atribuições: conexão física entre usuário e comutador ; montagem e desmontagem do quadro ATM; sincronismo dos sinais na interface; emite células vazias para sincronização de mensagem quando não há o que transmitir; controle de erro no cabeçalho; sincronismo de célula; estruturação da célula; gerência das funções ATM; controle de tráfego e congestionamento (incluindo CAC que atua a partir da camada de gerência de rede nos planos de controle) e sinalização de rede.

UPC - Usage Parameter Control. É o agente de policiamento do gerenciador de recursos RM na UNI e verifica se a largura de banda alocado a um caminho virtual está sendo trafegada em conformidade.

VBR -Variable Bit Rate, taxa de bit variável É o tráfego que varia sua taxa de geração de células na camada ATM, conforme descrita por três parâmetros de tráfego: PCR, SCR e IBT (vide QoS), sendo que os dois últimos foram definidos em função do Algoritmo Genérico de Geração de Células-GCRA. O ITU-T e o ATM-FORUM distinguem VBR1, VBR2 e VBR3 dependendo do tráfego ser em tempo real (interativo) ou não, e das classes de QoS empregadas.

VCC - Virtual Channel Connection, conexão de canal virtual. Canal lógico para uma conexão ATM após o "setup"

VP - Virtual Path, caminho virtual. É a via onde ocorrerá o VPC ou a conexão de caminho virtual.

VPC - Virtual Path Connection, conexão de caminho virtual. É uma conexão que engloba e direciona vários canais virtuais na mesma via ou estabelece vários circuitos virtuais que possuem o mesmo destino num mesmo feixe de caminho virtual.

WAN - Wide Area Network, rede de longa distância. rede de computadores que cobre uma grande área geográfica (um país ou grupo de países)