

Este exemplar corresponde à redação final da
Tese/Dissertação devidamente corrigida e defendida
por: César Augusto Viana
Neto
e aprovada pela Banca Examinadora.
Campinas, 22 de dezembro de 1975
M. Mendes
COORDENADOR DE PÓS-GRADUAÇÃO
CPG-IC

**Multiplexação e Policiamento de Tráfego
Auto-Semelhante**
César Augusto Viana Neto
Dissertação de Mestrado

UNICAMP
BIBLIOTECA CENTRAL
SEÇÃO CIRCULANTE

Multiplexação e Policiamento de Tráfego Auto-Semelhante

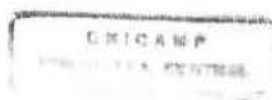
César Augusto Viana Neto¹

Dezembro de 1999

Banca Examinadora:

- Prof. Dr. Nelson Luis Saldanha da Fonseca
IC/UNICAMP (Presidente)
- Prof. Dr. Dalton Soares Arantes
FEEC/UNICAMP
- Prof. Dr. Célio Cardoso Guimarães
IC/UNICAMP
- Prof. Dr. Edmundo R. M. Madeira (Suplente)
IC/UNICAMP

¹Supported in part by CNPq, CAPES/PICDT



DADE	BC
CHAMADA:	unicamp
	634m
Es.	
	40131
	278/00
	D <input checked="" type="checkbox"/>
	R \$ 11,00
	28/01/00
CPD	

CM-00134451-8

**FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DO IMECC DA UNICAMP**

Viana Neto, César Augusto

V654m Multiplexação e policiamento de tráfego auto-semelhante / César Augusto Viana Neto -- Campinas, [S.P. :s.n.], 1999.

Orientador : Nelson Luis Saldanha da Fonseca

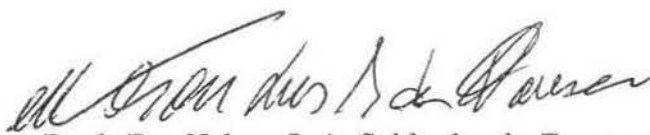
Dissertação (mestrado) - Universidade Estadual de Campinas, Instituto de Computação.

1. Redes de computadores. 2. Fractais. I. Fonseca, Nelson Luis Saldanha. II. Universidade Estadual de Campinas. Instituto de Computação. III. Título.

Multiplexação e Policiamento de Tráfego Auto-Semelhante

Este exemplar corresponde à redação final da
Dissertação devidamente corrigida e defendida
por César Augusto Viana Neto e aprovada pela
Banca Examinadora.

Campinas, 30 de novembro 1999.



Prof. Dr. Nelson Luis Saldanha da Fonseca
IC/UNICAMP (Presidente)

Dissertação apresentada ao Instituto de Com-
putação, UNICAMP, como requisito parcial para
a obtenção do título de Mestre em Ciência da
Computação.

TERMO DE APROVAÇÃO

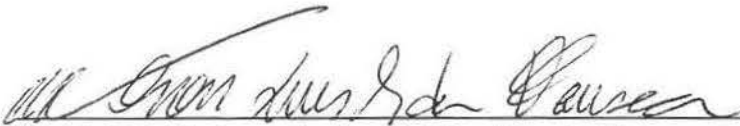
Tese defendida e aprovada em 19 de novembro de 1999, pela
Banca Examinadora composta pelos Professores Doutores:



Prof. Dr. Dalton Soares Arantes
FEEC - UNICAMP



Prof. Dr. Célio Cardoso Guimarães
IC - UNICAMP



Prof. Dr. Nelson Luís Saldanha da Fonseca
IC - UNICAMP

Resumo

Nessa dissertação avalia-se mecanismos para controle de tráfego auto-semelhante. Apresenta-se um conjunto de equações capazes de estimar os recursos necessários para fornecer Qualidade de Serviço a um agregado de fontes. Avalia-se o comportamento dessas equações em diferentes cenários e são enfatizadas as vantagens obtidas com a utilização do conceito de escala de tempo de interesse. Introduce-se também um mecanismo de policiamento de tráfego auto-semelhante, o Balde Furado Fractal, e avalia-se sua efetividade no policiamento de tais tráfegos.

Abstract

In this thesis we evaluate flow control mechanisms for self-similar traffic. We show a framework capable to compute resource requirements to provide Quality of Service for an aggregate of sources using the concepts of envelope process and time-scales of interest. We evaluate this framework with different traffic sets and show the obtained multiplexing gain. Moreover, we introduce the fractal Leaky Bucket, a policing mechanism for self-similar traffic.

Agradecimentos

Eu gostaria de agradecer a todas as pessoas que de alguma forma ajudaram na realização desse trabalho. Segue uma pequena lista de nomes, os esquecidos serão lembrados para sempre.

À minha família pelo seu amor irrestrito, sem vocês nada disso seria possível. A boa prosa, ao telefone, que reanimava. Vó, Mãe, Pai, Zaida, Dó, Boba, Raquel, Sara, Zé, tios, tias, primos, em fim, todos.

Ao Prof. Nelson pela amizade e orientação. Foi um período de muito aprendizado e que, com certeza, levarei comigo para sempre.

Ao Dário, ao Silvio e ao Cláudio pela convivência, paciência e grande lição de generosidade.

Ao Marcelo pela acolhida, dicas, indicações e pela conversa fiada.

Ao Arlindo e ao Pimentel, valeu rapaziada! a conversa num deixou de ser agradável.

E por último, porém não menos importante, aos “guardiões do laboratório”, muito bem representados pela pessoa do amigo Lídio, por tornarem essa passagem mais divertida.

Conteúdo

Resumo	v
Abstract	vi
Agradecimentos	vii
1 Introdução	1
2 Modelos Matemáticos	4
2.1 Modelos <i>On/Off</i>	4
2.2 Modelo Markoviano Modulado	5
2.3 Modelo Gaussiano Auto-regressivo	5
2.4 Modelos Auto-Semelhantes	6
2.5 Modelos Pseudo Auto-Semelhantes	8
3 Tráfego Auto-Semelhante	10
3.1 Definição	11
3.2 Propriedades	11
3.3 Movimento Browniano Fractal	14
3.4 Modelos de fonte <i>on/off</i> com caudas longas	15
3.5 Métodos para Estimativa do Parâmetro H	16
3.6 Gerador de Ruído Gaussiano	18
4 Modelos Limitantes de Tráfego	21
4.1 Limitante Determinístico Dependente do Intervalo - D-BIND	22
4.2 Processo Envelope Variante no Tempo - PEVT	23
4.3 Limitante de explosividade (σ, ρ)	24
4.4 Limitante de Explosividade Exponencial	25
4.5 Limitante Estocástico Dependente do Interval - LEDI	26
4.6 Taxa Envelope Mínima com relação a θ	27

4.7	Processo Envelope do Movimento Browniano Fractal	28
5	Gerenciamento de Tráfego em Redes ATM	32
5.1	Controle de Admissão de Conexões	32
5.1.1	Descritores de Tráfego	33
5.1.2	Abordagens de CAC	33
5.2	Mecanismos de Policiamento de Tráfego	37
5.3	Mecanismos de Descarte Seletivo	41
5.4	Controle de Fluxo ABR	45
5.4.1	Alocação <i>Max-Min</i>	46
6	Multiplexação Estatística de Fontes Auto-semelhantes	48
6.1	Escala de Tempo de Interesse	48
6.1.1	Tempo até a Fila Alcançar o Máximo	48
6.1.2	Escala de Tempo Crítica	51
6.1.3	Escala de Tempo Mínima para que Ocorra Perda de Células	52
6.2	Resultados Analíticos	53
6.3	Resultados Numéricos	54
6.4	Algoritmo para Controle de Admissão de Conexões	59
7	Policiamento de Fontes Auto-semelhantes	65
7.1	O Balde Furado	65
7.2	O Balde Furado Fractal	68
7.3	Resultados Numéricos	69
8	Conclusões	73
8.1	Trabalhos futuros	74
	Bibliografia	75

Lista de Tabelas

3.1	Estimativa do parâmetro H da amostra gerada.	20
6.1	Parâmetros de tráfego para a figura 6.5.a.	58
6.2	Parâmetros de tráfego para a figura 6.5.b.	58
6.3	Classes de conexão para as fontes de vídeo.	62

Lista de Figuras

3.1	Função de autocorrelação de um processo exatamente auto-semelhante de segunda-ordem com parâmetro Hurst $H = 1 - \beta/2$	11
3.2	Exemplos de estruturas de autocorrelação a) processo de dependência de curta duração b) processo de dependência de longa duração.	12
3.3	Autocorrelação do Gerador de Ruído Gaussiano Fractal e dados de Bellcore.	19
4.1	Comportamento do PEVT quando modela fluxo de vídeo	23
4.2	A acurácia do processo envelope mBf: $Y(\tau)$ (curva central), processo envelope mBf $H = 0.5$ (curva de inferior) e $H = 0.83$ (curva superior)	29
4.3	A precisão do processo envelope mBf de um processo mBf com $\bar{a} = 0.8$ $\sigma = 1.0$ e $\epsilon = 10^{-6}$ e diferentes parâmetros H	30
5.1	Probabilidade de Violação X Limites (de [JSS96] © IEEE)	41
5.2	Exemplos de políticas de organização de <i>buffer</i>	43
5.3	Um exemplo de perda de célula devido a diferentes distribuições de tamanho de fila produzidas pelas políticas de descarte UCPD e PCPD.	45
6.1	A precisão da probabilidade de transbordo prevista pelas equações 6.9/6.10 para uma única fonte em função do tamanho do <i>buffer</i>	55
6.2	Ganho com a multiplexação para fluxos com diferentes parâmetros Hurst.	55
6.3	Ganho com a multiplexação em função do parâmetro Hurst para diferentes números de fontes.	56
6.4	Região de admissão para classes de fontes heterogêneas.	57
6.5	Probabilidade de transbordo X Tamanho do buffer para fontes heterogêneas.	58
6.6	Comparação do número de fontes admitida da pelas equações 6.9/6.10 e o trabalho em [SM97] e [Kel96]. No gráfico a curva das equações 6.9/6.10 está sobreposta pela curva derivada a partir da equação 3.35 em [Kel96].	59
6.7	Avaliação do algoritmo de CAC para diferentes tempos médios entre chegadas de pedidos de conexão	63
6.8	Avaliação do algoritmo de CAC para diferentes tempos médios de duração de conexão	64

7.1	Tamanho do balde x a razão entre a taxa média de chegada e a taxa de geração de permissões.	67
7.2	O Balde Furado Fractal.	69
7.3	Uma comparação entre o BFF e o BF para fontes violando o contrato baseado numa seqüência de fluxo de vídeo do filme <i>STAR WARS</i>	70
7.4	Uma comparação entre a probabilidade de violação apresentada pelo BFF e a probabilidade de violação apresentada pelo BF.	71
7.5	Probabilidade de violação apresentada pelo BFF para fontes mal-comportadas.	72
7.6	Sensibilidade da Probabilidade de Violação do BFF para durações diferentes da janela de tempo.	72

Capítulo 1

Introdução

Com as Redes Digitais de Serviço Integrado de Banda Larga será possível disponibilizar um conjunto heterogêneo de serviços com diferentes requisitos de qualidade de serviços (QoS - *Quality of Service*). Em outras palavras, a transmissão de dados, som e imagem será feita por um mesmo canal de comunicação. O ATM (*Asynchronous Transfer Mode*) tem sido utilizado para implementar tais redes.

O desenvolvimento de modelos analíticos com o objetivo de avaliar novos mecanismos de controle de fluxo, tendo como base fontes de tráfego real, tem sido objeto de pesquisa nos últimos anos. Os primeiros estudos realizados para modelagem de tráfego consideraram que o processo de chegada era Markoviano. Os modelos derivados a partir desses estudos são atraentes devido a sua simplicidade de solução. O problema dessa abordagem é que a caracterização do processo de chegada não considera a existência de correlações. Mostrou-se em [Kle76] que isto pode superestimar o desempenho da rede. Os modelos Markov modulado, propostos em seguida, descrevem a correlação presente no tráfego. De fato, esses modelos representam de forma precisa diferentes tipos de tráfego multimídia, *i.e.*, voz [ea91, HL86] e fontes de vídeo [MAS⁺98]. Em [AMS82, Bou91, Fon94] investigou-se a solução para sistemas de filas alimentado por chegadas Markovianas moduladas. A solução destes pode ter um alto custo computacional quando o número de fontes aumenta. Aproximações foram desenvolvidas de forma a gerar resultados com razoável precisão e que tenham um menor custo computacional [AMS82, HL86].

Em 1993 Leland et. al. publicaram os resultados de um estudo que avaliou, de forma minuciosa, o tráfego coletado de uma rede local (LAN). Mostrou-se que esse tipo de tráfego apresenta estatísticas importantes que não são capturadas pelos modelos baseados em processos Markoviano. Além disso, mostrou-se que um processo auto-semelhante é capaz de caracterizar, de forma precisa, esse tráfego. Em seguida, outros estudos com outros tipos de tráfego, taxa de bit variável (*Variable Bit Rate-VBR*) vídeo, [GW94, BSTW95], tráfego de redes de longa distância [PF95] e o tráfego www, reforçaram esses achados.

Os mecanismos de controle de tráfego tais como controle de admissão de conexão, policiamento e descarte seletivo quando dimensionados tendo como base os modelos Markovianos, podem subutilizar os recursos da rede e penalizar injustamente os usuários se eles gerarem tráfego com características auto-semelhantes. Em [FF98] estudou-se os mecanismos de descarte seletivo tanto em nível de células quanto em nível de pacote, sob diversas políticas de gerenciamento do *buffer*, quando o sistema é alimentado por um processo auto-semelhante. Os achados demonstram resultados distintos daqueles encontrados em estudos realizados quando o tráfego não é auto-semelhante.

A utilização eficiente dos recursos de uma rede é o objetivo da multiplexação estatística de tráfego. O conceito de multiplexação estatística condiciona o compartilhamento dos recursos a um parâmetro probabilístico. O sistema fornecerá os recursos com uma probabilidade ϵ de que num instante t do tempo requisições não serão atendidas. Nesta dissertação avalia-se um conjunto de expressões, (capítulo 6), capazes de computar a banda passante requerida para se multiplexar o tráfego gerado por um agregado de fontes auto-semelhantes que operam com certo requisito de qualidade de serviço. Empregam-se os conceitos de modelos limitantes de tráfego, mais especificamente o processo envelope do movimento Browniano Fractal, e escala de tempo de interesse. Mostra-se que a utilização do conceito de escala de tempo de interesse, pelas expressões, propicia um uso mais racional dos recursos, se comparado com uma abordagem simplista que despreza esse conceito. Além disso, algumas avaliações comparam essas expressões com outras, que consideram a teoria dos grandes desvios, e verifica-se o mesmo desempenho. Como estudo de caso de utilização dessas expressões, implementa-se e avalia-se um mecanismo para controle de admissão de conexões.

Um mecanismo de policiamento é responsável pela verificação da conformidade do tráfego gerado por uma conexão, em relação aos descritores declarados no contrato de admissão. Em [MS95] mostrou-se que os mecanismos baseados em janelas e o Balde Furado (BF) não apresentam um bom desempenho quando as fontes monitoradas transmitem seus fluxos em rajadas. Nesta dissertação avalia-se o desempenho do mecanismo de policiamento BF, o mais popular na literatura, quando o fluxo a ser policiado é auto-semelhante. O comportamento do BF reforça os achados de outros estudos. Um novo mecanismo de policiamento, chamado Balde Furado Fractal (BFF) (capítulo 7), capaz de monitorar tráfego auto-semelhante, é proposto. Considera-se que o fluxo pode ser modelado por um processo envelope do movimento Browniano fractal, como proposto em [MSA]. Observa-se o comportamento do mecanismo quando o usuário opera em conformidade com o contrato estabelecido e quando este é violado. Os resultados das simulações mostram que o BFF apresenta um comportamento muito próximo daquele atribuído a um mecanismo de policiamento ideal.

Todos os programas utilizados para realizar as simulações descritas nesta dissertação

foram desenvolvidas em linguagem C padrão ANSI. O ambiente computacional utilizado é formado por máquinas IBM/RS6000 power2 de 120MHz com 512 MB de memória RAM, disponível no ambiente do CENAPAD-SP. Os tempos de simulação variaram de 1 hora até 384 horas.

Os demais capítulos têm a seguinte organização: no capítulo 2 alguns modelos matemáticos para caracterização de tráfego são apresentados e classificados em duas grandes classes, os auto-semelhantes e os não auto-semelhantes. No capítulo 3 discute-se o tráfego auto-semelhante. O Capítulo 4 descreve o conceito de modelos limitantes de tráfego. O conjunto de funções para o gerenciamento do tráfego em redes ATM é discutido no capítulo 5. São apresentadas abordagens para o controle de admissão de conexões, mecanismos de policiamento de tráfego, mecanismos de descarte seletivo e controle de fluxo de taxa mínima. No capítulo 6 são apresentados os estudos envolvendo multiplexação de tráfego auto-semelhante. Os resultados do policiamento de tráfego auto-semelhante são apresentados no capítulo 7. Finalmente, no capítulo 8 são apresentadas as conclusões e possíveis extensões deste trabalho.

Capítulo 2

Modelos Matemáticos

Caracterizar o tráfego real de uma rede de comunicação através de suas estatísticas é o objetivo dos modelos matemáticos descritos neste capítulo. De um modo geral, esses modelos podem ser agrupados em duas categorias, os que apresentam e os que não apresentam dependência de longa duração. Inicialmente, discutem-se os modelos com dependência de curta duração, a saber: O modelo *On/Off*, o processo Markoviano modulado e o processo Gaussiano Auto-regressivo. Em seguida, são apresentados os modelos com dependência de longa duração, a saber: O Modelo *On/Off* de cauda longa, o Movimento Browniano Fractal e os Modelos pseudo auto-semelhantes.

2.1 Modelos *On/Off*

De acordo com o modelo *On/Off*, o tráfego alterna entre dois estados: *On* e *Off*. Durante um período *On* o tráfego é gerado (transmitido) numa taxa constante r , e durante o período *Off* nenhum tráfego é gerado. A duração dos períodos *On* e *Off* são independentes. Os períodos tem uma distribuição comum. Sendo $E(On)$ e $E(Off)$ as médias dos períodos *On* e *Off*, respectivamente. Então, a probabilidade do processo estar no estado *On* é dada por:

$$P_{on} = \frac{E(on)}{E(On) + E(Off)}. \quad (2.1)$$

Seja X_t o processo que representa o estado do processo *On/Off* no tempo t com $X_t = 1$ e $X_t = 0$ denotando os estados *On* e *Off* no tempo t , respectivamente, e R_t o processo representando a taxa do processo *On/Off* no tempo t . Assim sendo, $R_t = rX_t$. Devido a variável randômica X ser uma variável randômica de Bernoulli com parâmetro P_{on} , a média de R_t é dada por $E(R_t) = rP_{on}$ e sua variância é definida como $Var(R_t) = r^2(P_{on})(1 - P_{on})$.

As duas últimas equações definem a média e variância de um processo *On/Off* e são de capital importância na modelagem de tráfego. Esses resultados podem ser aplicados a um grande número de processos, inclusive processo com dependência de longa duração, uma vez que não se assume nada a respeito das distribuições dos períodos *On* e *Off*.

2.2 Modelo Markoviano Modulado

Num processo Markoviano Modulado a taxa de chegada é modulada por uma cadeia de Markov embutida. A vantagem do processo Markoviano Modulado está na possibilidade da incorporação de diferentes padrões de correlação. Processos Markov Modulado são definidos em tempos contínuo e discreto. Eles também podem ser classificados baseado na possibilidade das chegadas ocorrerem em grupos. Os processos markov modulados mais usados são:

- Processo de Chegada Markoviano (*Markovian Arrival Process* - BMAP) - Tempos contínuos com chegadas em grupos;
- Processo Poisson Markov Modulado (*Markov Modulated Poisson Process* - MMPP) - Tempo contínuo com chegadas únicas;
- Processo de Chegada Markoviano em grupo em tempo discreto (*Discrete-Time Batch Markovian Arrival Process* - D-MAP) - Tempo discreto com chegadas em grupos;
- Processo de Bernoulli Modulado Markov (*Markov Modulated Bernoulli Process* - MMBP) - Tempo discreto com chegadas únicas;

2.3 Modelo Gaussiano Auto-regressivo

Um modelo de tráfego baseado num processo Gaussiano pode ser descrito como um processo de tráfego onde a quantidade de tráfego gerada durante um intervalo de tempo tem distribuição Gaussiana. Existem várias maneiras de representar um processo Gaussiano. O Auto-regressivo Gaussiano é uma delas. Nessa seção, define-se o processo em tempo discreto. Uma representação de tempo discreto pode ser justificada pela natureza do ATM onde o tempo é normalmente considerado discreto, com a menor unidade de tempo sendo o tempo necessário para transmitir uma célula.

Considera-se o tempo dividido em intervalos de tamanho fixo. Toma-se A_n como sendo uma variável randômica contínua representando a quantidade de trabalho entrando

no sistema durante o n -ésimo intervalo. A variável A_n pode representar o número de bits ou células ATM entrando num sistema durante o n -ésimo intervalo.

De acordo com um modelo Gaussiano Auto-regressivo assume-se que A_n , com $n = 1, 2, 3, \dots$, é um processo Auto-regressivo de ordem k , ou seja,

$$A_n = a_1 A_{n-1} + a_2 A_{n-2} + \dots + a_k A_{n-k} + b \tilde{U}_n \quad (2.2)$$

onde \tilde{U}_n é normalmente distribuído com média η e variância σ^2 , sendo a e b números reais com $|a| < 1$.

De forma a caracterizar o tráfego real, é necessário achar o melhor ajuste para os parâmetros a_1, \dots, a_k , e b , bem como η e σ . Por outro lado, em [AZ, AZ94a, AZ94b] mostrou-se que qualquer processo Gaussiano com três parâmetros estimam o desempenho de uma fila. Dessa forma, reduz-se a complexidade envolvida no ajuste de todos os parâmetros requerido para o caso de um modelo Auto-regressivo de ordem k e utiliza-se somente o processo Auto-regressivo de ordem 1. Nesse caso, assume-se que o processo A_n é dado por

$$A_n = a A_{n-1} + b \tilde{U}_n \quad (2.3)$$

com \tilde{U}_n normalmente distribuído com média η e variância σ^2 , sendo a e b números reais com $|a| < 1$. Esse modelo foi proposto em [MAS⁺98] para modelar um fluxo de tráfego VBR gerado por uma única fonte de vídeo-fone.

Os A_n s podem ser negativos com probabilidade positiva. Dessa forma, é limitada a aplicação desse modelo para o processo que descreve o tráfego real.

2.4 Modelos Auto-Semelhantes

Nessa seção, são apresentados três modelos de tráfego com dependência de longa duração. O primeiro baseia-se no modelo *On/Off*, o segundo no modelo *M/Pareto/ ∞* , e o terceiro é o processo movimento Browniano fractal (mBf, que é um processo Gaussiano com dependência de longa duração). Um modelo *On/Off* com o período *Off* (silêncio), e/ou o período *On* (ativo) com uma distribuição com cauda longa é um processo auto-semelhante. Um exemplo dessa distribuição é a de Pareto. Um número randômico X tem uma distribuição Pareto com parâmetro $\theta > 0$ e $1 < \gamma < 2$ se

$$P(X > t) = [\theta/(t + \theta)]^\gamma \quad t > 0. \quad (2.4)$$

Uma distribuição de cauda longa para o período *On* pode ser justificada pela transmissão de arquivos extremamente longos por uma fonte. Para o período *Off*, a justificativa está na ausência de transmissão por longos períodos. Para reduzir o número de parâmetros é conveniente usar a distribuição Pareto com $\theta = 1$, e considerar o parâmetro γ definido no intervalo aberto de $(1, 2)$. Nesse caso, a função de distribuição complementar é dada por:

$$P(X > t) = [(t + 1)]^{-\theta} \quad t > 0. \quad (2.5)$$

A média da distribuição Pareto é dada por

$$E\{X\} = \int_0^{\infty} x(x + 1)^{-\gamma} dx = 1/(\gamma - 1). \quad (2.6)$$

O processo *M/Pareto/∞* tem sido usado para modelar o tráfego gerado por um grande número de fontes alimentando um *buffer* num comutador. De acordo com esse modelo, o trabalho gerado ao longo do tempo é descrito por um processo de Poisson com parâmetro λ . O tamanho dessas porções de trabalho tem distribuição Pareto, e cada uma delas é transmitida a uma taxa fixa r . Em um certo tempo t , pode-se ter um número qualquer de fontes transmitindo com taxa r simultaneamente. De acordo com o modelo, uma nova fonte pode começar a transmissão enquanto outras estão ativas. Se n fontes estão simultaneamente ativas, a taxa total é igual a nr . O símbolo ∞ na notação do modelo é a representação do número ilimitado de recursos ativos transmitindo simultaneamente.

Considerando um processo *M/Pareto/∞* em tempo discreto, denota-se A_n como uma variável randômica contínua representando a quantidade de trabalho entrando no sistema durante o n -ésimo intervalo. Por conveniência, assume-se que a taxa r é a quantidade de tráfego transmitida por uma única fonte dentro de um intervalo de tempo caso a fonte permaneça ativa durante todo o intervalo.

Para calcular a média de A_n do processo *M/Pareto/∞*, considera-se a quantidade total de trabalho gerado num intervalo de tempo. Pode-se notar que a média da quantidade total de trabalho gerado num único intervalo de tempo é igual a média da quantidade de trabalho transmitido num único intervalo de tempo. Assim,

$$E[A_n] = \lambda r / (\gamma - 1). \quad (2.7)$$

Outro importante relacionamento para esse modelo é

$$\gamma = 3 - 2H \quad (2.8)$$

onde H é o parâmetro Hurst. Tendo-se essas duas últimas equações, pode-se ajustar a média do processo $E\{A_n\}$ e o parâmetro Hurst, para gerar tráfego baseado no modelo M/Pareto/ ∞ .

O movimento Browniano fractal (mBf) [MN68] é um processo Gaussiano. Isto é, a quantidade de trabalho gerado em qualquer intervalo de tempo tem distribuição Gaussiana. Esse processo é estacionário e exibe dependência de longa duração. Tomando-se Z_t como a quantidade de trabalho gerado dentro do intervalo $(0, t]$, então Z_t é Gaussiana, $Z_0 = 0$, $E\{Z_t\} = 0$, para todo t , $E\{Z_t^2\} = t^{2H}$ e Z_t tem amostras contínuas.

Para usar o mBf na modelagem de um processo A_n , toma-se o processo S_n definido anteriormente como a soma dos A_n s de 1 até n , e define-se:

$$S_n = nE\{A_n\} + \sigma Z_n \quad (2.9)$$

onde σ é o desvio padrão da quantidade de trabalho que chega em um intervalo. Se $Var\{Z_1\} = 1$, pela equação (2.9), $VAR\{S_1\} = \sigma^2$. Ou seja, σ^2 é a variância de A_n . Dessa maneira, o modelo de tráfego é definido por três parâmetros: (1) a média, (2) a variância (σ^2), e (3) o parâmetro Hurst do processo A_n . No capítulo 3 esse modelo é discutido com mais detalhes.

Apesar dos modelos de tráfego auto-semelhantes capturarem importantes estatísticas do tráfego real, não existe ainda, concordância a cerca de um modelo de tráfego que possa ser aplicado a todo tipo de fluxo multimídia. O estudo em [NAZH95] demonstra que o modelo de tráfego baseado no mBf tende a superestimar o desempenho de um multiplexador com tráfego real, enquanto a M/Pareto/ ∞ tende a subestimá-los.

2.5 Modelos Pseudo Auto-Semelhantes

Modelos de tráfego com dependência de curta duração tais como aqueles baseado em cadeia de Markov e Auto-regressivo Gaussiano são tratáveis analiticamente. Por outro lado, modelos auto-semelhantes capturam melhor estatísticas próprias do tráfego real. Essa situação sugere o uso de modelos que tenham dependência de curta duração, ou seja que tenham funções de autocorrelação exponencial e que sejam flexíveis a ponto de modelar dependência de longa duração em escalas de tempo de interesse. Nota-se que não é muito importante achar um modelo de tráfego que tenha a mesma função de autocorrelação como aquela apresentada pelo tráfego real em todas as escalas de tempo. Por exemplo, considerando-se o caso da aproximação para um *buffer* de tamanho zero, as correlações não têm qualquer efeito no desempenho da fila. Em um *buffer* "infinito", o intervalo no qual a correlação deve casar é maior. Da mesma forma, uma vez que o *buffer*

não é de tamanho “infinito”, existe um intervalo em que a função de autocorrelação tem relevante significância.

Em [AZN95] uma superposição de 2 processos Auto-regressivo de primeira ordem é usado para gerar tráfego pseudo auto-semelhante. Um desses dois processos tem seu coeficiente a (ver equação (2.3)) muito próximo de 1 (isto é, $a = 0.9999$). O outro processo é estável (ou seja, $a = 0.75$). Dessa maneira a inclinação da curva da variância em relação ao tempo na escala log-log com superposição de tráfego pode ser definida por qualquer número entre 1 e 2, para um conjunto arbitrário de períodos de tempo. Isso significa que se pode definir o intervalo dentro do qual a correlação tem cauda longa, tanto quanto necessário. Demonstra-se através de simulações de filas, [NAZH95], que se tal processo é posto numa fila com um único servidor, este apresenta desempenho igual quando a fila é submetida a tráfego baseado em mBf, com mesma média, variância e parâmetro Hurst. Idéia similar pode ser aplicada para gerar um processo pseudo auto-semelhante baseado em processos Markoviano modulado [And95].

Capítulo 3

Tráfego Auto-Semelhante

Os processos estocásticos auto-semelhantes são invariantes em sua estrutura de correlação em diferentes escalas de tempo. Ao se observar vários gráficos de um processo auto-semelhante em diferentes escalas de tempo eles irão parecer muito semelhantes.

Esses processos são de grande interesse na física, economia, geofísica, turbulência e na modelagem de fenômenos aleatórios com dependência de longa duração. Taqqu [TL86] fornece um guia bibliográfico comentado para muitas aplicações.

Os processos auto-semelhantes têm despertado, atualmente, especial interesse na modelagem de tráfego de redes dado que diversos estudos evidenciaram que o tráfego de redes tem o comportamento estatisticamente auto-semelhante (“fractal”). Entre esses estudos incluem-se uma análise de centenas de milhões de pacotes observados em várias redes locais *ethernet* nos laboratórios da Bellcore (*Bellcore Morristown Research and Engineering Center*) [LTWW93, WTE96], uma análise de milhões de dados de serviços de vídeo VBR (*Variable-Bit-Rate*) [BSTW95], o estudo de certas aplicações de redes geograficamente distribuídas tais como *telnet*, *ftp* [PF95] e *www* [CB95] e uma investigação detalhada de tráfego de dados coletados em nível de pacote de múltiplos núcleos de comutadores distribuídos pelo E.U.A. [KM94a]. Esse comportamento “fractal” é muito diferente dos tráfegos de telefonia e dos modelos até então utilizados para tráfego de pacotes (*e.g.* Poisson, Processos de Poisson Markoviano Modulado [HL86], modelos de fluxo de fluido [AMS82]).

Os principais conceitos relacionados com processos auto-semelhantes são apresentados neste capítulo. A seção 3.1 define processos auto-semelhantes. Na seção 3.2 comenta-se algumas propriedades dos processos auto-semelhantes. Em seguida na seção 3.3 o movimento Browniano fractal (mBf) é introduzido. O fenômeno de auto-semelhança em nível de fonte de dados é abordado na seção 3.4. Na seção 3.5 discute-se vários métodos para se estimar o grau de auto-semelhança de um processo. Mostra-se ainda um método para geração de dados auto-semelhantes baseado no mBf, na seção 3.6.

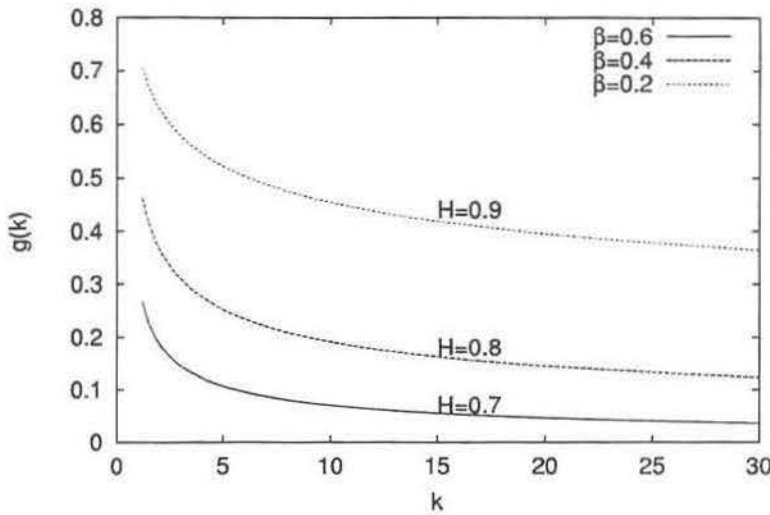


Figura 3.1: Função de autocorrelação de um processo exatamente auto-semelhante de segunda-ordem com parâmetro Hurst $H = 1 - \beta/2$.

3.1 Definição

Um processo X é chamado *exatamente auto-semelhante de segunda-ordem* com parâmetro $H = 1 - \beta/2$, $0 < \beta < 1$, se sua função de autocorrelação é dada por [TG97] [Cox84]:

$$r^{(m)}(k) = \frac{1}{2}[(k+1)^{2-\beta} - 2k^{2-\beta} + (k-1)^{2-\beta}] \triangleq g(k), \quad 0 < \beta < 1, \quad k = 1, 2, 3, \dots \quad (3.1)$$

e X é chamado *assintoticamente auto-semelhante de segunda-ordem* com parâmetro $H = 1 - \beta/2$, $0 < \beta < 1$, se para todo $k = 1, 2, 3, \dots$,

$$\lim_{m \rightarrow \infty} r^{(m)}(k) = \frac{1}{2}[(k+1)^{2-\beta} - 2k^{2-\beta} + (k-1)^{2-\beta}] \triangleq y(k). \quad (3.2)$$

A Figura 3.1 mostra a função de autocorrelação $g(k)$ (equação (3.1)) como função de k para três valores de β . Os valores de $\beta = 0.2, 0.4$ e 0.6 representam conforme a equação $H = 1 - \beta/2$, respectivamente $H = 0.7, 0.8$ e 0.9 .

3.2 Propriedades

A auto-semelhança é expressada, matematicamente, em um número de modos equivalentes:

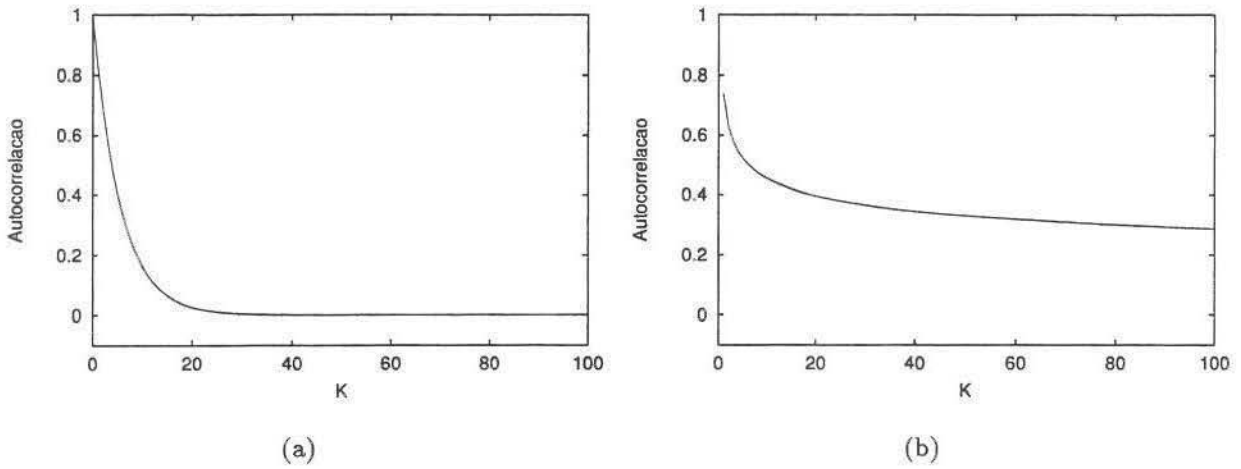


Figura 3.2: Exemplos de estruturas de autocorrelação a) processo de dependência de curta duração b) processo de dependência de longa duração.

- (i) a variância da média amostral decresce mais lentamente do que a variância do tamanho da amostra (*variância decaindo lentamente*), *i.e.*, $\text{var}(X^{(m)}) \sim am^{-\beta}$, com $m \rightarrow \infty$, com $0 < \beta < 1$;
- (ii) as autocorrelações decrescem hiperbolicamente ao invés de exponencialmente, implicando em uma função de autocorrelação não totalizável $\sum_k r(k) = \infty$, dependência de longa duração (*Long Range Dependence - LRD*), veja Figura 3.2b, e
- (iii) a densidade espectral (Transformada de Fourier) $f(\cdot)$ perto da origem comporta-se como $f(\lambda) \sim a\lambda^{-\gamma}$, para $\lambda \rightarrow 0$, $0 < \gamma < 1$ e $\gamma = 1 - \beta$.

Os processos auto-semelhantes se contrapõem a modelos de tráfego atualmente vigentes na literatura devido a estes últimos serem caracterizados por :

- (i) a variância da média amostral decresce com a variância do tamanho da amostra, *i.e.*, $\text{var}(X^{(m)}) \sim am^{-1}$, com $m \rightarrow \infty$;
- (ii) uma função de autocorrelação que decresce exponencialmente (*i.e.*, $r(k) \sim \rho^k$, $0 < \rho < 1$), implicando numa função de autocorrelação totalizável $\sum_k r(k) < \infty$, dependência de curta duração (*Short Range Dependence - SRD*), veja Figura 3.2a, e
- (iii) a densidade espectral limitada na origem.

A Figura 3.2 compara o comportamento da autocorrelação de um processo com dependência de curta duração (Figura 3.2a) com o comportamento de um processo com

dependência de longa duração (Figura 3.2b). Nota-se que no caso de dependência de curta duração têm-se valores expressivos de autocorrelação apenas para pequenos valores de k ($k < 10$), enquanto que para dependência de longa duração continua-se a ter valores expressivos de autocorrelação mesmo para valores maiores de k (veja também Figura 3.1).

A característica mais notável dos processos auto-semelhantes de segunda-ordem é que seus processos agregados $X^{(m)}$ possuem uma estrutura de correlação não degenerativa, com $m \rightarrow \infty$. A existência de uma estrutura de correlação não degenerativa para os processos $X^{(m)}$, com $m \rightarrow \infty$, está em completo contraste com os modelos típicos de tráfego de pacotes até então considerados na literatura, os quais têm processos agregados, $X^{(m)}$, que tendem para ruído puro de segunda-ordem, *i.e.*; para todo $k \geq 1$,

$$r^{(m)}(k) \rightarrow 0, \text{ com } m \rightarrow \infty, \text{ para } k \geq 1. \quad (3.3)$$

Uma das grandes vantagens dos processos auto-semelhantes é a parcimonidade do número de parâmetros para representar uma série temporal de centenas de milhares de observações. No caso de processos que não são auto-semelhantes, existe a necessidade de se ter um grande número de parâmetros para modelar longas séries. Em contraste, a componente de longa duração pode ser modelada com um único parâmetro quando se usa processos auto-semelhantes.

Dois processos estocásticos que produzem representações elegantes de fenômenos auto-semelhantes são: o ruído *Gaussiano fractal* (*fractional Gaussian noise*) e os *processos fractal auto-regressivo de médias móveis integradas* (*fractional autoregressive integrated moving-average processes* - (F-ARIMA)). O ruído Gaussiano fractal $X = (X_k : k \geq 0)$ com parâmetro $H \in (0, 1)$ foi introduzido em [MN68], e é um processo Gaussiano estacionário com média μ , variância σ^2 , e função de autocorrelação $r(k) = 1/2(|k + 1|^{2H} - |k|^{2H} + |k - 1|^{2H}), k > 0$. É possível mostrar que o ruído Gaussiano fractal é exatamente auto-semelhante de segunda-ordem com parâmetro de auto-semelhança H entre $1/2 < H < 1$. A série de ruídos Gaussianos fractais $X_i, i > 1$ é o incremento do movimento Browniano fractal (mBf), isto é $X_i = B_H(i + 1) - B_H(i), i \geq 1$ (mais detalhes sobre mBf na seção 3.3).

Os *processos fractais ARIMA* (p, d, q) são uma generalização da classe de modelos Box-Jenkins [BJ76], onde se permite que o parâmetro d assuma valores não inteiros. São processos assintoticamente auto-semelhantes de segunda-ordem com parâmetro auto-semelhante $d + 1/2$, onde $0 < d < 1/2$. O F-ARIMA foi usado para modelar tráfego de vídeo VBR [BSTW95, HDLK95b] e o desempenho de fila do F-ARIMA $(1, d, 0)$ foi analisado através de simulação para um servidor (*First Come First Serve* - FCFS)[eAM95].

3.3 Movimento Browniano Fractal

O movimento Browniano ordinário, $B(t)$, descreve o movimento de uma partícula em um líquido sujeito a colisões e outras forças [Pap91]. É uma função aleatória real com incrementos Gaussianos independentes tal que

$$\begin{aligned} E[B(t+s) - B(t)] &= 0 \\ \text{Var}[B(t+s) - B(t)] &= \sigma^2|s| \end{aligned}$$

Mandelbrot define um movimento Browniano fractal (mBf) como sendo o movimento médio de $dB(t)$ no qual incrementos que ultrapassam $B(t)$ são ponderados por $(t-s)^{H-1/2}$ [MN68].

Definição: Seja H tal que $0 < H < 1$. O mBf é definido pela integral de Weyl de $B(t)$.

$$B_H(t) = \frac{1}{\Gamma(H + 1/2)} \int_{-\infty}^{\circ} ((t-s)^{H-1/2} - (-s)^{H-1/2}) dB(s) + \int_0^t (t-s)^{H-1/2} dB(s).$$

Essa equação tende para o movimento Browniano ordinário se $H = 1/2$. Sua propriedade auto-semelhante é baseada no fato de que $B_H(\rho s)$ é idêntica em distribuição a $\rho^H * B_H(s)$. Os incrementos do mBf, Y_j , formam uma seqüência estacionária chamada *ruído Gaussiano fractal* (rGf):

$$Y_j = B_H(j+1) - B_H(j), j = \dots, -1, 0, 1, \dots$$

Esses incrementos não são independentes a não ser que se tenha movimento Browniano puro, *i.e.* $H = 1/2$. Além disso, a lei de Hurst estabelece que $\text{Var}[B_H(t+s) - B_H(t)] = \sigma s^{2H}$, *i.e.*, um modelo de chegada mBf é também capaz de capturar a alta *variabilidade* inerente exibida pelo tráfego de rede real.

O mBf normalizado com parâmetro $H \in [\frac{1}{2}, 1)$ é um processo estocástico $B(t), t \in (-\infty, +\infty)$, caracterizado pelas seguintes propriedades [Nor95]:

1. $B(t)$ tem incrementos estacionários;
2. $B(0) = 0$, e $EB(t) = 0$ para todo t ;
3. $EB(t)^2 = |t|^{2H}$ para todo t ;
4. $B(t)$ tem caminhos contínuos; e
5. $B(t)$ é Gaussiano, *i.e.*, todas suas distribuições conjuntas são Gaussianas.

Esse processo foi descoberto por Kolmogorov [Kol40], mas pouca atenção lhe foi dispensada até o artigo pioneiro de Mandelbrot e Van Ness [MN68].

3.4 Modelos de fonte *on/off* com caudas longas

Diversos estudos de medidas de tráfego de uma variedade de tipos de redes demonstraram que o tráfego atual de rede tem natureza auto-semelhante. Apesar dessas descobertas faltava compreender qual a explicação física para que o tráfego agregado observado tivesse esse comportamento. Willinger et. al [WTE96, WTSW97] fornecem uma explicação simples e plausível em termos de tráfego de fontes individuais que formam o tráfego agregado. Desenvolvendo uma abordagem proposta originalmente por Mandelbrot [Man69], eles mostram que a superposição de muitas fontes *on/off* independentes e identicamente distribuídas (*i.i.d*), cada uma das quais exibindo um fenômeno chamado de efeito Noé (*Noah effect*), resulta em um mBf, que é um processo exatamente auto-semelhante de segunda-ordem.

Uma fonte *on/off* idealizada é uma fonte onde períodos de geração de pacotes (períodos *on*) podem ser sucedidos por outros períodos *on*. Da mesma forma períodos sem geração de pacotes (períodos *off*) podem ser sucedidos por outros períodos *off*. Em um modelo *on/off* de alternância estrita, um período *on* não pode ser seguido de outro período *on*, e um período *off* por outro período *off*. Os modelos de fontes *on/off* de alternância estrita são comumente considerados na literatura de comunicações. Em [WTE96] foram demonstradas as condições para que fontes *on/off* idealizadas gerassem um processo auto-semelhante, e em [WTSW97] os mesmos resultados foram obtidos usando fontes de alternância estrita. Apesar dos resultados em [WTE96] serem essencialmente os mesmos apresentados em [WTSW97], a prova da convergência com fontes *on/off* com alternância estrita é muito mais complexa. Uma prova rigorosa é fornecida em [WTS97]. Heath et al. [HRS96] obtiveram independentemente resultados que são essencialmente idênticos aos apresentados em [WTS97].

O efeito Noé para um modelo de fonte *on/off* individual significa longos períodos *on* e longos períodos *off* com probabilidade não negligenciável. Em outras palavras, o efeito Noé implica que cada fonte *on/off* exibe características que encerram uma grande extensão de escalas de tempo. O efeito Noé é sinônimo de síndrome de variância infinita – a observação empírica de que muitos fenômenos naturais podem ser bem descritos usando distribuições com variância infinita (para referências veja [ST94, TL86, WTE96]). As distribuições com caudas longa, ou seja, distribuições que decaem hiperbolicamente,

$$P(U > u) \sim cu^{-\alpha} \quad \text{com } u \rightarrow \infty, 1 < \alpha < 2, \quad (3.4)$$

onde c é uma constante finita positiva, independente de u , apresentam a síndrome da variância infinita. Uma distribuição que satisfaz as condições (3.4) é a distribuição discreta Pareto (e.g., veja [ST94]). A distribuição Pareto com parâmetro de forma α e parâmetro de localização a tem a função de distribuição cumulativa dada por:

$$F(x) = P[X \leq x] = 1 - (a/x)^\alpha, a, \alpha \geq 0, x \geq a,$$

e função densidade de probabilidade correspondente dada por:

$$f(x) = \alpha a^\alpha x^{-\alpha-1}.$$

Se $\alpha \leq 2$, então a distribuição tem variância infinita, e se $\alpha \leq 1$, então a distribuição tem média infinita.

A distribuição Pareto tem sido usada para modelar distribuições de rendimentos, tamanhos de asteróides, ilhas, cidades e eventos de extinção [Kau93, Man63]. Constatou-se também que uma distribuição Pareto com $1.05 < \alpha < 1.25$ é um bom modelo para a quantidade de tempo de CPU consumida por um processo arbitrário [LO86]. Em telecomunicações, distribuições com caudas longa têm sido usadas para modelar tempo de duração de chamadas telefônicas e tamanhos de quadros em vídeo com taxa variável.

Demonstrou-se, também, uma relação simples entre o parâmetro α e o parâmetro H dado por $H = 3 - \alpha/2$, onde H mede o grau de auto-semelhança do fluxo de tráfego agregado [WTSW97].

3.5 Métodos para Estimativa do Parâmetro H

O grau de auto-semelhança (parâmetro H) de uma série temporal pode ser estimado através das seguintes ferramentas gráficas: (1) análise do domínio de tempo baseado em estatística R/S, (2) análise de variância de processos agregados $X^{(m)}$, e (3) análise baseada em periodograma no domínio da frequência. A seguir são descritos os métodos para estimativa do grau de auto-semelhança de um processo.

1. **estatística R/S** — Descrita em [MW69] consiste em plotar $\log(R(n)/S(n))$ versus $\log(n)$, onde $R(n)/S(n) = 1/S(n)[\max(0, W_1, W_2, \dots, W_n) - \min(0, W_1, W_2, \dots, W_n)]$, com $W_k = (X_1 + X_2 + \dots + X_k) - k\bar{X}(n)$ ($k \geq 1$). Quando H é bem definido, a plotagem típica de extensão de re-escala ajustada inicia com uma zona transiente de natureza de dependência de curta duração na amostra, mas eventualmente decresce e oscila em torno de uma reta com certo declive. Análise gráfica R/S é usada para determinar se tais comportamentos assintóticos parecem sustentados pelos dados. No caso afirmativo, um valor estimado de H , \hat{H} , é dado pelo declive assintótico, podendo variar entre 1/2 e 1.
2. **análise de variância de processos agregados X^m** — Observa-se que para processos auto-semelhantes de segunda-ordem, a variância dos processos agregados $X^{(m)}$, $m \geq 1$, decresce linearmente (para grandes m) em uma escala log-log. A

análise de variância de processos agregados é obtida pela plotagem $\log(\text{var}(X^{(m)}))$ contra $\log(m)$ (“tempo”) e pelo ajuste de uma linha quadrática mínima, ignorando-se os valores menores de m . Valores estimados do declive assintótico, $-\hat{\beta}$, entre -1 e 0 sugerem auto-semelhança, e uma estimativa de grau de auto-semelhança é dada por $\hat{H} = 1 - \hat{\beta}/2$.

3. **análise baseada em periodograma** — Consiste em plotar $\log(g(w))$ versus w , onde $g(w) = \frac{1}{2\pi n} |\sum_{j=1}^n X_j e^{ijw}|^2$, $0 \leq w \leq \pi$ de $X = (X_1, X_2, \dots, X_n)$, e aplicar um ajuste linear de mínimo-quadrado em uma seção do gráfico correspondente a w pequeno. O declive desse ajuste é igualado a $-(1 - \hat{\beta})$, de onde \hat{H} é computado com $1 - \hat{\beta}/2$.

A estatística R/S tem propriedades robustas, particularmente com respeito às distribuições com caudas longas. Contudo, ela tem a desvantagem de poder conduzir a estimativas imprecisas [Ber92]. A análise de variância de processos agregados é também um ponto de partida útil na análise de dados.

Análises de dados mais refinadas baseadas em métodos MLE (*Maximum Likelihood Estimate*) são usadas para estimar H quando X_t é um processo Gaussiano ou aproximadamente Gaussiano. Esses métodos e suas aproximações são baseados na densidade espectral de X_t e propriedades dos métodos MLE são discutidas por vários autores [Ber92, Dah89, Yaj85, FT86]. Um método específico que tem sido usado amplamente é aproximação de *Whittle* MLE. Em [TT] discute-se as qualidades dos estimadores *Whittle* “padrão”, *Whittle* “agregado” e *Whittle* “local”.

O estimador *Whittle* padrão é um método paramétrico que assume como conhecida a densidade da série espectral, com exceção de poucos parâmetros que são estimados. Essa suposição de parâmetros permite estimativas muito precisas quando a série sendo examinada ajusta-se exatamente ao modelo assumido. Se, por outro lado, a série não é especificada de forma correta no modelo o estimador paramétrico pode gerar valores incorretos. Um técnica adicional para obtenção de um valor estimado de H é fornecida pelo estimador *Whittle* agregado. Esse estimador pode ser usado se a série temporal é muito longa. A idéia é agregar os dados, o que cria uma nova série mais curta,

$$X_i^{(m)} := \frac{1}{m} \sum_{j=m(i-1)+1}^{mi} X_j.$$

A nova série aumenta o desvio padrão do estimador. Por outro lado, $m^{1-H} X_i^{(m)}$ converge fracamente para um ruído Gaussiano fractal. Se o nível de agregação m é bastante alto e a dependência de longa duração está presente, então a nova série estará perto de um ruído Gaussiano fractal. O estimador *Whittle* padrão é então usado com um modelo de rGf na série agregada.

Alternativamente, pode-se utilizar um estimador semi-paramétrico, *Whittle* local. Para estimar a dependência de longa duração, o estimador *Whittle* local geralmente assume que a autocorrelação para a série comporta-se como h^{2H-2} , onde h é o *lag*, e H é o parâmetro de dependência de longa duração. De forma equivalente ele pode assumir que a densidade espectral é proporcional a $|v|^{1-2H}$ para frequências pequenas de v . Esse estimador foi recentemente desenvolvido por [Rob94]. Ele é obtido através de um método não gráfico pela minimização de uma função similar aquela do *Whittle* padrão.

3.6 Gerador de Ruído Gaussiano

A solução para a distribuição de uma fila alimentada por um mBf não é conhecida até a presente data. Valores limites são derivados analiticamente através da Teoria dos Valores Extremos. Uma outra maneira para derivação dos resultados de uma fila com mBf é através de simulação. Existem vários algoritmos para geração de um tráfego mBf [Hos84],[MH78]. Novos métodos foram desenvolvidos recentemente. Huang [HDLK95a] propôs um método de simulação baseado em importância de amostragem (*importance samples*), Pruthi [PE95] usa mapas caóticos e Lau [LEWW95] usa um algoritmo de deslocamento de ponto médio aleatório. Na presente dissertação utiliza-se um algoritmo proposto originalmente por Mandelbrot e aperfeiçoado posteriormente por Chi [CNY73]. Esse algoritmo gera amostras de rGf de tempo discreto. Escolheu-se esse algoritmo levando em consideração os seguintes fatores: i) o algoritmo é baseado em uma soma de processos de Markov, portanto trabalhos futuros poderão investigar possíveis soluções numéricas para esse sistema, ii) as amostras mBf são compostas de uma soma de processos de alta e a baixa frequência, logo esse algoritmo é adequado para investigar as contribuições de cada componente da frequência para o desempenho do sistema e iii) é um algoritmo relativamente rápido (em aproximadamente 4 minutos gera uma amostra de um milhão de pontos em uma máquina Pentium 90).

Processos de ruído Gaussiano fractal discreto normalizados com média zero e variância unitária são processos aleatórios Gaussianos com incrementos Gaussianos e que tem autocovariância [ST94]:

$$C(s; H) = E[B_H(t+s+1) - B_H(t+s)][B_H(t+1) - B_H(t)] = 2^{-1}[|s+1|^{2H} - 2|s|^{2H} + |s-1|^{2H}].$$

O método proposto por Chi aproxima o rGf por uma soma de processos Markovianos de baixa e alta frequência. O processo agregado resultante tem uma função de autocovariância muito similar a [LTWW94]. Pode-se dizer que a autocorrelação de um processo de Markov simples decai exponencialmente enquanto que a autocorrelação de rGf decai linearmente. Portanto, é necessário somar vários processos de Markov independentes a fim

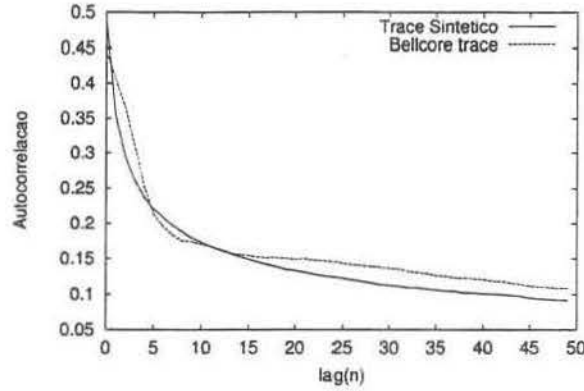


Figura 3.3: Autocorrelação do Gerador de Ruído Gaussiano Fractal e dados de Bellcore.

de se obter essa mesma estrutura de correlação. Um valor arbitrário de autocorrelação, aproximadamente $1/3$, é escolhido como valor de limite para separar as componentes de alta e baixa frequência. Considere:

$$X_f(t) = X_L(t) + X_H(t)$$

onde X_L e X_H denotam os termos de baixa e alta frequência respectivamente. X_f é o processo agregado. Para grandes lags (s) a covariância de $X_L(t)$, $C_L(s; H)$, é aproximadamente:

$$C_L(s; H) = H(2H - 1)s^{2H-2}.$$

Ao invés de aproximar o termo de baixa frequência por um processo de Markov simples, uma soma ponderada de N processos de Markov-Gauss padronizada é usada, isto é:

$$X_L(t) = \sum_{n=0}^N (W_n)^{1/2} M^{(n)}(t).$$

Os processos Markov-Gauss, $M^{(n)}(t)$, são assumidos dois-a-dois não correlacionados. O objetivo é calcular W_n tal que a covariância de $X_L(t)$ seja igual a $C_L(s; H)$. Dessa forma, após alguns cálculos obtém-se:

$$W_n = \frac{H(2H - 1)}{\Gamma(3 - 2H)} (B^{1-H} - B^{H-1}) B^{2(H-1)n}$$

para $0 \leq n \leq N$, onde $\Gamma(3 - 2H)$ é a função gama.

A discrepância devida à aproximação é: $D(s; H) = C(s; H) - C_L(s; H)$.

Para gerar a discrepância de alta frequência utiliza-se outro processo Markov-Gauss com variância $D(0; H)$ e atraso de covariância $D(1; H)$, tal que a covariância da soma do processo de baixa e alta frequência terá a estrutura auto-semelhante [LTWW94].

A aproximação do rGf é dada por:

$$X_f(t) = X_L(t) + X_H(t)$$

$$C_f(s) = C_L(s) + C_H(s).$$

A título de exemplo gerou-se dados com valores de H entre [0.7 - 0.85] tal que se tem os mesmos valores coletados em [LTWW94]. Mayor [May97] estimou o parâmetro *Hurst* (H) pelo uso da análise R/S.

A Tabela 3.1 mostra a avaliação através dos métodos análise R/S e análise de variância..

H	Análise R/S	Análise de Variância
0.7	0.69	0.71
0.75	0.73	0.76
0.8	0.78	0.81
0.85	0.83	0.84

Tabela 3.1: Estimativa do parâmetro H da amostra gerada.

Pelos dados mostrados na Tabela 3.1 verifica-se que esse algoritmo de geração tem uma boa precisão. Em [May97] computou-se também o coeficiente de correlação de 1.000.000 pontos da amostra de dados de Bellcore e comparou-se com as amostras do gerador (Figura 3.3). Verifica-se que i) os dados gerados tem uma estrutura com correlação de longa duração e ii) ambas as estruturas de correlação são muito semelhantes. É uma indicação clara de que esse gerador é uma boa caracterização de tráfego real de rede.

Capítulo 4

Modelos Limitantes de Tráfego

Uma descrição exata do tráfego gerado por uma fonte durante todo o intervalo de tempo é uma tarefa difícil e pode ter um custo computacional elevado. Modelos limitantes de tráfego são abordagens que tentam resolver este impasse, custo computacional/acurácia, pelo estabelecimento de limites para o tráfego gerado pela fonte durante intervalos de tempo. Os limites derivados a partir desses modelos podem ser utilizados pelos mecanismos de gerência da rede, mecanismos de CAC e policiamento [Cru91, E.K96, ZPT94]. Os modelos limitantes de tráfego podem ser agrupados em duas categorias, a saber: modelos limitantes determinísticos e modelos limitantes probabilísticos, ou simplesmente, modelos determinísticos e modelos probabilísticos.

Modelos determinísticos produzem limites superiores para chegadas de cada fonte. Essa abordagem possibilita uma análise de pior caso, de onde derivam-se limites absolutos para atraso e vazão. Mais especificamente, cada modelo determinístico define uma função $b(t)$ que limita o tráfego gerado pela fonte sobre intervalos de tempo de duração t . Denotando $A[t_1, t_2]$ como sendo as chegadas que ocorreram em uma conexão no intervalo $[t_1, t_2]$, a função $b(t)$ impõe que

$$A[s, s + t] \leq b(t), \quad \forall s, t > 0.$$

A vantagem dessa abordagem é que a caracterização do tráfego se dá com um reduzido número de parâmetros e que, geralmente, são de fácil obtenção. Essa característica, no entanto, pode produzir limites que não possuem a precisão “desejada”, ou seja, são limites “largos” [E.K96].

As propriedades estocásticas do tráfego gerado por uma fonte são utilizadas na abordagem probabilística para derivar os limites. Diferentemente da abordagem determinística, os limites derivados numa abordagem probabilística não são absolutos, isto é, não se considera exclusivamente o tráfego de pior caso. A vantagem dessa abordagem é a utilização menos conservadora dos recursos disponíveis na rede. A desvantagem reside na dificuldade de se derivar um único modelo que represente o tráfego com fidelidade. Os modelos

probabilísticos apresentam uma estrutura mais complexa, tanto quanto ao tipo quanto à quantidade de informação que é necessária conhecer [Sch96].

A limitação de tráfego utilizando modelos determinísticos e probabilísticos é muito discutida na literatura. Nas seções 4.1, 4.2 e 4.3, são apresentadas abordagens determinísticas para a limitação do tráfego. As abordagens probabilística são discutidas das seções 4.4 a 4.7.

4.1 Limitante Determinístico Dependente do Intervalo - D-BIND

O modelo D-BIND pode ser definido como a seguir. Uma fonte j , é descrita por uma curva $R_j(I)$ onde $R_j(I)$ é a taxa limitante sobre todo intervalo de tamanho I de forma que

$$A_j[t, t + I]/I \leq R_j(I) \quad \forall t, I > 0.$$

Assim, a fonte é deterministicamente limitada transmitindo no máximo $b_j(t) = tR_j(t)$ bits durante qualquer intervalo de tempo de duração t [KZ95].

Nessa caracterização dois pontos devem ser destacados, são eles: o comportamento da curva $R_j(I)$ e as informações limitadas que a fonte pode fornecer sobre si mesma. Na caracterização de uma dada fonte j , a tendência geral da curva $R_j(I)$ em pequenos intervalos I , é de $R_j(I)$ se aproximar da taxa de pico da fonte enquanto que para grandes intervalos, I , $R_j(I)$ irá se aproximar da taxa média da fonte.

Na prática, especifica-se o tráfego gerado pela fonte com um número pequeno de parâmetros. Por essa razão o modelo D-BIND é implementado com N pares de taxa-intervalo, i.e., $\{(R_n, I_n)\} | n = 1, 2, \dots, N\}$, distinguindo-se assim do modelo que consiste da caracterização de toda a curva $R(I)$.

No modelo D-BIND parametrizado a função que limita o tráfego requer interpolação entre os pares taxa-intervalo. Uma aproximação para o número de chegadas em qualquer intervalo de tempo de duração t é dada pela função de limitação do modelo com vários pares $(b_n/t_n, t_n)$, onde para pares taxa-intervalo (R_n, I_n) a função que limita o tráfego gerado pela fonte é dado por

$$b(t) = \frac{R_k I_k - R_{k-1} I_{k-1}}{I_k - I_{k-1}} (t - I_{k-1}) + R_{k-1} I_{k-1}, \quad I_{k-1} \leq t \leq I_k$$

assumindo $R_0 = I_0 = 0$

A precisão com que a função $b(t)$ descreve o tráfego de uma fonte está diretamente relacionada com o número de pares taxa-intervalo (R_k, t_k) , quanto maior o número de pares maior será a precisão de $b(t)$.

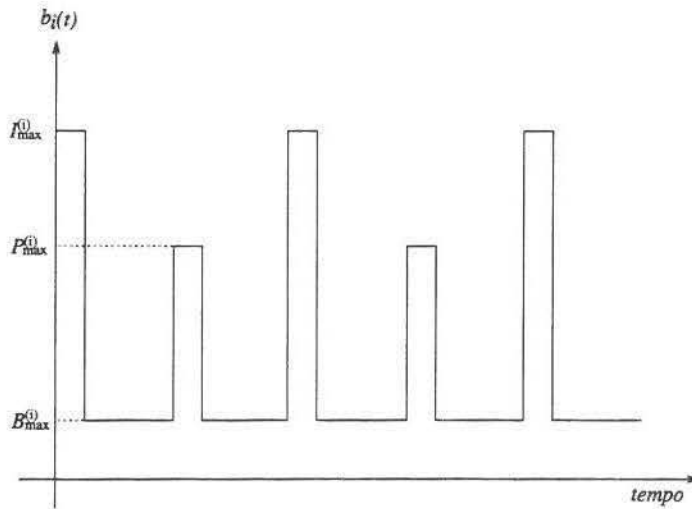


Figura 4.1: Comportamento do PEVT quando modela fluxo de vídeo

4.2 Processo Envelope Variante no Tempo - PEVT

Um codificador MPEG padrão gera três tipos de quadros: *Intra-coded (I)*, *Predictive (P)*, e *Bidirectional (B)*. Em geral, quadros *I* são maiores que quadros *P* que por sua vez são maiores que quadros *B*. Para simplificar o projeto do *hardware*, muitos codificadores MPEG usam um padrão para agrupar figuras (*Group-of-Picture - GOP*) quando comprimem uma seqüência de vídeo. Além disso, esses padrões são freqüentemente “regulares” no sentido de que o número de quadros *B* sucessivos entre dois quadro de referência é constante. Um padrão GOP regular é especificado por dois parâmetros: N (distância entre quadros *I*) e M (distância entre quadros *P*) com N múltiplo de M . O PEVT considera que o fluxo gerado pela fonte é GOP fixo e regular, e usa isso para alocar recursos.

Definição: Para o i -ésimo fluxo de vídeo, S_i , o processo envelope é a função periódica variante no tempo $\bar{b}_i(t)$ que é parametrizada por uma 5-tupla $E_i = (I_{max}^{(i)}, P_{max}^{(i)}, B_{max}^{(i)}, N^{(i)}, M^{(i)})$. Onde, $I_{max}^{(i)}$ é o maior frame na seqüência, $P_{max}^{(i)}$ é o maior frame considerando-se os quadros *P* e *B*, e $B_{max}^{(i)}$ é o maior quadro *B*. Os dois últimos parâmetros descrevem o padrão do GOP de S_i [KT97]. Por definição $I_{max}^{(i)} \geq P_{max}^{(i)} > B_{max}^{(i)}$. A figura 4.1 mostra como a função limitante $\bar{b}_i(t)$ se comporta quando modela um fluxo de vídeo utilizando os parâmetros $M = 6$ e $N = 2$.

Esse modelo é similar ao modelo D-BIND. Contudo, o modelo D-BIND fornece um limite invariante no tempo para o fluxo acumulado gerado pela fonte, que é diferente de um limite variante no tempo para a taxa de chegada do fluxo gerado pela fonte.

4.3 Limitante de explosividade (σ, ρ)

Representa-se o fluxo pela função não negativa R como segue: Para qualquer $y \geq x$ $\int_x^y R$ é a quantidade de dados num fluxo que é transmitida durante um intervalo $[x, y]$. Assim, diz-se que, $R(t)$ representa a taxa instantânea do tráfego que segue numa linha no tempo t . Diz-se que R é a função taxa de fluxo.

A função taxa de fluxo do tráfego, R , pode assumir somente dois valores, a saber: 0 ou a taxa de transmissão da linha. Em outras palavras, a linha está sendo utilizada em sua capacidade total ou está livre.

O limitante de explosividade (σ, ρ) é então definido como: dado um $\sigma \geq 0$ e $\rho \geq 0$, estabelece-se que $R \sim (\sigma, \rho)$ se e somente se para todo x, y com $y \geq x$ tem-se que:

$$\int_x^y R \leq \sigma + \rho(y - x).$$

Assim, se $R \sim (\sigma, \rho)$, existe um limite superior para a quantidade de tráfego contida em qualquer intervalo que é igual a uma constante σ mais uma quantidade proporcional ao tamanho do intervalo. A constante de proporcionalidade ρ determina um limite superior para a taxa média do fluxo de tráfego. Para um valor fixo de ρ , o termo σ permite certa explosividade, um σ maior é sinônimo de uma maior explosividade.

Em [Cru91] o modelo é discutido em detalhes. Limites são derivados para o atraso sofrido pelo tráfego de entrada sujeito ao modelo, tendo como base elementos isolados da rede, além disso, avalia-se o comportamento do tráfego dentro da rede. Um resumo dessas avaliações é apresentado a seguir:

- Se todos os tráfegos de entrada para um elemento isolado da rede (servidor) satisfizerem o limitante de explosividade, então o tráfego de saída tem o mesmo comportamento para os elementos analisados, não necessariamente com os mesmos parâmetros. Essa afirmação é demonstrada para uma variedade de elementos, sob diversas políticas de atendimento.
- No caso acima o atraso sofrido por cada byte é limitado superiormente por uma constante D , que dependem dos parâmetros do tráfego e da natureza do elemento examinado.
- Se todo o tráfego que entra na rede satisfizer o limitante de explosividade, e se seus parâmetros forem suficientemente pequenos, então todos os fluxos de tráfego dentro da rede também satisfazem algum limitante de explosividade.

4.4 Limitante de Explosividade Exponencial

Nessa abordagem, assume-se que a distribuição do tamanho da explosividade decai exponencialmente ao invés de considerar o tráfego com explosividade limitada. Procuram-se limites superiores nas funções de distribuição de probabilidade que descrevem a operação da rede e suas qualidades de serviço.

Um fluxo de tráfego que segue pelos canais de comunicação é caracterizado pela sua taxa de tráfego $R(t)$. Essa taxa é um processo estocástico que expressa a intensidade instantânea do fluxo de dados no canal. A quantidade de dados seguindo nesse fluxo de qualquer instante de tempo t_1 a qualquer instante (posterior) t_2 é exatamente $\int_{t_1}^{t_2} R(t)dt$. Se o fluxo segue numa linha com capacidade finita C , então para qualquer t essa taxa deve satisfazer a desigualdade $R(t) \leq C$.

Definição 1. Tome $W(t)$ com sendo qualquer processo estocástico. $W(t)$ é exponencialmente Limitado (*Exponentially Bounded* - E.B) com parâmetros A, α (ou (A, α) -E.B) se ele satisfaz [YS93]:

$$\Pr\{W(t) \geq \sigma\} \leq Ae^{-\alpha\sigma}$$

para todo $\sigma \geq 0$ e todo $t \geq 0$.

Definição 2: Toma-se $R(t)$ como sendo uma taxa de tráfego. $R(t)$ tem Explosividade Limitada Exponencialmente (*Exponentially Bounded Burstiness* - E.B.B) com parâmetros ρ, A, α (ou $R(t)$ é (ρ, A, α) -E.B.B) se ela satisfaz:

$$\Pr\left\{\int_s^t R(u)du \geq \rho(t-s) + \sigma\right\} \leq Ae^{-\alpha\sigma}$$

para todo $\sigma \geq 0$ e todo $0 \leq s < t$

Neste caso diz-se que $R(t)$ é E.B.B, e o processo é referenciado como um processo E.B.B.

São propriedades básicas dessa caracterização:

- 1) Para uma taxa de tráfego $R(t)$ caracterizada como (ρ, A, α) -E.B.B, com ρ definido como a taxa superior de $R(t)$, segue que para $\nu > \rho$, $R(t)$ também é E.B.B com taxa superior definida por ν .
- 2) Considere um sistema conservador de trabalho e que transmite seu fluxo a uma taxa ρ . Supondo que o sistema é alimentado por um fluxo com taxa de tráfego $R(t)$, e $W(t)$ representa a quantidade de dados armazenada no sistema no tempo t , tem-se então que:
 - i) Se $W(t)$ é E.B então $R(t)$ é E.B.B, com uma taxa superior ρ .
 - ii) Se $R(t)$ é E.B.B com taxa superior $\rho - \epsilon$ para algum $\epsilon > 0$ então $W(t)$ é E.B.

Em [YS93] encontram-se a prova matemática para as propriedades citadas anteriormente e uma análise de elementos da rede quando sujeitos a processos E.B.B.

4.5 Limitante Estocástico Dependente do Interval - LEDI

Nessa abordagem uma família de variáveis randômicas é definida tal que, um limite estocástico para o número de bytes enviados durante intervalos de tamanhos diferentes t seja estabelecido. As variáveis randômicas são funções explícitas de t .

O modelo baseia-se na idéia proposta em [J.K92], no qual uma fonte é caracterizada por uma família de 2-tupla $\{(R_{t_1}, t_1), (R_{t_2}, t_2), (R_{t_3}, t_3), \dots\}$ com R_{t_i} representando uma variável randômica que é estocasticamente maior que o número de bytes gerado pela fonte durante qualquer intervalo de tamanho t_i . Uma variável randômica X é dita estocasticamente maior que uma variável randômica Y (denota-se por $X \succeq_{st} Y$) se e somente se $Prob(X > x) \geq Prob(Y > x)$ para todo x . Ao invés de modelar de forma exata o processo de chegada da fonte, o modelo limita estocasticamente o número de bytes transmitido em intervalos de diferentes tamanhos.

A extensão proposta a esse modelo consiste na utilização dos componentes do modelo determinístico $(X_{mim}, X_{ave}, I, S_{max})$, descrito em [FV90], para derivar o limite R_{t_i} no intervalo t_i . A utilização do modelo determinístico torna a derivação dos limites função explícita do tamanho do intervalo.

São dois os requisitos gerais para o modelo LEDI:

$$R_t + R_s \succeq_{st} R_{t+s} \quad (4.1)$$

$$\frac{E(R_t)}{t} \leq \frac{E(R_s)}{s} \quad se \quad t > s. \quad (4.2)$$

A primeira propriedade é subaditiva estocástica. Essa propriedade define que a soma das variáveis randômicas R_t e R_s que limita os intervalos de tempo t e s , respectivamente, é estocasticamente maior que a variável randômica que limita o intervalo $t + s$.

A segunda propriedade estabelece que a média limitando a taxa sobre intervalos de tempo menor é maior que a média limitando a taxa sobre intervalos de tempo maior.

Em [ZK94] encontra-se a descrição completa do modelo com a derivação de exemplos de família de variáveis randômicas utilizada na modelagem do processo de entrada.

4.6 Taxa Envelope Mínima com relação a θ

O conceito de taxa envelope mínima com relação a θ é utilizada como ferramenta para computar limites em sistemas de filas estocásticos. Consideram-se limites para funções de geração de momento (*moment generating functions - mgf*). Diz-se que uma variável randômica X é limitada exponencialmente com relação a θ ($0 < \theta < \infty$) se o θ -norm da $\exp(X)$ é finito, ou seja, existe uma constante $d < \infty$ tal que

$$(Ee^{\theta X})^{1/\theta} \leq d.$$

Assim, tem-se do limite de Chernoff que

$$\Pr(X > x) \leq d^\theta e^{-\theta x} \quad \forall x$$

o que fornece um limite para a distribuição da cauda de X .

Considera-se o processo envelope (*envelope process - EP*), como descrito em [Cru91], do processo de entrada em relação a θ , mostra-se que se o processo de entrada em uma única fila tem EP linear com taxa menor que sua capacidade, C , e a fila está operando sobre uma política conservadora de trabalho, então i) o tamanho da fila é limitado exponencialmente com relação a θ , ii) existe um EP linear do processo de saída que pode ser representado como uma função do EP linear do processo de entrada, e iii) o atraso é limitado exponencialmente com relação a θC se a política de escalonamento é FCFS (*First Come First Serve*).

A taxa envelope mínima com relação a θ (*Minimum Envelope Rate - MER*) é definida como o ínfimo das taxas da classe de EP linear. Uma condição suficiente para o tamanho da fila ser limitado exponencialmente com relação a θ é que o MER do processo de entrada seja menor que a capacidade. Por outro lado, se o MER é maior que a capacidade, então o tamanho da fila não pode ser limitado exponencialmente com relação a θ .

Considera-se uma seqüência de variáveis randômicas não negativas $\{a(t), t = 0, 1, 2, \dots\}$. Seja $A(t_1, t_2) = \sum_{t=t_1}^{t_2-1} a(t)$. Utilizando-se a definição de processo envelope determinístico, o seguinte limitante para $a(t)$ é derivado:

$$\frac{1}{\theta} \log Ee^{\theta A(t_1, t_2)} \leq \hat{A}(\theta, t_2 - t_1) \quad \forall t_1 \leq t_2.$$

O processo $\hat{A}(\theta, t)$ será chamado de processo envelope de $a(t)$ com relação a θ (EP). Claramente, o processo envelope mínimo com relação a θ (MEP) é:

$$A^*(\theta, t) = \sup_{s \geq 0} \frac{1}{\theta} \log Ee^{\theta A(s, s+t)}.$$

A taxa envelope mínima com relação a θ é definida como:

$$a^*(\theta) = \limsup_{t \rightarrow \infty} \frac{A^*(\theta, t)}{t}$$

pode-se ver o MER considerando a família de EP linear

$$\Psi_\theta = \{\hat{a}(\theta) : A^*(\theta, t) \leq \hat{a}(\theta)t + \hat{\sigma}(\theta)\} \text{ para alguma constante não negativa } \hat{\sigma}(\theta)$$

onde $\hat{a}(\theta)$ é a taxa de um EP linear. Observa-se que $\hat{\sigma}(\theta)$ é constante em t , entretanto é uma função de θ . Ver-se que para cada θ fixo

$$a^*(\theta) = \inf[\hat{a}(\theta) : \hat{a}(\theta) \in \Psi_\theta].$$

Baseado na noção de MER com relação a θ desenvolve-se um conjunto de regras para as operações na rede tais como superposição, relação entrada-saída de uma fila única, e roteamento. Esse conjunto de regras fornece um método para responder dois tipos de problemas de instabilidade de redes de filas: i) condições para redes de filas finitas e com atraso máximo limitado, e ii) condições para rede de fila em que a distribuição do tamanho da fila tem cauda exponencial com taxa θ [Cha94].

4.7 Processo Envelope do Movimento Browniano Fractal

Um processo movimento Browniano (mB) $A(t)$ com média \bar{a} e variância σ^2 , tem processo envelope $\hat{A}(t)$ definido por [Pap91]:

$$\hat{A}(t) \stackrel{\text{def}}{=} \bar{a}t + \kappa\sqrt{\sigma^2 t} = \bar{a}t + \kappa\sigma t^{1/2}.$$

O parâmetro κ expressa a probabilidade de $A(t)$ exceder $\hat{A}(t)$ no tempo t . Sendo $A(t)$ um processo mB segue que:

$$\Pr\left(\frac{A(t) - \bar{a}t}{\sigma t^{1/2}} > \kappa\right) = \Phi(\kappa)$$

onde $\Phi(y)$ é a função de distribuição residual da distribuição Gaussiana padrão. Usando a aproximação,

$$\Phi(y) \approx (2\Pi)^{-1/2}(1+y)^{-1}\exp(-y^2/2) \approx \exp(-y^2/2) \quad (4.3)$$

obtem-se κ tal que $\Phi(\kappa) \leq \epsilon$, ou seja, κ é dado por $\kappa = \sqrt{-2\ln \epsilon}$. Assim sendo, pode-se afirmar que $\Pr(A(t) > \hat{A}(t)) \approx \epsilon$, onde $\kappa = \sqrt{-2\ln \epsilon}$. Este raciocínio pode ser estendido para processos com dependência de longa duração. Em um processo movimento Browniano fractal (mBf) com média \bar{a} , a variância dos incrementos é dada por

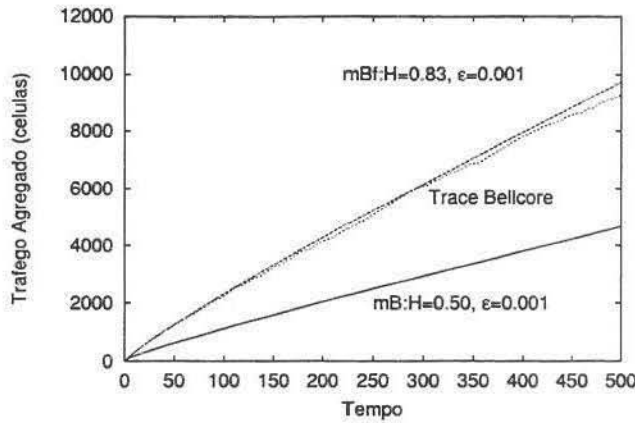


Figura 4.2: A acurácia do processo envelope mBf: $Y(\tau)$ (curva central), processo envelope mBf $H = 0.5$ (curva de inferior) e $H = 0.83$ (curva superior)

$\text{Var}[A_H(t+s) - A_H(t)] = \sigma^2 s^{2H}$, $H \in [1/2, 1)$ é o parâmetro Hurst e κ expressa a probabilidade de $A_H(t)$ exceder $\hat{A}_H(t)$. Define-se o processo envelope mBf:

$$\hat{A}_H(t) \stackrel{\text{def}}{=} \bar{a}t + \kappa\sqrt{\sigma^2 t^{2H}} = \bar{a}t + \kappa\sigma t^H. \quad (4.4)$$

Uma vez que o processo mBf exhibe LRD, se $A_H(t)$ exceder $\hat{A}_H(t)$ no tempo t essa situação permanecerá por um longo período de tempo.

Propõe-se um método muito simples para computar os parâmetros do processo envelope para esse trace: computa-se o processo envelope ótimo do trace. A vantagem dessa abordagem está na estimativa do parâmetro Hurst do trace que não precisa ser exata. Assume-se que o tráfego de entrada é caracterizado por um trace com N pontos definido por $A(t)$, onde $A(t)$ representa o número acumulado de células que chegam até o tempo t , $t \in [1, 2, \dots, N]$. O processo envelope ótimo para esse trace é definido por $Y(t-s) = \max_{s < t} (A(t) - A(s))$. Considera-se que o processo é estacionário de forma que $Y(\tau)$, $\tau = t - s$ define o número máximo de células que chegam num intervalo de tempo τ . Desta forma, escolhe-se os parâmetros do processo envelope mBf de forma que se tenha o mesmo comportamento de $Y(\cdot)$.

Comparou-se a representação do processo envelope para o trace extraído de uma rede local nos laboratórios bellcore. Desse trace, computou-se a taxa média de chegada e a variância, substituindo \bar{a} e σ^2 na equação 4.4. Computou-se o processo envelope ótimo, isto é, $Y(\cdot)$, e escolheu-se H de forma que $\hat{A}_H(\cdot)$ esteja de acordo com o comportamento de $Y(\cdot)$. Na Figura 4.2, a curva superior corresponde ao processo envelope mBf com $\epsilon = 10^{-3}$. A curva inferior representa o processo envelope do movimento Browniano com o mesmo ϵ . A curva do meio corresponde a $Y(\tau)$. Pode-se ver que o processo envelope mBf

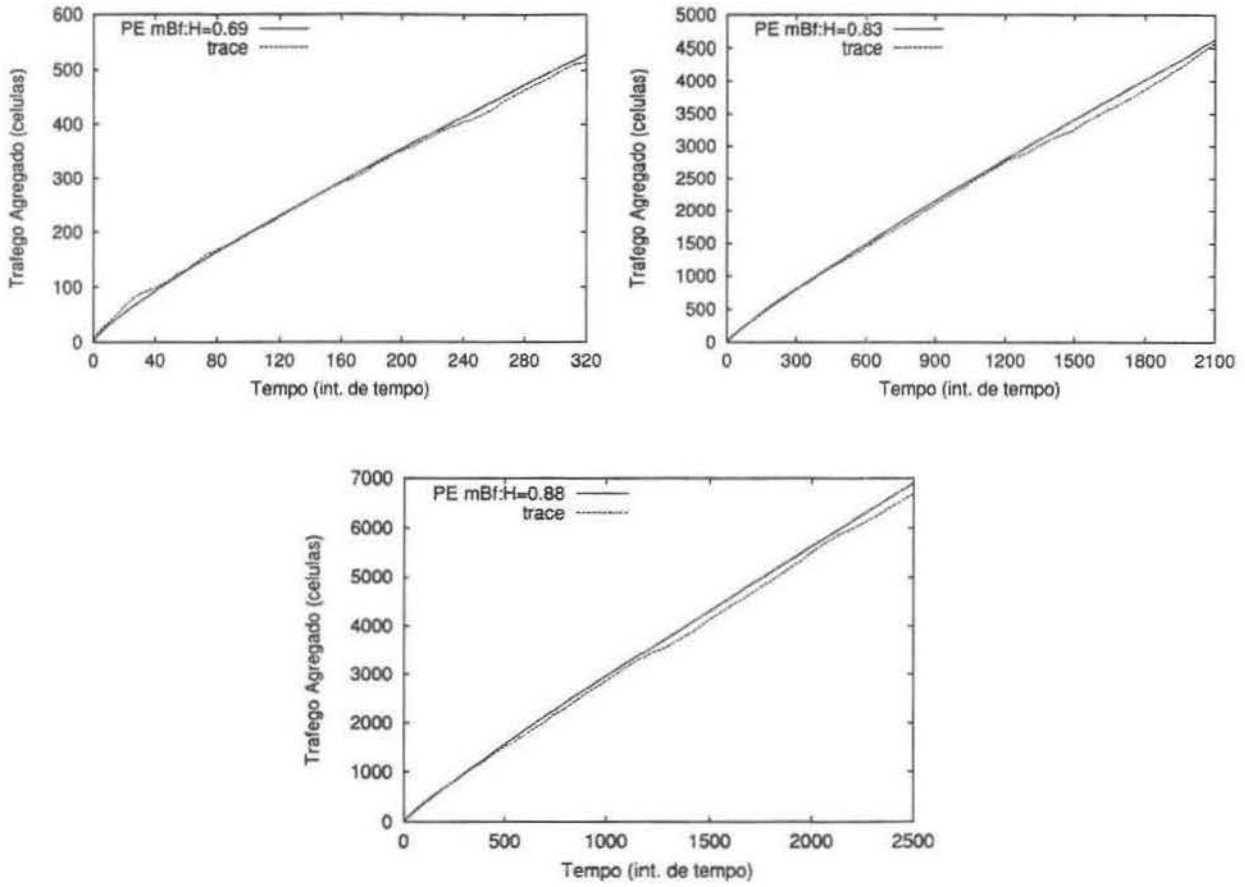


Figura 4.3: A precisão do processo envelope mBf de um processo mBf com $\bar{a} = 0.8$ $\sigma = 1.0$ e $\epsilon = 10^{-6}$ e diferentes parâmetros H .

tem um comportamento bem próximo do trace com LRD. Além disso, nota-se também que o processo envelope do movimento Browniano é incapaz de limitar o comportamento da fonte LRD mesmo se a ϵ for atribuído um valor menor.

Validou-se extensivamente a eficiência do processo envelope mBf utilizando traces sintéticos gerados pelo procedimento descrito por Mandelbort [Man69]. Em todos os traces usados nos experimentos de simulação, testes estatísticos foram realizados para verificar se a média, variância e o parâmetro Hurst do trace estavam de acordo com os valores especificados. Verificou-se a precisão do processo envelope variando-se os parâmetros de tráfego nos seguintes intervalos: $\bar{a} \in [0.5, 1.0]$, $\sigma^2 \in [0.05\bar{a}, 0.7\bar{a}]$, $H \in [0.5, 1.0]$, $\epsilon \in [10^{-3}, 10^{-9}]$ onde a média e a variância são normalizadas pela capacidade do canal. Os resultados indicam que o processo envelope mBf é um limite superior muito próximo para um processo mBf. Uma vez que, ele é preciso em todos os intervalos mencionados. A

Figura 4.3 ilustra a precisão do processo envelope mBf para um processo com $\bar{a} = 0.8$, $\sigma^2 = 1.0$, $\epsilon = 10^{-6}$ e diferentes valores para H .

O processo envelope mBf apresenta várias vantagens :

- Ele é parcimonioso, i.e., com apenas três parâmetros faz-se a caracterização da fonte.
- Ele pode representar SRD e LRD, i.e., a fonte não precisa necessariamente apresentar LRD. Sendo necessário apenas que os parâmetros do processo envelope mBf sejam escolhidos de forma a se ajustar com o processo envelope ótimo da fonte na escala de tempo de interesse que será discutida na seção 6.1.
- Os parâmetros de entrada \bar{a} , σ e H podem ser fornecidos pela fonte ou estimados, em tempo real, das amostras iniciais de tráfego usando-se o processo envelope ótimo.
- Com complexidade computacional mínima, limites para atrasos são derivados com acurada precisão.

Capítulo 5

Gerenciamento de Tráfego em Redes ATM

Um dos maiores desafios em redes ATM é prover recursos que permitam às conexões operarem com diferentes requisitos de Qualidades de Serviço (*Quality of Service - QoS*). Por exemplo, uma conversação telefônica pode tolerar uma taxa de perda alta (até 10^{-3}) sem prejudicar a conversação enquanto que um alto retardo pode ser intolerável, já aplicações de vídeo podem se contentar com um certo retardo, mas não toleram taxas de perda alta (maior que 10^{-9}).

Conforme [CFZ98], o conjunto de funções de gerenciamento pode ser divididas em dois grupos: (1) aquelas que são tomadas durante o estabelecimento da conexão, e (2) aquelas que são tomadas durante a conexão. Como exemplos de mecanismos de gerenciamento de tráfego que se encaixam no grupo (1) tem-se o controle de admissão e roteamento. Policiamento de tráfego, descarte seletivo e controle de fluxo com taxa mínima (*Available Bit Rate - ABR*) são exemplos de mecanismos de gerenciamento que se encaixam no grupo (2).

Este capítulo está dividido da seguinte maneira: na seção 5.1 apresenta-se o mecanismo de controle de admissão. Na seção 5.2 mostra-se o mecanismo de policiamento de tráfego. A seção 5.3 mostra os mecanismos de descarte seletivo. Finalmente na seção 5.4, mostra-se abordagens para controle do fluxo ABR.

5.1 Controle de Admissão de Conexões

O Controle de Admissão de Conexões (*Connection Admission Control - CAC*) é o conjunto de ações tomadas pela rede durante a fase do estabelecimento da chamada (ou durante a fase de renegociação dos parâmetros de conexão) para determinar se uma conexão requisitada pode ser aceita ou rejeitada.

Alguns tipos de chamadas podem envolver mais de uma conexão como é o caso de conexões multimídia ou que envolvam mais de um parceiro (por exemplo, videoconferência). Neste caso, cada conexão deve ser aceita ou não independentemente pelo CAC. Baseado no esquema CAC uma requisição de conexão é admitida somente quando recursos suficientes estão disponíveis para garantir os requisitos de QoS fim-a-fim desta conexão e para todas as outras conexões existentes. Uma das dificuldades enfrentadas pelo CAC é que nem todos os parâmetros de uma determinada fonte de tráfego são conhecidos e, portanto não é possível fazer uma alocação precisa, e certamente haverá desperdício de recursos ou violação dos parâmetros de QoS.

Nesta seção discute-se alguns princípios gerais relacionados com CAC, como por exemplo, quais os critérios para admissão de uma conexão e quais os diferentes princípios e abordagens conceituais no projeto de CAC.

5.1.1 Descritores de Tráfego

No estabelecimento de uma conexão, o usuário e a rede negociam os parâmetros de tráfego tais como: Taxa de Pico de Célula (*Peak Cell Rate - PCR*), Taxa de Célula Sustentada (*Sustainable Cell Rate - SCR*), Taxa de Célula Mínima (*Minimum Cell Rate - MCR*), Tolerância de Rajada Intrínseca (*Intrinsic Burst Transfer - IBT*), Tolerância da Variação do Atraso de Célula (*Cell Delay Variation Tolerance - CDVT*) e os requisitos de QoS solicitados Taxa de Perda de Célula (*Cell Loss Rate - CLR*), Atraso de Transferência de Célula (*Cell Tolerance Delay - CTD*), Variação de Atraso de Célula (*Cell Delay Variation - CDV*) e a Capacidade de Transferência ATM (*ATM Transfer Capability - ATC*). O resultado dessa negociação é o *contrato de tráfego*. A função do contrato de tráfego é permitir uma operação da rede eficiente onde os requisitos de QoS de cada conexão são satisfeitos. ATC diferentes requerem parâmetros de tráfego diferentes, conseqüentemente o conjunto de descritores de conexão pode variar de serviço para serviço. Por exemplo, um serviço que requer capacidade de transferência constante (*Constant Bit Rate - CBR*) não precisa especificar a SCR como um serviço de transferência variável (*Variable Bit Rate - VBR*). Somente uma conexão ABR (*Avaliable Bit Rate*) pode especificar taxa de célula mínima. O CAC pode ser entendido como um sistema no qual as entradas são os descritores da conexão, o CDVT, o QoS e o ATC da conexão proposta, e a saída é a decisão se a conexão será ou não aceita.

5.1.2 Abordagens de CAC

Apesar de muitas propostas para CAC terem sido analisadas e estudadas na literatura, não há clara concordância de qual é a melhor para um dado conjunto de serviços. Nesta seção são discutidas quatro abordagens diferentes para implementação de CAC. A



primeira é a mais simples e a mais conservativa. Ela se baseia na alocação da taxa de pico. A segunda é baseada na aproximação de *buffer* zero ou Multiplexação de Taxa de Envelope (REM - *Rate Envelope Multiplexing*). A terceira é baseada em modelos de tráfego e enfileiramento e leva em consideração a capacidade de *buffer* da rede para compartilhar a carga excessiva durante o período de tráfego pesado. A quarta abordagem baseia-se no conceito de banda passante efetiva que caracteriza o fluxo de uma fonte a partir de suas propriedades estocásticas.

CAC baseada em Alocação de Taxa de Pico - Esta abordagem considera a taxa de pico de cada nova conexão. Se para cada *link* da rede no roteamento da nova conexão a soma das taxas de pico de todas as conexões já em progresso, mais a taxa de pico da nova conexão, for menor do que a capacidade total do *link*, admite-se a nova conexão, caso contrário rejeita-se a nova conexão. Este esquema pode levar a uma subutilização da capacidade da rede (no caso de fluxos VBR), além de não levar em consideração o efeito do CDV.

Enquanto CAC baseado na alocação de taxa de pico pode ser eficiente para fluxo CBR, ele é geralmente ineficiente para fluxos VBR. Em uma conexão VBR típica a taxa de pico é várias vezes maior que a taxa média, logo é possível aceitar um total de conexões cuja soma de taxa de pico seja maior que a capacidade disponível.

O efeito CDV é significativo principalmente no caso de alocação de taxa de pico para fontes CBR. Devido ao CDV, quando se aceita chamadas tais que a soma de diferentes taxas de pico é igual ou bem perto da capacidade disponível, a CLR pode exceder a CLR requerida conforme especificado no contrato de tráfego. Portanto, é importante considerar se o CDV das diferentes conexões assegura que suas CLR's estejam de acordo com o contrato de tráfego.

CAC baseado em Multiplexação de Taxa de Envelope (REM - *Rate Envelope Multiplexing*) - O REM é baseado na hipótese de que não existe *buffer* ou que o tamanho do *buffer* é muito pequeno. Essa hipótese é chamada de "Aproximação de *buffer* zero" (*The Zero Buffer Approximation*). Essa hipótese é interessante pelas seguintes razões. Primeiro, ela é prática em muitas aplicações de tempo real onde os *buffers* são muito pequenos para absorver flutuações em nível de rajada. Portanto, é conveniente assumir que os *buffers* evitam todas as perdas de células que possam ser causadas por flutuações na escala do tempo de célula, e não conseguem evitar as perdas de células devido à flutuações em nível de rajada. Assim sendo, pode-se usar modelos de tráfego em nível de rajada, tais como modelos *on/off* em conjunto com a aproximação de *buffer* zero para obter uma estimativa precisa de perda de células. Segundo, com a aproximação de *buffer* zero não há necessidade de análise de fila. Terceiro, sobre a aproximação de *buffer* zero não há necessidade de se considerar a correlação no processo de chegada, logo pode-se considerar processos com chegadas independentes. Isso torna o processo de modelagem de tráfego e

caracterização mais simples.

Seja S uma variável aleatória representando a quantidade total de trabalho que chega em um intervalo de tempo pequeno (que inclui a quantidade de trabalho que chega das conexões existentes mais a quantidade da nova conexão). Seja C a capacidade de *link* disponível. A CLR é dada por:

$$P_{loss} = \frac{E(S - C)^+}{E(S)}. \quad (5.1)$$

Nesta equação, a CLR é simplesmente a razão entre a quantidade de trabalho perdida e a quantidade de trabalho que chega.

Para se implementar um CAC baseado em REM, precisa-se ter a distribuição de S em todo ponto do tempo que houver uma nova chegada de requisição de conexão. Isso pode ser estimado pelas medidas de tráfego. Tendo-se o comportamento de S , então para uma nova requisição de conexão, computa-se a CLR usando a equação (5.1), caso a CLR prevista seja mais baixa que a CLR requerida (como especificado no contrato de tráfego), a conexão é aceita, caso contrário ela é rejeitada.

CAC baseado em Compartilhamento de Taxa - CAC baseado em REM não é muito eficiente no caso onde comutadores usam grandes *buffers* capazes de absorver grandes rajadas. Há portanto a necessidade de um CAC que leve em consideração as capacidades de *buffer* dos comutadores. Uma das funções importantes de tal CAC é estimar a CLR. Para toda nova chegada de requisição de conexão, tal estimativa irá computar qual será a CLR. Se a conexão for admitida, este irá compara-lá com a CLR requerida. Uma forma de prever a CLR é usar medições e estimativas da distribuição da quantidade de chegadas de células durante diferentes intervalos de tempo. O método é baseado no uso da equação (5.1) para o caso de *buffer* infinito.

Seja $S(t)$ uma variável aleatória que representa a quantidade de trabalho que chega em um intervalo de tempo t . Sabe-se que se durante o tempo t , a quantidade de trabalho que chega é maior do que pode ser servido e ou armazenado nos *buffers* então algum trabalho deve ser perdido. Sabe-se também que a perda deve ser pelo menos o excesso de trabalho acima do que pode ser servido e armazenado. Seja $CLR(t)$ o limite inferior para a proporção de perda de célula baseada na variável aleatória $S(t)$, assim obtém-se:

$$CLR(t) \geq \frac{E(S(t) - C(t) - b)^+}{E(S(t))} \quad (5.2)$$

onde $C(t)$ é a quantidade de trabalho que pode ser servida durante o tempo t e b é o tamanho do *buffer*. Uma vez que $CLR(t)$ para todo t é um limite inferior da CLR, tem-se que:

$$CLR \geq \max CLR(t). \quad (5.3)$$

Considerando-se a equação (5.3) como uma igualdade, obtém-se uma estimativa para CLR. Em [ZT97] demonstra-se que o lado direito da equação (5.3) é um estimador preciso para a CLR.

Banda Passante Efetiva Aditiva - Vários CAC baseados no conceito de banda passante efetiva (*Effective Bandwidth* - EBW) têm sido propostos na literatura [CW95, EM93, GAN91, KWC93]. Em tais esquemas, cada fonte, de forma independente, reserva uma quantidade de banda passante que corresponde a um valor maior que a sua taxa média e menor que a taxa de pico. Essa quantidade de banda passante, o EBW, é estimada em função da probabilidade de perda requerida P_l (qualidade de serviço) e as propriedades estocásticas próprias de um fluxo (ou seja, função de autocorrelação ou taxa de pico e taxa média junto com duração média de rajadas). Tendo sido o EBW do fluxo j determinado, $E_j(P_l)$ (ou, $E_j(P(Q > B))$), o teste realizado pelo CAC requer que

$$\sum_{j=1}^N E_j(P_l) < C,$$

onde N é o número de fluxos multiplexados. Várias técnicas, incluindo decomposição espectral de fluxo Markoviano [EM93], teoria dos grandes desvios [CW95, KWC93] e a teoria de processos envelope [Cha94], têm sido usadas para derivar o EBW que descreve a fonte de tráfego.

Em [CW95], o EBW é definido por

$$E_j(P_l) = r_{av,j} + \frac{\delta \gamma_j}{2B}$$

onde

$$\gamma_j = \lim_{t \rightarrow \infty} \frac{1}{t} \text{var}(A_j[0, t])$$

é o índice de dispersão da seqüência de chegada e

$$\delta = -\frac{\log(P_l)}{B}.$$

O EBW de um fluxo é independente das propriedades de todos os outros fluxos bem como do número de fontes N e da capacidade da linha C . Ele é determinado a partir das propriedades estocásticas do fluxo, probabilidade de perda requerida P_l e o tamanho do *buffer* B .

Os problemas dessa abordagem são: a inadequabilidade quando os fluxos exibem dependência de longa duração; e, simplesmente, adicionando o EBW requerido pelo fluxo não se exploram os efeitos das escalas de tempo quando o número de fluxos multiplexados é elevado.

Uma definição alternativa para o EBW é dada em [Kel96]

$$E_j(s, t) = \frac{1}{st} \log E e^{sA_j[0,t]},$$

de forma que a probabilidade da cauda da distribuição do tamanho da fila satisfaça

$$\lim_{N \rightarrow \infty} \frac{1}{N} \log \Pr(Q > B) = \sup_t \inf_s \left[st \sum_j \rho_j E_j(s, t) - s(b + ct) \right], \quad (5.4)$$

com $C = Nc$, $B = Nb$ e $N\rho_j$ fontes do tipo j .

Essa definição baseia-se na teoria dos grandes desvios, e tem sido usada para fornecer resultados gerais do comportamento assintótico do $\log \Pr(Q > B)$ [DO95, BD95]. A definição de CAC banda passante equivalente baseada nessa caracterização não pode ser dita EBW “aditivo”, uma vez que os recursos requeridos por um único fluxo dependem das propriedades de todos os outros fluxos (os parâmetros s e t expressam essa dependência). As vantagens dessa abordagem são: a possibilidade de identificar e explorar escalas de tempo na multiplexação de fluxos, e uma maior acurácia para os testes realizados pelo CAC uma vez que esse EBW é mais preciso. A desvantagem é que, sem suposições adicionais a cerca do fluxo, a implementação do teste realizado pelo CAC pode ter um alto custo computacional devido ao cálculo do supremo na equação 5.4.

5.2 Mecanismos de Policiamento de Tráfego

Um mecanismo de policiamento é responsável por avaliar se as fontes geradoras de tráfego estão respeitando os parâmetros descritos no contrato de tráfego negociado quando da aceitação da conexão pela rede. A violação do contrato de tráfego pode ser causada por vários fatores, desde a estimativa imprecisa dos parâmetros de tráfego até comportamento malicioso por parte do usuário. Diferentes ações podem ser tomadas por um mecanismo de policiamento quando uma conexão viola seu contrato. Uma célula que causa violação pode ser descartada ou simplesmente marcada para posterior descarte caso visite um ponto congestionado da rede.

O mais conhecido dentre os vários mecanismos de policiamento para redes ATM, é o **Balde Furado (BF - Leaky Bucket)**. No mecanismo de BF, uma célula precisa consumir uma permissão antes de entrar na rede. Permissões são geradas a taxa constante e existe

um número máximo de permissões (tamanho do balde) permitidas em um reservatório de permissões (balde). Se uma célula chega em um ponto da rede e não existe permissão no reservatório, ela será descartada ou marcada. Um mecanismo BF pode também ser implementado por um contador, uma taxa de geração de permissões e um limite. O contador é incrementado a cada chegada de célula até um limite, e é decrementado na taxa de geração de permissões. Toda vez que o contador alcançar o valor limite, o mecanismo BF marca (descarta) células. Diversas variações do BF têm sido propostas na literatura. Uma destas variações tem um *buffer* para “suavizar” o tráfego ao invés de descartar ou apenas marcar a célula (BF com *buffer*). O mecanismo adotado pelo ATM Fórum, conhecido por *The Generic Cell Rate Algorithm*, é equivalente ao mecanismo de balde furado [tAF96].

Além do mecanismo BF, vários mecanismos baseados em janela de tempo têm sido definidos [Rat91]. Num mecanismo **Janela Saltitante (JS - *Jumping Window*)** o tempo é dividido em intervalos de tamanho fixo chamados janelas. O número de células emitidas pela fonte policiada durante qualquer uma dessas janelas é limitado. Para um JS ter uma taxa equivalente ao mecanismo BF a razão entre o limite superior do número de célula permitido durante um intervalo e a duração do intervalo deve ser igual a taxa de geração de permissões. No mecanismo **Janela Saltitante com Gatilho (JSG - *Triggered Jumping Window*)** o início de uma janela é disparado pela chegada de uma célula. Assim, janelas consecutivas não são necessariamente consecutivas no tempo como no caso do JS. Em um mecanismo **Janela Deslizante (JD - *Moving Window*)** uma célula é sempre “lembrada” por um intervalo constante. A diferença entre JD e o JS/JSG é que no primeiro a célula é sempre “lembrada” por um intervalo constante, enquanto que no último ela pode ser “esquecida” após um intervalo curto de tempo. Finalmente, no mecanismo **Média Móvel Ponderada Exponencialmente (MMPE - *Exponentially Weighted Moving Average*)**, o número de células aceitas durante uma janela é determinada por uma soma ponderada pelo número de células aceitas nas janelas anteriores. Em [Rat91], uma comparação mostra que o BF e o MMPE são muito mais promissores que o JS e JSG em relação ao tempo de reação, tráfego de pior caso e sensibilidade a variação de carga.

Uma das principais características da análise de mecanismos de policiamento é determinar a explosividade do processo de saída do mecanismo de policiamento (regulador). Mostrou-se em [Kua94] que o primeiro e o segundo momento do processo de saída do BF com *buffer* depende do tamanho do *buffer* de célula e do tamanho do balde. Em [Lee94], uma expressão para o atraso de enfileiramento do processo de saída de um regulador BF foi derivado. O tráfego de pior caso foi caracterizado por uma chegada em lote (*batch*) do tamanho do balde. Notou-se também que o crescimento assintótico do atraso médio como uma função do tamanho do balde é independente do número de fontes multiplexadas.

Em [SLG89] Sidi *et.al.* mostram que um regulador com *buffer* de célula infinito re-

duz o coeficiente de variação quadrático do processo de saída do BF para fontes Markov modulada. Em [Kue79], demonstrou-se que o BF é capaz de reduzir a explosividade de qualquer processo pontual (*point process*) e que a explosividade diminui com a redução do tamanho do balde. Uma prova da redução da variância do BF foi dada em [AK94]. Em [BW94, BW92] demonstrou-se que num BF com *buffer*, o transbordo do *buffer* de célula depende somente da soma $S_T + S_C$, onde S_T é o tamanho do balde e S_C é o tamanho do *buffer* de célula. Com o aumento do tamanho do balde, porém mantendo-se $S_T + S_C$ constante, aumenta-se o atraso das células no regulador. Contudo, o aumento do atraso tem pouco impacto na redução do requisito tamanho de *buffer* em uma fila que é alimentada por um único processo de saída. Similarmente, achou-se que o *buffer* de célula “suaviza” o processo de saída do BF (coeficiente de variação) em uma escala de tempo curta. Numa escala de tempo longa, a influência do *buffer* não é significativa. Contrastando, quando ocorre multiplexação de um grande número de processos de saída, o *buffer* de célula pode estabelecer dramática “suavização” ao fluxo que parte pela linha. Obviamente, o efeito de sincronização de vários reguladores com idênticas taxas de geração de permissões, pode aumentar significativamente o atraso na fila do fluxo de saída. Adicionalmente, demonstrou-se que a vazão aumenta se ocorre transferência do espaço de *buffer* do esquema de regulador todo-*buffer* para célula (mantendo-se constante $S_T + S_C$) para a fila do fluxo de saída.

Em [AGS93] analisou-se um balde furado com *buffer* via método Matriz Analítica e derivou-se uma solução recursiva elegante para uma fila M/G/1 que evita instabilidade numérica com altas cargas. Notou-se que a distribuição de tamanhos da cauda da fila de um BF com *buffer* não é geométrica. Indicações para definir os parâmetros do BF foram derivadas. Para cargas entre 0.75 e 0.8, o tamanho do balde deve ser aproximadamente duas vezes o tamanho máximo do lote (*batch*). Essas cargas correspondem a uma taxa de geração de permissões 30% maior que a taxa média de chegada. Observou-se também que para pequenos tamanhos de balde, o coeficiente de variação quadrática do processo de saída aumenta com a variabilidade da distribuição do tamanho do lote (*batch*) do processo de entrada.

Em um mecanismo de policiamento que marca células para posterior descarte, pode haver intervalos no fluxo de células correspondendo a períodos de violação. Em [LA96] diferentes políticas para marcação de células foram propostas de forma a minimizar o tamanho do intervalo sem células. Essa minimização é alcançada mudando-se o procedimento de marcação sem alterar os parâmetros do Balde Furado. Um esquema de marcação alterna entre marca com violação/de acordo sempre que o número de permissões disponível cai abaixo de um certo valor limite. Essa política de marcação pode sempre ser modificada para marcar como violadora cada duas ou três células que estão de acordo, ao invés de marcar todas que estão de acordo. Mostrou-se que essa política minimiza a média de

intervalo sem perda para sistemas com entrada Poisson e fontes *on-off*.

Nem os mecanismos de BF, nem os mecanismos baseados em janelas são eficientes no policiamento de fontes VBR [BCT91, YSS95]. De fato, todas as fontes descritas por suas taxas média e de pico são difíceis de monitorar usando esses reguladores. Na realidade, o BF e os mecanismos baseados em janelas não apresentam muita flexibilidade devido o pequeno número de parâmetros disponíveis para ajustar o processo de policiamento. Por exemplo, para as fontes que transmitem seu tráfego em rajadas, se houver um limite pequeno no mecanismo de BF, células bem comportadas podem ser marcadas e a probabilidade de violação pode ser alta. No caso contrário, se o limite for alto, o tempo para detectar a violação é alto. Além disso, a explosividade do fluxo de saída de um sistema de fila aumenta com o limite. Outra opção é escolher uma taxa de geração de permissão alta, mas assim não seria possível policiar a taxa média da fonte. A solução usual é escolher uma taxa de geração de permissão perto da taxa de pico da fonte, o que pode desperdiçar banda passante uma vez que fontes que transmitem seu tráfego em rajada usualmente não operam na taxa máxima o tempo todo. Em mecanismos baseados em janela o tempo de reação a violação é fortemente dependente da duração da janela para uma certa taxa.

Em [HY96], um experimento foi apresentado no qual uma conexão transmitindo tráfego VBR mal-comportado passou pelo BF sem ser detectada. Este fato também foi demonstrado para rajadas médias, e para rajadas maiores que as negociadas. Além disso, a alocação de banda passante de acordo com a taxa permitida não é suficiente para limitar o atraso médio na fila do fluxo de saída.

Para superar a limitação dos mecanismos de policiamento tradicionais, mecanismos de policiamento multi-estado foram definidos. Em cada estado de um mecanismo multi-estado existe um mecanismo de estado simples [JSS96]. Por exemplo, em cada estado de um mecanismo de Balde Furado multi-estado (*MultiLevel Leaky Bucket*) existe uma taxa de geração de permissão, um contador e um limite [MS95]. Quando o limite do estado i é atingido, o mecanismo pula para o próximo estado que tem uma taxa de geração de permissão mais alta. Quando o contador voltar a zero, o mecanismo retorna ao estado anterior. Além desses parâmetros, em cada estado existe um tempo máximo de residência dentro de um intervalo de duração fixa (ciclo). Se o tempo de residência em um certo estado excede seu valor máximo permitido, células são consideradas em violação. Demonstrou-se em [JSS96] que um mecanismo multi-estado é mais eficiente no monitoramento de fontes VBR do que mecanismos de estado simples, uma vez que o mecanismo de multi-estado reage mais rápido a mudança de estado da fonte. Além disso, mecanismos multi-estados são mais apropriados para detectar variações no tamanho da rajada que mecanismos de estado simples. Mecanismo de estado simples pode monitorar de forma ineficiente fontes que operam em conformidade com o contrato. A figura 5.1 ilustra esse fato mostrado que enquanto o BF alcança uma alta probabilidade de violação

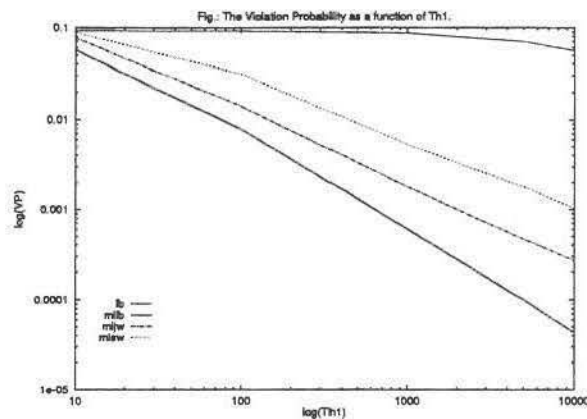


Figura 5.1: Probabilidade de Violação X Limites (de [JSS96] © IEEE)

para altos valores de limite tem-se uma forma decrescente para os mecanismos de multi-estado.

5.3 Mecanismos de Descarte Seletivo

Multiplexação estatística foi adotada no padrão ATM a fim de se maximizar os recursos da rede frente à natureza da taxa variável de fontes multimídia. Contudo, o ganho obtido através da multiplexação estatística vem com um custo: células concorrem por recursos compartilhados, e portanto, algumas destas podem ser perdidas. Isso ocorre porque nas redes multimídia enfrenta-se a difícil tarefa de satisfazer às necessidades de conexões que requerem diferentes QoS. Os requisitos de QoS de perda de uma aplicação são usualmente traduzidos em duas métricas: a taxa de perda (razão entre as células perdidas e o número total de células transmitidas) e o tamanho do intervalo de perdas (o número de células perdidas consecutivamente). O descarte seletivo é um mecanismo que visa habilitar a rede a lidar com diversos requisitos de perda. Em um mecanismo de descarte seletivo células são descartadas de acordo com seus níveis de prioridade.

Descarte seletivo está relacionado somente com o gerenciamento do espaço de *buffer* e não com a ordem de transmissão. Um mecanismo de descarte seletivo é completamente especificado por uma política de organização de *buffer* e pela política de descarte. Uma política de organização de *buffer* define qual parte do *buffer* pode ser ocupado por qual célula, uma política de descarte escolhe uma célula para ser descartada entre as células da classe com prioridade mais baixa. As políticas de organização de *buffer* mais comuns são [KK80]:

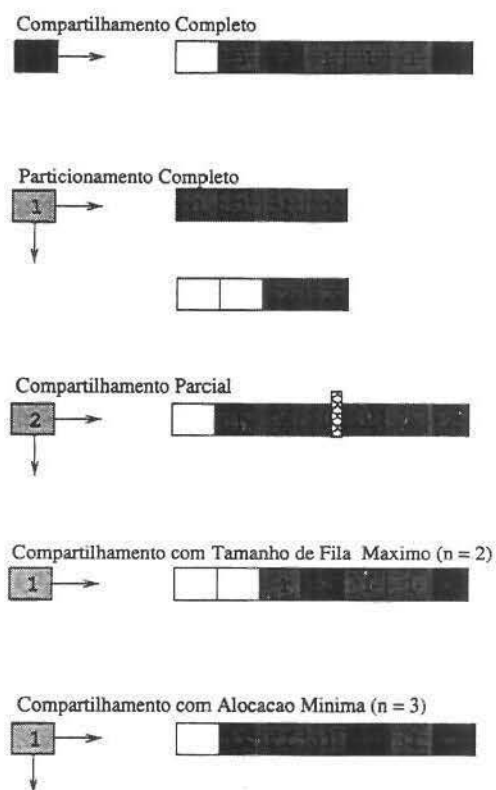
- (i) **Particionamento Completo** - cada classe de cliente tem sua própria fila indivi-

dual. Na realidade, não existe compartilhamento de *buffer*. Cada espaço de *buffer* pode ser otimizado para manipular seu fluxo de tráfego esperado. Apesar de ser uma política atrativa por não ter custo adicional de controle, perdas ocorrem mesmo se existir espaço livre em algumas das filas;

- (ii) **Compartilhamento Total** - todos os espaços de *buffer* são acessíveis por todos os clientes. Células são descartadas se e somente se o *buffer* está cheio. Ela minimiza a perda total de célula;
- (iii) **Compartilhamento Total com Descarte** - funciona da mesma maneira que a política de Compartilhamento Total, mas se uma célula encontra o *buffer* cheio e existe uma célula da classe com prioridade baixa na fila então esta célula é descartada (Compartilhamento Total é algumas vezes usado para designar Compartilhamento Total com Descarte);
- (iv) **Compartilhamento Parcial** - cada classe tem uma posição limite até a qual células podem ser aceitas dentro do espaço de *buffer*. Se uma célula chega e existe mais células na fila do que a posição limite da sua classe então a célula é perdida;
- (v) **Compartilhamento com Tamanho de Fila Máximo** - existe um limite máximo para o número de clientes de cada classe;
- (vi) **Compartilhamento com Alocação Mínima** - existe uma quantidade mínima de espaço de *buffer* alocado para cada classe de clientes. O resto do espaço de *buffer* é compartilhado por todas as classes de clientes.
- (vii) **Compartilhamento Total com Garantia de Tamanho Mínimo de Fila** - difere do compartilhamento total por garantir um número mínimo de *buffers* para cada classe em situação de transbordo [LZ]. Nestas situações, células das classes que excedem este mínimo são descartadas, de acordo com seu nível de prioridade.

A Figura 5.2 ilustra os cenários de perda para diferentes políticas de organização de *buffer*.

Uma política de organização de *buffer* é chamada de política de conservação de taxa de perda, se células (pacotes de tamanho fixo) são perdidas somente em situações de transbordo. Em outras palavras, uma disciplina que conserva a taxa de perda sempre admite uma célula no *buffer* se existir espaço disponível. Das políticas de organização citadas acima, Compartilhamento Total (com e sem descarte) e Compartilhamento Total com Garantia de Tamanho Mínimo de Fila são exemplos de disciplinas que conservam a taxa de perda. Um exemplo de disciplina que não conserva a taxa de perda é a Compartilhamento Parcial, na qual células da classe com prioridade baixa podem ocupar até uma

Figura 5.2: Exemplos de políticas de organização de *buffer*.

certa posição no *buffer* e conseqüentemente perdas podem ocorrer mesmo se o *buffer* não estiver cheio. Disciplinas que conservam a taxa de perda são de interesse especial porque elas minimizam a perda total de células e conseqüentemente maximizam a vazão.

O nível de prioridade de uma célula pode ser definido estatisticamente ou dinamicamente. Se atribuído estatisticamente, o nível de prioridade indica a importância de uma célula. Por exemplo, células que carregam informações importantes em um fluxo, tais como células MPEG do quadro *I*, devem ter prioridade alta. Além disso, se todas as células de uma aplicação tem o mesmo nível de prioridade, eles indicam os requisitos de perda dessa aplicação. Prioridades atribuídas dinamicamente são relacionadas à discrepância entre os parâmetros negociados e os observados da transmissão de uma conexão. Como por exemplo, um mecanismo de policiamento marca com prioridade baixa as células correspondentes a períodos de violação, períodos nos quais as características de transmissão de uma conexão não estão de acordo com os valores negociados em tempo de admissão da conexão.

Uma política de descarte seleciona uma célula para ser descartada entre as células da classe com prioridade baixa. As políticas mais comuns são Último-a-Chegar-Primeiro-Descartado (UCPD - *Last-In-First-Drop*), Primeiro-a-Chegar-Primeiro-Descartado (PCPD - *First-In-First-Drop*), Seleção Aleatória (RAND - *Random*) e Primeiro-a-Chegar-Primeiro a ser-Descartado Modificada (PCPD-M - *Modified First-In-First-Drop*). A política PCPD Modificada sempre descarta uma célula da classe com prioridade mais baixa para liberar espaço de *buffer* para uma célula que chegue, independente do nível de prioridade da célula que chegue. Descartar células da classe com prioridade baixa em diferentes posições define diferentes distribuições de fila e, conseqüentemente, pode ter impacto diferenciado no parâmetro de QoS. Como por exemplo, suponha que existam duas células da classe com prioridade baixa na fila, uma no final e outra no início (Figura 5.3). Se uma célula da classe com prioridade alta encontra o *buffer* cheio, conforme a política UCPD, a célula da classe com prioridade baixa no final da fila é descartada. De acordo com a política PCPD a célula do início da fila é descartada. Após chegada de várias células com prioridade alta, o *buffer* estará cheio novamente. Na fila com UCPD, a célula no início da fila antes das chegadas de células da classe com prioridade alta (que não foi descartada) já foi transmitida e, conseqüentemente, a célula da classe com prioridade alta é perdida. Na fila com PCPD, contudo, a célula da classe com prioridade baixa no final da fila antes das chegadas de células da classe com prioridade alta não foi descartada e pode ser agora descartada para liberar espaço para a nova célula da classe com prioridade alta. A política PCPD-M é a expressão máxima da diferença entre PCPD e UCPD pois a política PCPD-M tenta concentrar as células da classe com prioridade baixa no final da fila e, portanto, aumenta a probabilidade de uma célula da classe com prioridade baixa ser descartada. Dessa forma, a política PCPD-M minimiza a taxa de perda da classe com prioridade alta [THP94].

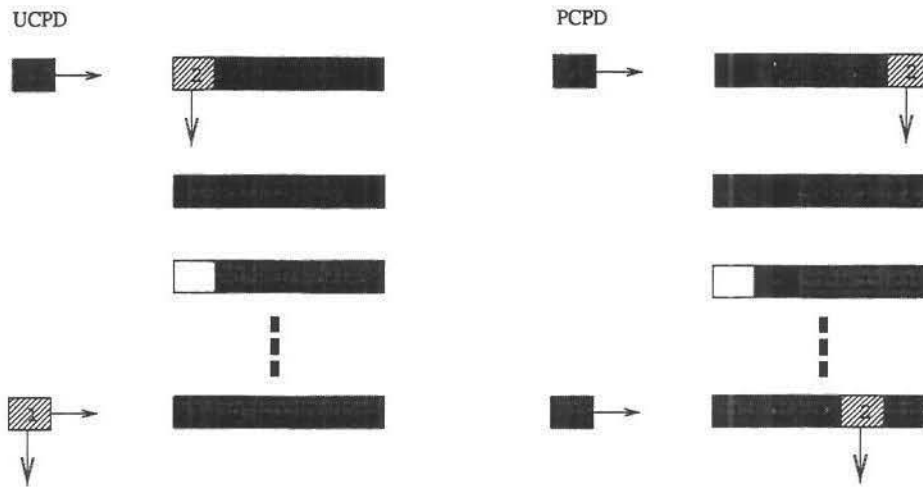


Figura 5.3: Um exemplo de perda de célula devido a diferentes distribuições de tamanho de fila produzidas pelas políticas de descarte UCPD e PCPD.

5.4 Controle de Fluxo ABR

O serviço ABR é definido para o suporte de aplicações que podem requerer banda passante mínima. Ele garante uma proporção baixa ou zero de perda de célula e uma distribuição justa da banda passante disponível ao se moldar o tráfego de acordo com a informação de retorno recebida da rede. Tal serviço é geralmente solicitado por aplicações de dados que não podem prever seus próprios requisitos de banda passante, e esperam conseguir algum compartilhamento da banda passante disponível. Em contraste, a categoria de serviço *Unspecified Bit Rate* (UBR) oferece um serviço de melhor-esforço sem garantias de QoS. Este serviço é projetado para aplicações de dados que querem usar banda passante disponível e não são sensíveis a perda e retardo de célula. Uma conexão UBR não é rejeitada baseada na falta de banda passante e não é policiada pela rede.

O serviço ABR permite que aplicações utilizem banda passante não usada pelos serviços VBR e CBR, pelo ajuste contínuo de suas taxas de transmissão instantânea à capacidade disponível no momento para ABR. Um esquema de controle de congestionamento é essencial para o suporte de tráfego ABR utilizar a banda passante disponível sem causar congestionamento, e para dividir essa banda passante entre as conexões ABR de maneira justa. Computar a divisão justa de uma conexão ABR de um *link* de saída de um comutador ATM requer um algoritmo de alocação justa. Outro algoritmo é necessário no final do sistema para ajustar a sua taxa ao retorno recebido da rede. O comportamento da fonte é importante para garantir a taxa de perda com prioridade baixa. Se uma fonte não se comporta como esperado, isso pode aumentar a taxa de perda de todos os usuários

ABR.

5.4.1 Alocação *Max-Min*

O serviço ABR deve supostamente dividir a banda passante deixada pelas conexões VBR e CBR entre conexões ABR dando a cada fonte ABR uma parte justa. Existem muitas definições para justiça. O Forum ATM decidiu pela definição particular, chamada de justiça *max-min* [Jaf81], que maximiza a vazão total. Alocação *max-min* permite a todas as conexões VC (*Virtual Channel*) usando um *link* em comum compartilhar igualmente a banda passante disponível tanto quanto elas não provoquem um gargalo em outro lugar da rede. As conexões VC competindo por banda passante em um *link* l são divididas em duas categorias: conexões restritas (*constrained*) e conexões não restritas (*unconstrained*). Conexões restritas são aquelas que não podem usar sua divisão justa em l devido à banda passante limitada em outro *link*. Conexões não restritas são aquelas que não podem usar sua divisão justa em outro *link* devido à banda passante limitada em l . Para essas conexões, o *link* l é referido como o “*link* gargalo” (*bottleneck link*). A idéia básica por trás da justiça *max-min* é alocar as conexões não restritas de l e compartilhar igualmente a banda passante deixada pelas conexões restritas.

Um algoritmo centralizado para computar alocações *max-min* trabalha da seguinte maneira. Suponha que a rede tenha L *links*, (designados) por $1 \dots L$. Para cada *link* l , indicando por b_l a capacidade (banda passante) de l , por C_l o conjunto de conexões VC ABR atravessando l , e pelo $|C_l|$ o número de conexões. Uma seqüência de alocação de taxas podem ser representadas por um vetor β no *link* l . Tal alocação é dita *max-min* justa se as seguintes condições são satisfeitas:

- (c1) - A taxa $\beta(c)$ alocada por β para uma conexão, c , VC ABR não é negativa.
- (c2) - Para cada *link* l , $1 \leq l \leq L$, $\beta[l] \geq b_l$.
- (c3) - $\beta(c)$ não pode ser aumentado sem diminuir a taxa alocada para outra conexão c' para qual $\beta(c) \geq \beta(c')$.

Foi demonstrado que as condições (c1)-(c3) são preenchidas por β se e somente se cada conexão c tem um *link* de gargalo, isto é, um *link* l sobre o qual a taxa atribuída para c não pode ser aumentada porque toda a capacidade l foi alocada (*i.e.* $B[l] = b_l$) e porque nenhuma das VCs atravessando l foi atribuída mais banda passante do que c (*i.e.* para cada $c' \in C_l$, $\beta(c) \geq \beta(c')$).

O algoritmo para achar um vetor *max-min* justo trabalha em iterações até achar o *link* gargalo para cada conexão VC. Ele inicia com um vetor β^0 de taxa zero, o que obviamente preenche as condições (c1) e (c2). Ele então aumenta algumas das componentes de β^0

tal que (c2) continue valendo e um *link* engarrafado seja encontrado, neste caso β^1 é definido. Para garantir que (c2) não é violada o algoritmo verifica no *link* l , qual a banda passante disponível dividida pelo número de VCs não engarrafados ainda é “*mínimo*”. O algoritmo então aloca uma distribuição igual a banda passante do *link* selecionado para toda VC atravessando esse *link*. Essa alocação afeta a banda passante disponível nos outros *links*, que ainda não são gargalos. Então, o algoritmo desconsidera o *link* gargalo e todas as VCs atravessando esse *link*, e continua para a próxima iteração onde um novo *link* é selecionado. Esse processo continua até todas as VCs se tornarem gargalo.

Esse algoritmo não é útil em uma rede ATM dinâmica real, porque ele assume um controle centralizado. Contudo, seus conceitos principais são úteis para o projeto de algoritmos distribuídos, para ser executado pelos comutadores ATM e nós de fontes.

Capítulo 6

Multiplexação Estatística de Fontes Auto-semelhantes

Neste capítulo utiliza-se o conceito de escala de tempo em que a fila alcança sua ocupação máxima, num sentido probabilístico, para derivar expressões que estimam a banda passante e o tamanho do *buffer* demandados por um agregado de fontes auto-semelhantes que alimenta essa fila. Propõe-se, essencialmente, um modo de computar a banda passante necessária para prover um certo requisito de perda, bem como um atraso máximo probabilístico para um agregado de fontes. O problema estudado neste capítulo pode ser definido como: *Dado um conjunto de fontes com média \bar{a}_i , desvio padrão σ_i e parâmetro Hurst H_i , qual deve ser a capacidade do canal para que o tamanho máximo da fila seja limitado por q_{max} com probabilidade ϵ ?* Na seção 6.1 discute-se o conceito de escala de tempo de interesse num sistema de filas alimentado por um processo mBf. Na seção 6.2 utilizam-se a MaxTS e o processo envelope do movimento Browniano fractal, para derivar as ferramentas analíticas que possibilitarão responder à questão proposta anteriormente. Os resultados numéricos das avaliações realizadas são apresentados na seção 6.3. Um algoritmo para controle de admissão de conexões é discutido na seção 6.4.

6.1 Escala de Tempo de Interesse

6.1.1 Tempo até a Fila Alcançar o Máximo

Nesta seção mostra-se o instante no qual uma fila alcança sua ocupação máxima num sentido probabilístico. O tamanho da fila no referido instante fornece um limite simples para o cálculo do atraso. Uma derivação matemática mais rigorosa pode ser encontrada em [MS98]. Apresenta-se uma derivação heurística de forma a enfatizar a intuição por trás dos conceitos básicos presentes neste capítulo. Considera-se uma fila em tempo contínuo,

com serviço determinístico dado por c . O processo de chegada acumulado é dado por $A_H(t)$ ($A_H(0) = 0$). Tem-se $\hat{A}_H(t)$, contínuo e diferenciável, como o processo envelope probabilístico de $A_H(t)$, ou seja:

$$\Pr(A_H(t) > \hat{A}_H(t)) \leq \epsilon.$$

Durante um período de ocupação, que inicia no tempo 0, o número de células no sistema no tempo t é dado por $q(t)$.

$$q(t) = A_H(t) - ct \geq 0. \quad (6.1)$$

Define-se $\hat{q}(t)$ como,

$$\hat{q}(t) = \hat{A}_H(t) - ct \geq 0 \quad (6.2)$$

tem-se conseqüentemente que:

$$\Pr(q(t) > \hat{q}(t)) = \Pr(A_H(t) > \hat{A}_H(t)) \leq \epsilon.$$

O atraso máximo em uma fila FIFO é dado pelo número máximo de células na fila durante o período de ocupação. Define-se,

$$q_{max} \stackrel{\text{def}}{=} \max(\hat{q}(t)), t \geq 0.$$

Dessa forma,

$$\Pr(q(t) > q_{max}) \leq \Pr(q(t) > \hat{q}(t)) \leq \epsilon$$

$$\Pr(q(t) > q_{max}) \approx \epsilon.$$

Pode-se dizer que o tamanho da fila no tempo t , $q(t)$, excederá o tamanho máximo q_{max} com probabilidade ϵ . Em outras palavras, somente quando o processo de chegada exceder o processo envelope, o número máximo de células na fila irá exceder o seu valor estimado. Limitando-se o comportamento do processo de chegada é possível transformar o problema de achar um limite probabilístico de um sistema estocástico definido por 6.1 num problema, mais fácil, de achar o máximo de um sistema determinístico descrito por 6.2.

Para o caso de um processo mBf, substitui-se o processo envelope definido previamente na equação 6.2 obtendo-se:

$$\hat{q}(t) = \hat{A}_H(t) - ct = \bar{a}t - \kappa\sigma t^H - ct \quad (6.3)$$

Para se computar q_{max} é necessário achar t^* tal que:

$$\frac{d\hat{q}(t^*)}{dt} = 0$$

ou de forma equivalente,

$$\frac{d\hat{A}_H(t^*)}{dt} = c. \quad (6.4)$$

Assim sendo, t^* é dado por:

$$t^* = \left[\frac{\kappa\sigma H}{(c - \bar{a})} \right]^{\frac{1}{1-H}}.$$

A escala de tempo de interesse é definida pelo tempo necessário para que o tamanho da fila alcance o seu maior valor, i.e., t^* . Chama-se esse valor de escala de tempo máxima (MaxTS), correspondendo ao instante de tempo no qual o trabalho não realizado na fila do sistema alcança seu valor máximo no sentido probabilístico. Isto significa que a taxa média de chegada deve estar abaixo da capacidade da linha, de forma que o tamanho da fila comece a diminuir. Sabe-se que a taxa média de chegada converge para a taxa média da fonte pela lei dos grandes números. Assim sendo, deve-se dispensar atenção somente para a escala de tempo na qual a taxa média de chegadas excede a capacidade da linha, num sentido probabilístico. Em outras palavras, após um intervalo de tempo, a probabilidade da taxa média de chegada exceder a capacidade da linha pode ser negligenciada, de tal forma que o modelo de chegada não precisa reproduzir o comportamento da fonte em todas as escalas de tempo.

Substituindo t^* na equação 6.2, obtêm-se:

$$q_{max} = \hat{A}_H(t^*) - ct^* \quad (6.5)$$

$$q_{max} = (c - \bar{a})^{\frac{H}{H-1}} (\kappa\sigma)^{\frac{1}{1-H}} H^{\frac{H}{1-H}} (1 - H).$$

Uma vez que o processo mBf não excede $\hat{A}_H(t)$ com probabilidade $1 - \epsilon$, o número máximo de células é limitado por q_{max} com a mesma probabilidade. Computa-se \hat{c} para um valor de q_{max} que corresponde a um tamanho de *buffer* K . Em outras palavras, num *buffer* de tamanho K haverá transbordo com probabilidade ϵ se a capacidade da linha for dada por \hat{c} . Assim, \hat{c} é dado por

$$\hat{c} = \bar{a} + K^{\frac{H-1}{H}} (\kappa\sigma)^{1/H} H(1 - H)^{\frac{H-1}{H}}.$$

Esse resultado foi obtido anteriormente por Norros [Nor94] e Duffield [DLO⁺95]. Em resumo, os conceitos básicos usados aqui permitem computar limites para atraso em uma

fila determinística com esforço computacional pequeno, alcançando a mesma acurácia de resultados derivados a partir da teoria dos grandes desvios.

6.1.2 Escala de Tempo Crítica

Em [MdV96] Montgomery propõe um método para investigar a escala de tempo em que perdas de células são mais prováveis de ocorrer. Nessa seção, um sistema de fila ATM alimentado por uma fonte mBf é avaliado usando esse método. Seguindo [Nor94], assume-se que o processo de chegada $A_H(t)$ é um processo mBf dado por:

$$A_H(t) = \bar{a}t + \sigma Z(t) \quad (6.6)$$

onde \bar{a} é a taxa média de entrada, $\sigma > 0$ é o desvio padrão, $H \in [1/2, 1)$ é o parâmetro de auto-semelhança (Hurst) e $Z(t)$ é um movimento Browniano fractal normalizado. A probabilidade que durante um intervalo de tempo de duração t a fonte, $A_H(t)$, gere mais tráfego que a taxa de serviço, ct , a ponto de exceder um *buffer* de tamanho b é dada por:

$$Pr(A_H(t) > ct + b) = Pr(\bar{a}t + \sigma Z(t) > ct + b)$$

$$Pr\left(Z(t) > \frac{t(c - \bar{a}) + b}{\sigma}\right).$$

Pela propriedade da auto-semelhança $Z(t) = t^H Z(1)$, tem-se que:

$$Pr\left(Z(1) > \frac{t(c - \bar{a}) + b}{\sigma t^H}\right) = \Phi\left(\frac{t(c - \bar{a}) + b}{\sigma t^H}\right)$$

onde $\Phi(y) = Pr(Z(1) > y)$ é a função de distribuição residual da distribuição Gaussiana padrão. Assim, usando a aproximação 4.3 obtêm-se

$$Pr(A_H(t) > ct + b) = \Phi\left(\frac{t(c - \bar{a}) + b}{\sigma t^H}\right) \approx \exp\left(\frac{-1}{2}g(t)^2\right)$$

$$\exp\left(\frac{-1}{2}g(t)^2\right) = \exp\left(\frac{-1}{2}\left(\frac{t(c - \bar{a}) + b}{\sigma t^H}\right)^2\right).$$

Portanto, o mínimo $t' \in \operatorname{arg\,inf}_{t>0} g(t)$ indica que a probabilidade de transbordo é maximizada, e que isto pode corresponder a escala de tempo mais provável onde se verifica transbordo no sistema. Disso, t' é dado por:

$$t' = \frac{bH}{(c - \bar{a})(1 - H)}.$$

Esse resultado foi derivado independentemente por Addie *et al* [AEN95] e Ryu *et al* [RE96]. Ryu chamou isto de Escala de Tempo Crítica (*Critical Time Scale-CTS*) e argumenta que esta é a mais importante escala de tempo na determinação de perdas de células em um sistema de filas ATM. No entanto, Mayor em [MS98] apresenta uma outra interpretação para a CTS, indicando que essa é a mais provável escala de tempo quando ocorre transbordo do *buffer*, porém ela não indica nada a respeito da distribuição conjunta da variável randômica que descreve a probabilidade do transbordo, caso ele ocorra em qualquer tempo t .

6.1.3 Escala de Tempo Mínima para que Ocorra Perda de Células

Em [MS96] estudou-se a eficácia de mecanismos de detecção de congestionamento baseado em taxa, e definiu-se uma nova escala de tempo chamada escala de tempo mínima (*Minimum Time Scale-MinTS*) de perda de células. Essa escala de tempo é tal que a probabilidade de ocorrer perda num sistema ATM de filas antes da MinTS é muito baixa. Seguindo a mesma abordagem anterior, a MinTS é derivada a partir da seguinte equação:

$$Pr(A_H(t) > ct + b) \leq \epsilon. \quad (6.7)$$

Substituindo o mBf de 6.6 em 6.7 e usando a mesma abordagem, a equação 6.7 reduz-se a:

$$t(c - \bar{a}) + \kappa\sigma t^H - b = 0. \quad (6.8)$$

Assim sendo, pode-se derivar MinTS usando a caracterização do processo envelope. Nesse caso, MinTS é dado por t'

$$t' \stackrel{\text{def}}{=} \inf\{t \geq 0 : \hat{q}(t) > b\}.$$

Assim,

$$\hat{q}(t) = t(c - \bar{a}) + \kappa\sigma t^H > b.$$

Logo, MinTS é dado pela seguinte equação

$$t(c - \bar{a}) + \kappa\sigma t^H - b = 0.$$

Chegou-se a equação 6.8 como antes, com pequeno esforço computacional. Pode-se dizer, intuitivamente, que se o processo de chegada não excede o processo envelope, então o transbordo do *buffer* não irá ocorrer antes do tempo MinTS. O MinTS estimado tem várias

aplicações. Por exemplo, ele pode ser usado para definir os parâmetros de um algoritmo de detecção de congestionamento baseado em taxa [MS96]. Esse mecanismo computa a taxa média de chegada durante intervalos de tempo fixo, consecutivos, e não sobrepostos de tamanho t . Sempre que esta taxa estimada excede a taxa limite, assume-se que o congestionamento é eminente. t pode ser derivado a partir da MinTS estimada. Se $t \gg \text{MinTS}$, perdas de célula podem ocorrer antes do mecanismo detectar o congestionamento. Por outro lado, se $t \ll \text{MinTS}$, pode-se assumir, erroneamente, que pequenas flutuações estocástica da taxa seja um evento de congestionamento.

6.2 Resultados Analíticos

Assume-se a existência de N fontes independentes com $A_H^i(t)$ definido pelos seguintes parâmetros: média \bar{a}_i , desvio padrão σ_i e parâmetro Hurst H_i para $i \in [1, N]$. O tráfego agregado é dado por $A_H(t) = \sum_{i=1}^N A_H^i(t)$. O processo envelope de cada fonte é dado por $\hat{A}_H^i(t)$. O processo envelope do tráfego agregado é dado por $\hat{A}_H(t)$. A média do tráfego agregado é dada pela soma das médias de cada fonte. Da mesma forma, uma vez que as fontes são independentes, a variância do tráfego agregado é dada pela soma das variâncias de cada fonte. Assim, o processo envelope do tráfego agregado é definido por

$$\hat{A}_H(t) = \sum_{i=1}^N \bar{a}_i t + \kappa \left(\sum_{i=1}^N \sigma_i^2 t^{2H_i} \right)^{1/2}.$$

Substituindo $\hat{A}_H(t)$ na equação 6.4, tem-se:

$$\kappa \frac{1}{2} \left(\sum_{i=1}^N \sigma_i^2 t^{2H_i} \right)^{-1/2} \left(\sum_{i=1}^N \sigma_i^2 2H_i t^{2H_i-1} \right) = c - \sum_{i=1}^N \bar{a}_i. \quad (6.9)$$

Resolve-se a equação 6.9 através de métodos numéricos para se obter t^* e substitui-se t^* na equação 6.5 para se obter q_{max} .

Além disso, combinando as equações 6.4 e 6.5, tem-se:

$$\kappa \frac{1}{2} \left(\sum_{i=1}^N \sigma_i^2 t^{2H_i} \right)^{-1/2} \left(\sum_{i=1}^N \sigma_i^2 2H_i t^{2H_i-1} \right) - \kappa \left(\sum_{i=1}^N \sigma_i^2 t^{2H_i-2} \right)^{1/2} + \frac{q_{max}}{t} = 0. \quad (6.10)$$

Usando as equações 6.9 e 6.10 pode-se responder à questão fundamental proposta no início dessa seção. Para o caso especial da multiplexação de N fontes homogêneas, o processo envelope é dado por $\hat{A}_H(t) = N\bar{a}t + \sqrt{N}\kappa\sigma t^H$ uma vez que o parâmetro Hurst é preservado quando N fontes idênticas são agregadas [ENW96]. Neste caso a equação 6.9 é reduzida a:

$$\frac{\kappa (N\sigma^2 2Ht^{2H-1})}{2 (\sqrt{N}\sigma t^H)} = N(c - \bar{a}).$$

Usando-se a mesma abordagem, computa-se t^* e q_{max} :

$$t^* = \left[\frac{\sqrt{N}\kappa\sigma H}{N(c - \bar{a})} \right]^{\frac{1}{1-H}} = N^{\frac{1}{2(H-1)}} t_i^*$$

$$q_{max} = N(\bar{a} - c)N^{\frac{1}{2(H-1)}} t_i^* + N^{\frac{H}{2(H-1)}} N^{1/2} \kappa\sigma (t_i^*)^H = N^{\frac{(H-1/2)}{H-1}} \hat{q}_{max}$$

onde:

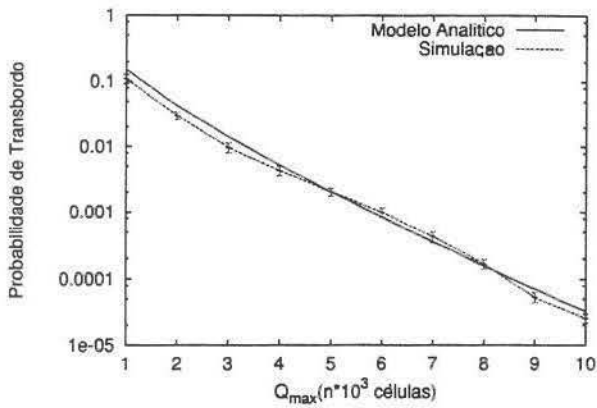
$$t_i^* = \left[\frac{\kappa\sigma H}{(c - \bar{a})} \right]^{\frac{1}{1-H}} \text{ e } \hat{q}_{max} = \hat{A}_H^i(t_i^*) - ct_i^*$$

t_i^* e \hat{q}_{max} correspondem a um sistema de filas com uma única fonte de entrada.

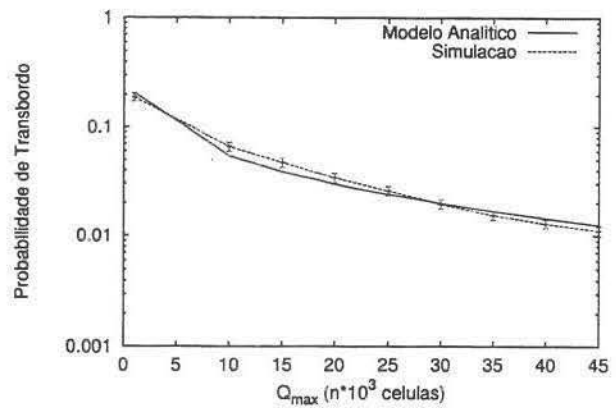
6.3 Resultados Numéricos

Analisa-se primeiramente o caso específico de uma única fonte. A Figura 6.1 mostra a probabilidade de transbordo do *buffer* em função do tamanho do *buffer* para uma utilização de 60% da linha. Compara-se a probabilidade de transbordo dada pelos modelos analíticos com a probabilidade de transbordo observada nos experimentos de simulação. É possível perceber que a probabilidade de transbordo apresentada pela resolução analítica está bem próxima das obtidas pelo experimento de simulação. Observa-se nos resultados derivados dos experimentos de simulação que para altos valores do parâmetro Hurst, a precisão da previsão é ainda maior. Além disso, observou-se que a precisão da estimativa de banda passante cresce com a utilização do canal.

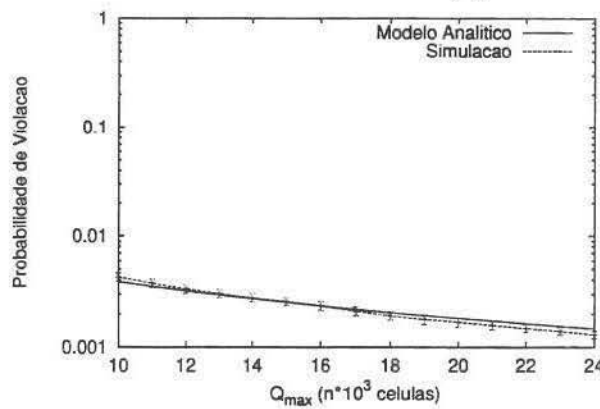
Para avaliar a eficácia das expressões que calculam a banda passante (equações 6.9 e 6.10) define-se o ganho obtido com a multiplexação estatística como sendo a razão entre N vezes a banda passante de uma única fonte e a banda passante de N fontes. Observa-se que o ganho pode ser significativo quando se considera a multiplexação de fontes homogêneas. Na Figura 6.2 mostra-se o ganho com a multiplexação em uma linha operando com a capacidade de 150Mbps quando são multiplexadas fontes com taxa média de chegada de 1.1Mbps e diferentes parâmetros Hurst. A figura 6.2.a mostra o ganho com a multiplexação para fontes com $\sigma = 0.10$ enquanto a figura 6.2.b considera fontes com $\sigma = 0.55$. Observa-se que o ganho para o fluxo com desvio padrão mais alto ($\sigma = 0.55$) é significativamente maior que para fluxos onde o desvio padrão é baixo ($\sigma = 0.10$). Enquanto que para fluxos com $H = 0.9$ e $\sigma = 0.10$ o ganho alcançado é de apenas 1.25, este chega a mais de 2.5 quando o fluxo tem $\sigma = 0.55$.



(a) Parâmetro Hurst = 0.63

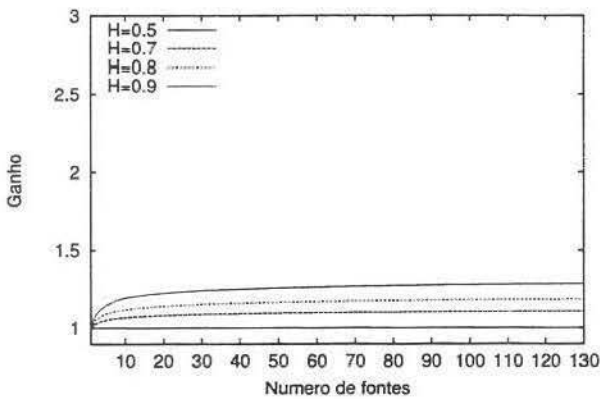


(b) Parâmetro Hurst = 0.86

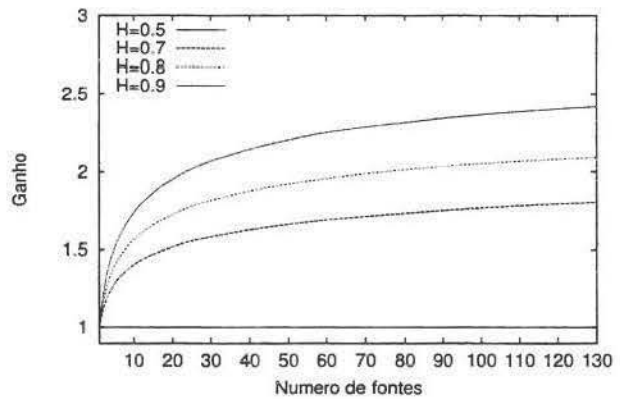


(c) Parâmetro Hurst = 0.90

Figura 6.1: A precisão da probabilidade de transbordo prevista pelas equações 6.9/6.10 para uma única fonte em função do tamanho do *buffer*.



(a) Fluxos com baixa variância ($\sigma = 0.10$)



(b) Fluxos com moderada variância ($\sigma = 0.55$)

Figura 6.2: Ganho com a multiplexação para fluxos com diferentes parâmetros Hurst.

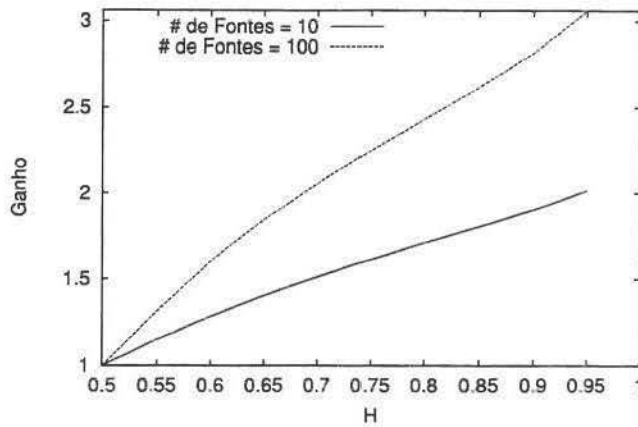


Figura 6.3: Ganho com a multiplexação em função do parâmetro Hurst para diferentes números de fontes.

Nos experimentos realizados, observou-se um ganho expressivo para fluxos com variâncias altas em relação a fluxos com variâncias baixas. O ganho com a multiplexação aumenta com o parâmetro Hurst, especialmente para fluxos com variâncias altas. Este comportamento é consequência das equações 6.9 e 6.10 levarem em consideração a existência de longos períodos nos quais não ocorrem chegadas nos fluxos com altos valores de parâmetro Hurst. Esses fluxos, conseqüentemente, demandam uma quantidade menor de banda passante quando várias fontes são multiplexadas, quando comparado a demanda de banda passante da abordagem que não considera a multiplexação estatística.

O ganho com a multiplexação estatística aumenta não só com a variância e parâmetro Hurst, mas também, com o número de fontes multiplexadas. A Figura 6.3 ilustra o ganho da multiplexação em função do parâmetro Hurst para 10 e 100 fontes. Observa-se claramente o benefício de se levar em consideração o MaxTS quando se computa a banda passante demandada, dado que o ganho aumenta com o número de fontes. Por exemplo, o ganho com a multiplexação para $H = 0.85$ é de 1.7 com 10 fontes, enquanto que com 100 fontes chega a 2.5. Observa-se também que para altos valores do parâmetro Hurst, a diferença entre a banda passante prevista pelas equações 6.9 e 6.10 e por uma aproximação que ignora a multiplexação estatística é maior. De fato, para 100 fontes nota-se que quando $H = 0.6$ o ganho com a multiplexação é de 1.5 enquanto que alcança-se uma ganho de 3.0 quando $H=0.95$.

Pode-se usar as equações 6.9/6.10 para derivar regiões de admissão em cenários compostos por fontes heterogêneas. Nas figuras 6.4.a e 6.4.b mostram-se as regiões de admissões para duas classes de fontes com probabilidade de transbordo de 10^{-6} e diferentes tamanhos de *buffers*. Nota-se que com o aumento da variância, diminui consideravel-

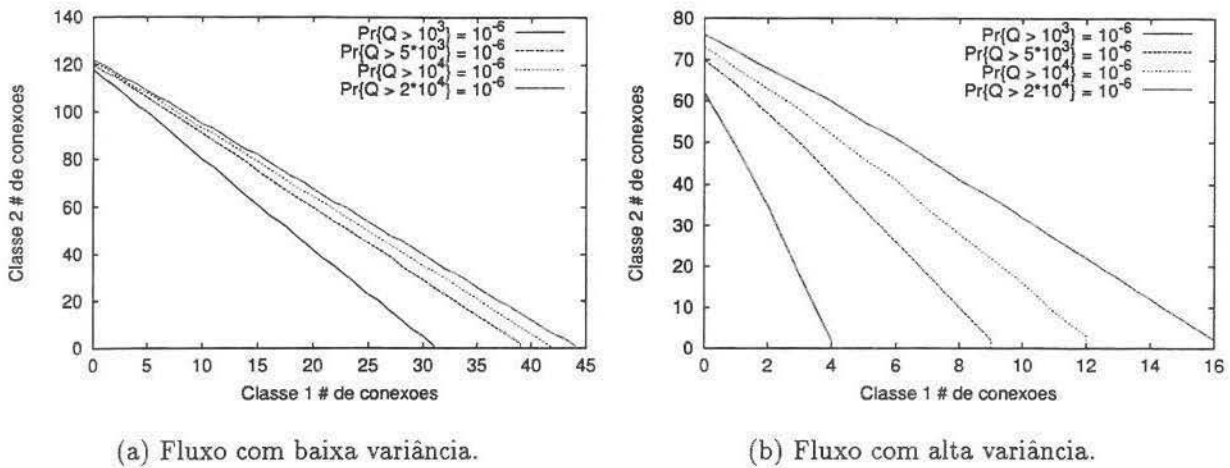


Figura 6.4: Região de admissão para classes de fontes heterogêneas.

mente o número de fontes aceitas. Além disso, o impacto da variância no número de fontes admitidas é maior quando o tamanho do *buffer* é pequeno.

Na figura 6.5 mostra-se a acurácia da probabilidade de transbordo computada pelas equações 6.9/6.10 em função do tamanho do *buffer*. Na figura 6.5.a considera-se 5 fontes com média e variância diferentes (tabela 6.1). Todas as fontes têm o mesmo parâmetro Hurst ($H = 0.86$). Na figura 6.5.b considera-se um agregado de fontes com diferentes parâmetros de tráfego (tabela 6.2). Como pode ser visto, a probabilidade de transbordo prevista pelas equações 6.9/6.10 está muito próxima dos resultados da simulação. Para grandes *buffers* os resultados tornam-se mais precisos. No caso de um *buffer* pequeno a diferença entre a probabilidade de transbordo dada pelo modelo analítico e pelo experimento de simulação é menor que uma ordem de magnitude.

Compara-se também as expressões de banda passante derivadas nesse capítulo com expressões de derivadas por Kelly [Kel96] e por Stathis e Maglaris [SM97]. Para comparar estas três abordagens, utiliza-se a Equação 3.35 em [Kel96] e a Equação 6 em [SM97]. Utilizam-se os parâmetros de fontes de tráfego reais descritas em [SM97]. Na Figura 6.6 mostra-se o número de conexões aceitas em função da probabilidade de transbordo do *buffer*. Considera-se nesta figura uma fonte com 1.4Mbits de taxa média, $\sigma = 0.28$, $H = 0.85$ e um *buffer* de 1000 células ATM. Pode-se ver que o número de fontes admitida prevista pelas equações 6.9 e 6.10 é igual ao número previsto pelo equação apresentada no trabalho de Kelly. Além disso, deve-se notar que se chega ao mesmo resultado previsto usando a teoria dos grandes desvios.

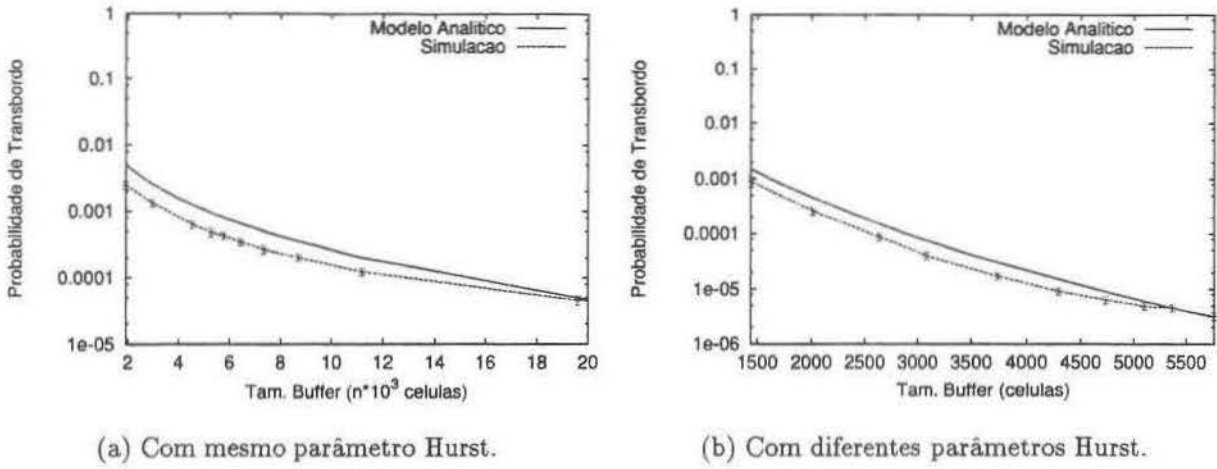


Figura 6.5: Probabilidade de transbordo X Tamanho do buffer para fontes heterogêneas.

Fonte	\bar{a}	σ	H
A	0.11	0.08	0.86
B	0.11	0.07	0.86
C	0.13	0.07	0.86
D	0.15	0.10	0.86
E	0.10	0.06	0.86

Tabela 6.1: Parâmetros de tráfego para a figura 6.5.a.

Fonte	\bar{a}	σ	H
A	0.13	0.10	0.63
B	0.11	0.07	0.72
C	0.13	0.07	0.78
D	0.12	0.07	0.86
E	0.11	0.04	0.90

Tabela 6.2: Parâmetros de tráfego para a figura 6.5.b.

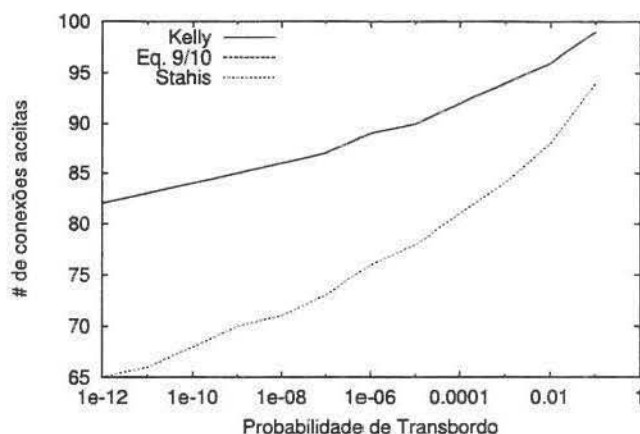


Figura 6.6: Comparação do número de fontes admitida da pelas equações 6.9/6.10 e o trabalho em [SM97] e [Kel96]. No gráfico a curva das equações 6.9/6.10 está sobreposta pela curva derivada a partir da equação 3.35 em [Kel96].

6.4 Algoritmo para Controle de Admissão de Conexões

Nessa seção um algoritmo para controle de admissão de conexões é descrito. Este algoritmo utiliza a resposta obtida pelas equações 6.9 e 6.10, na determinação da quantidade de banda passante necessária para atender os requisitos de qualidade de serviços dos pedidos de conexão. O algoritmo leva em consideração a existência de conexões ativas, e portanto, a admissão de novas conexões não deve causar nenhum prejuízo aos serviços das conexões ativas. Os pedidos de conexão são agrupados em classes caracterizadas pelos seguintes parâmetros: média (a), parâmetro Hurst (H), desvio padrão (σ), tempo médio entre chegadas (ρ) e o tempo médio de duração das conexões (μ). O desempenho do algoritmo é medido, utilizando-se um esquema de tarifação simples que avalia o volume de tráfego gerado e a duração da conexão, isto é,

$$Revenue = aT + bV \quad (6.11)$$

onde os valores a e b são tarifas para unidades de tempo e volume, selecionados pelo usuário de um conjunto de possíveis valores derivados pela rede a partir do contrato de tráfego. Assim, espera-se escolhas racionais por parte do usuário. Em [Kel96] esse esquema é proposto, e em [CSS97] avalia-se tal abordagem quanto a critérios de justiça.

Os pedidos de conexão que chegam num determinado tempo, irão competir pelos recursos disponíveis administrado pelo mecanismo de CAC. Assim, faz-se necessário o

estabelecimento de critérios para a aceitação das solicitações de conexão. A tarifação, descrita anteriormente, é usada como critério de escolha dos pedidos de conexão que serão aceitos pelo mecanismo. Na primeira implementação, estabelece-se prioridades entre as classes, e na segunda, determina-se que combinação de pedidos de conexão apresenta maior revenue.

Algoritmo de CAC com prioridades pré-estabelecida entre as classes - PRE. Nessa abordagem, as classes de conexão são ordenadas da seguinte forma: À classe C_i , que possui revenue R_i , atribui-se uma prioridade P_i , de tal forma que quanto maior o tarifa R_i maior será a prioridade da classe. Utiliza-se o cálculo do processo envelope de um mBf, a partir dos parâmetros que caracterizam cada classe, para estimar o volume de tráfego. O tempo de duração da conexão é substituído pelo tempo médio de duração das conexões. No conjunto τ estão os pedidos de conexão pendentes, avaliados em intervalos de tempo constante, Δt . Os pedidos são avaliados numa ordem decrescente de prioridade ($P_i > P_{i+1}$), definidas para as suas classes. Seguindo as prioridades, verifica-se a disponibilidade de recursos para a aceitação dos pedidos de conexão da classe C_i , em τ . Enquanto houver recurso disponível e pedidos de conexão de uma classe C_j qualquer, não avaliados no conjunto τ , a operação é repetida. Uma descrição algorítmica é apresentada a seguir.

```

j ← número de classes;
C ← descrição das classes;  $C_i = \{a_i, H_i, \sigma_i, \mu_i, \rho_i\}, i \in [1, j]$ .
 $\tau$  ← pedidos de conexões por classe;  $\tau = \{C_i, k\}$  k conexões pendentes da classe  $C_i$ .
S ← conexões ativas;  $S_i = \{C_i, m\}$  m conexões ativas da classe  $C_i$ .
P ← prioridades(C);  $P_i = \{C_h\}$ .
Para i=1 ate j faça
     $C_h \leftarrow P_i$ 
    Para l = 1 ate  $\tau\{C_h\}$  faça
        R ← RecursoDemandado( $S\{C_h, m + 1\}$ )
        Se  $R < \text{RecursoTotal}$ 
            AceitaPedido( $S\{C_h, m + 1\}$ );
        senão
            DescartaPedido();
    FimSe
FimPara
FimPara

```

Algoritmo de CAC baseado na escolha de pedidos de conexão com maior revenue - CAC. Nessa abordagem o conjunto τ , formado por n pedidos de conexão, é usado para derivar um conjunto τ' com $2^n - 1$ elementos. O elemento τ'_i representa

uma combinação composta por k pedidos de conexão dos n pedidos presente no conjunto τ . Para cada elemento τ'_i calcula-se a quantidade de recurso necessário para atender os requisitos de Qualidade de Serviço dos pedidos de conexão na combinação. Além disso, verifica-se dentre essas combinações aquela que apresenta um maior revenue. O revenue de um combinação τ'_i é calculado a partir da soma dos revenues de cada pedido de conexão através da equação 6.11, onde o tempo T , nesse caso, é igual a duração prevista para cada pedido em questão e não mais o tempo médio de duração atribuído à sua classe. A seguir, apresenta-se a descrição algorítmica dessa abordagem.

```

j ← número de classes;
τ ← pedidos de conexão por classe; τ = {Ci, k} k conexões pendentes da classe Ci.
n ← número de pedidos de conexões de todas as classes;
S ← conexões ativas; Si = {Ci, m} m conexões ativas da classe Ci.
τ' ← combinações(τ); τ'i combinação i com k pedidos de conexão.
t ← 0; melhor combinação.
MaiorRevenue ← 0;
Para i=1 ate 2n - 1 faça
    R = RecursoDemandado(S + τ'i);
    Se R < RecursoTotal
        Revenue ← RevenueCombinação(τ'i);
        Se Revenue > MaiorRevenue
            t ← i;
            MaiorRevenue ← Revenue;
        FimSe
    FimSe
FimPara
Se (t != 0)
    AceitaPedidos(τ't);
FimSe

```

O cenário simulado corresponde ao gerenciamento dos recursos de um multiplexador, por um mecanismo de CAC implementado através dos algoritmos descrito anteriormente. Mediu-se o desempenho de cada implementação pelo revenue produzido ao final dos experimento. O tráfego multiplexado são fontes de vídeo com características próprias que permite seu agrupamento em “classes”, tabela 6.3. Outras informações são os valores médio para o tempo de duração de uma conexão e o tempo entre as chegadas de pedidos de conexão. Valores característicos são 300 segundos e 0.4 segundos, respectivamente [Bol94, JDSZ97]. Os recursos gerenciados são a banda passante de uma linha que opera

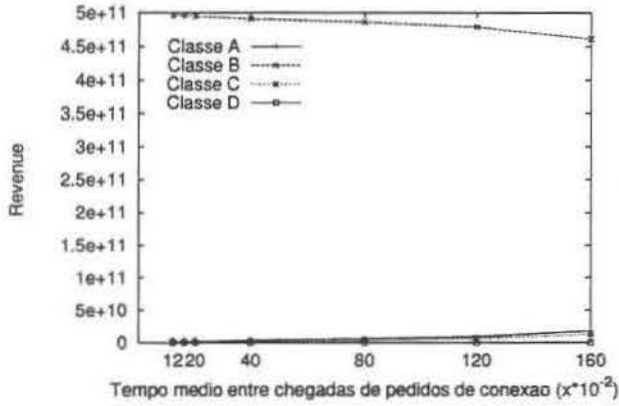
com uma utilização média de 70% e um *buffer* de 1000 células ATM. A política utilizada na administração dos recursos, a compartilhamento completo, possui a vantagem de ser facilmente implementada, entretanto ela é injusta [Ros97]. Assim, espera-se que uma classe seja beneficiada em detrimento às demais.

Classe	\bar{a}	σ	H
A	0.16	1.01	0.67
B	0.12	1.12	0.78
C	0.20	0.90	0.85
D	0.22	0.84	0.91

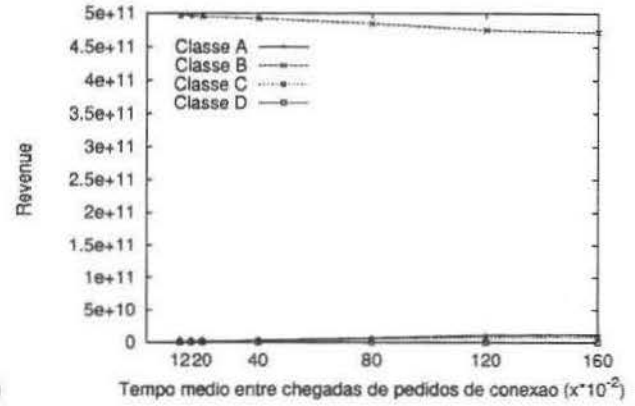
Tabela 6.3: Classes de conexão para as fontes de vídeo.

No primeiro experimento, avaliou-se o impacto da variação do tempo médio entre chegadas. À todas as classes, a cada repetição do experimento, atribuiu-se um valor definido no intervalo de 0.12 a 1.6 segundos, figura 6.7. Como destacado, anteriormente, ocorre predominância de uma classe, neste caso a classe B, sobre as demais, em ambos os experimentos, figuras 6.7.a e 6.7.b. Os valores de revenue apresentados por cada abordagem são bem próximos. Com a diminuição da taxa de chegada dos pedidos de conexão, ocorre também, uma diminuição no revenue produzido. A diferença de desempenho que aparece quando o tempo entre chegadas é 1.6 segundos é justificada pelo baixo volume de pedidos de conexão, uma vez que esse valor é uma ordem de grandeza maior que o valor característico. Assim, a abordagem OTM obtém vantagens da forma aleatória dos tempos de duração.

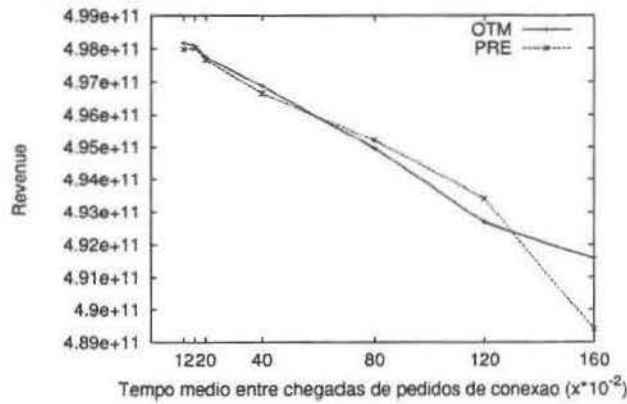
A segunda avaliação realizada estudou o impacto do tempo médio de duração da conexão no desempenho das abordagens. Os valores atribuídos para o tempo médio de duração das conexões variaram de 20 a 500 segundos. Os comentários sobre a predominância de uma classe, feito anteriormente, são transportados para essa avaliação, figura 6.8.a e 6.8.b. A forma crescente da curva na figura 6.8.c é justificada pelo período de tempo que as conexões permanecem ativas. Para tempos médios de duração da conexão menores que o tempo médio de duração característico a crescente é mais acentuada, tornando-se mais suave para tempos médios de duração maiores. No caso de tempos médios de duração da conexão menores, o que determina o aumento de revenue é a taxa de chegada de pedidos conexão. Altas taxas, nessa situação implicará em alto revenue.



(a) Pré-estabelece prioridades

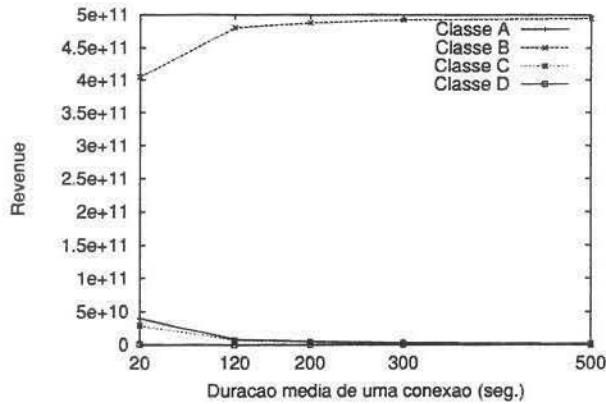


(b) Busca melhor combinação

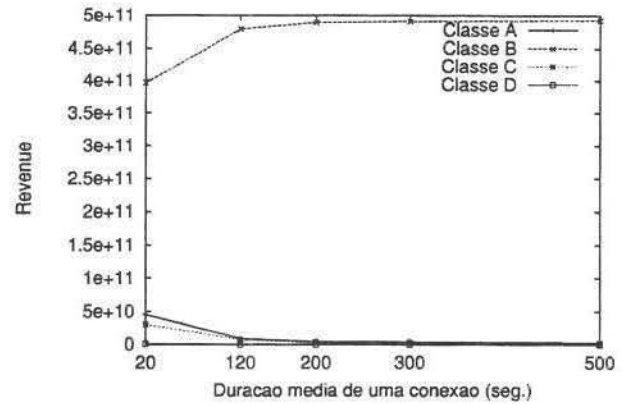


(c) Comparação das abordagens

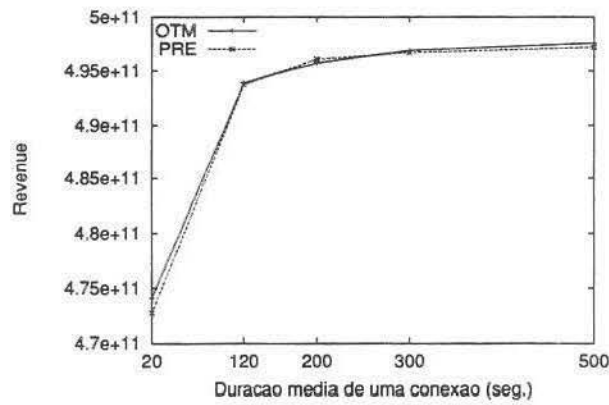
Figura 6.7: Avaliação do algoritmo de CAC para diferentes tempos médios entre chegadas de pedidos de conexão



(a) Pré-estabelece prioridades



(b) Busca melhor combinação



(c) Comparação das abordagens

Figura 6.8: Avaliação do algoritmo de CAC para diferentes tempos médios de duração de conexão

Capítulo 7

Policciamento de Fontes Auto-semelhantes

Após se admitir uma conexão é necessário monitorar o fluxo de células gerado por ela, garantindo que ele esteja de acordo com os descritores do tráfego declarados quando da solicitação do estabelecimento da conexão. Um mecanismo de policiamento ideal permite que células (pacotes) entrem na rede se e somente se a conexão é bem-comportada. Caso a fonte não esteja transmitindo de acordo com o contrato, ele deve descartar as células que chegam ou marca-las com baixa prioridade. Nesse capítulo, discute-se um novo mecanismo de policiamento para fontes que geram tráfego com dependência de longa duração (*Long Range Dependence* - LRD), o Balde Furado Fractal. Compara-se o seu desempenho, no policiamento desse tipo tráfego, com o mecanismo Balde Furado. A seção 7.1 descreve o mecanismo Balde Furado e a razão pela qual o mecanismo não apresenta desempenho satisfatório no policiamento de fontes auto-semelhantes. Apresenta-se também um método para ajustar os parâmetros do mecanismo sem a necessidade de resolver um sistema de filas. Na seção 7.2 apresenta-se um novo mecanismo para policiar fontes, chamado de Balde Furado Fractal, destacando-se a sua habilidade em reconhecer dependência de longa duração. Os resultados da avaliação feita nos dois mecanismos é apresentada na seção 7.3.

7.1 O Balde Furado

O mecanismo Balde Furado (BF) usa dois parâmetros para controlar a transmissão de uma conexão: a taxa de geração de permissões e o tamanho do balde. É sabido que policiar fontes que transmitem seus fluxos em rajadas usando somente esses dois parâmetros é muito difícil. Caso a taxa de geração de permissões esteja próxima da taxa média de transmissão da fonte pode-se descartar ou marcar células de fontes bem-

comportadas. Caso a taxa de geração de permissões esteja próxima a taxa de pico, perde-se as vantagens da multiplexação estatística. Além disso, se o tamanho do balde é redimensionado para um valor maior pode-se permitir que longas rajadas entrem na rede [Rat91, KM94b, JSS96]. Uma maneira de definir os parâmetros do balde furado é resolver uma fila $G/D/1/k$. Dependendo do processo de chegada, a escolha dos parâmetros do Balde Furado pode ser um problema complexo. Em virtude dessa dificuldade, propõe-se um cálculo baseado no processo envelope mBf para definir os parâmetros do BF sem ter que resolver um sistema de filas.

O BF pode ser visto como um regulador de tráfego com saída dada pelo processo $L(t)$ de forma que $L(t) \leq Rt + S$ onde R é a taxa média de transmissão e S é a rajada máxima permitida. Em resumo, num intervalo de tempo de duração t o BF aceita no máximo $(Rt + S)$ células. $L(t)$ pode ser visto também como um processo envelope determinístico, i.e., ele define o número máximo de células que uma fonte pode enviar em qualquer intervalo de tempo de duração t .

Define-se $A_H(t)$ e $\hat{A}_H(t)$ como processo de chegada acumulado, i.e., o número total de células que chegam acumulado até um tempo t , e seu processo envelope probabilístico até o tempo t , respectivamente. De forma a minimizar a probabilidade de se descartar células incorretamente, deve-se ter $L(t) \geq A_H(t) \forall t > 0$. Conseqüentemente, como a fonte é bem-comportada, o BF não deve marcar qualquer célula gerada pela fonte. Assumindo-se que a probabilidade da fonte exceder seu processo envelope probabilístico é negligenciável, pode-se escrever $A_H(t) \leq \hat{A}_H(t), \forall t > 0$. Assim, tem-se que:

$$L(t) \geq \hat{A}_H(t), \forall t > 0 \quad (7.1)$$

uma vez que o processo envelope probabilístico é um limite mais estrito que o processo envelope determinístico $L(t)$. Substituindo-se a fórmula do processo envelope mBf na equação 7.1, pode-se escrever:

$$\bar{a}t + \kappa\sigma t^H \leq Rt + S$$

obtem-se então:

$$t(\bar{a} - R) + \kappa\sigma t^H - S \leq 0 \quad (7.2)$$

Para derivar os parâmetros R e S do BF, calcula-se o t^* que maximiza a equação 7.2:

$$t^* = \left(\frac{\kappa\sigma H}{R - \bar{a}} \right)^{\frac{1}{1-H}}$$

Substituindo-se t^* na equação 7.2 tem-se que:

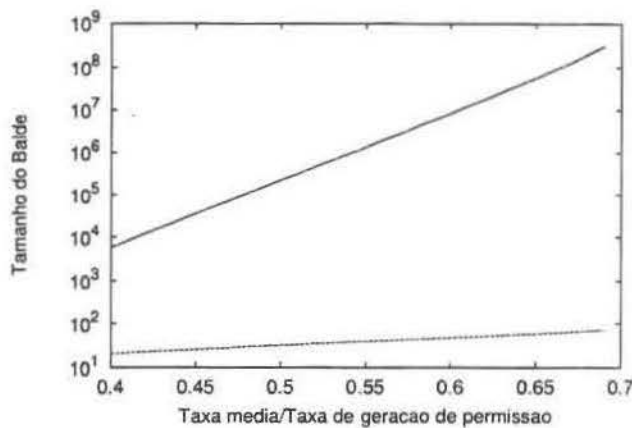


Figura 7.1: Tamanho do balde x a razão entre a taxa média de chegada e a taxa de geração de permissões.

$$(\bar{a} - R) \left[\frac{\kappa \sigma H}{R - \bar{a}} \right]^{\frac{1}{(1-H)}} + \kappa \sigma \left[\frac{\kappa \sigma H}{R - \bar{a}} \right]^{\frac{H}{(1-H)}} - S \leq 0 \quad (7.3)$$

usando a equação 7.3 pode-se computar um valor R para um certo valor de S ou vice-versa. No caso de um processo movimento browniano que tem chegadas independente e identicamente distribuídas, a equação 7.3 degenera em uma equação quadrática simples. Para o caso geral, pode-se resolver a equação através de métodos numéricos.

Na Figura 7.1 mostra-se o tamanho do balde necessário em função da taxa média de chegada da fonte normalizada pela taxa de geração de permissões. A linha pontilhada corresponde a $H = 0.5$ enquanto que a linha sólida corresponde a $H = 0.9$. Nota-se que para altos valores do parâmetro Hurst se R está próximo da taxa média, o valor atribuído a S é proibitivo.

O problema do mecanismo BF, é que ele assume que o processo de chegada agregado comporta-se como uma função linear no tempo. Isso não é verdade, nem mesmo para o caso do processo mB comum. Por exemplo, o processo envelope mB com média \bar{a} e desvio padrão σ é dado por $\hat{A}(t) = \bar{a}t + \kappa\sigma t^{1/2}$.

Nota-se que o tráfego agregado é uma função não linear no tempo uma vez que seu processo envelope tem um termo adicional $t^{1/2}$. Por essa razão, é extremamente difícil ajustar os parâmetros do BF para fontes que operam em rajadas, i.e., quando o desvio padrão é grande. Assim sendo, tem-se que escolher S de forma a absorver o termo $t^{1/2}$. Pode-se, alternativamente, escolher uma taxa de geração de permissões alta, de forma a absorver ambos os termos $\bar{a}t + \kappa\sigma t^{1/2}$. No caso de fontes com dependência de longa duração, o problema é ainda mais complexo, uma vez que a variância aumenta com t^{2H} ,

onde $2H > 1$. Afirma-se, portanto, que o BF não é um processo envelope adequado quando a fonte tem grande variabilidade.

7.2 O Balde Furado Fractal

Propõe-se um novo regulador de tráfego baseado numa melhor caracterização do tráfego de entrada chamado de Balde Furado Fractal (BFF). O trabalho total aceito pelo BFF é dado por:

$$\hat{L}(t) = \bar{a}t + \Psi t^H + S, \quad (7.4)$$

onde \bar{a} é a taxa média de chegada da fonte. Ψ é dado por $\kappa\sigma$, κ é uma constante e σ é o desvio padrão das chegadas geradas pela fonte.

O BFF opera da seguinte forma: define-se uma janela de tempo com duração de τ unidades de tempo. Verifica-se se o número de chegadas durante essa janela de tempo excede o valor médio declarado ($\bar{a}\tau$). Caso isto ocorra, compara-se o número acumulado de chegadas durante esta janela de tempo com o número de células do processo envelope BFF durante o mesmo período (equação 7.4). Se o número de chegadas ultrapassar o número de células permitido pelo processo envelope BFF, marca-se todas as células em excesso. Aumenta-se então a janela de tempo em τ unidades de tempo, i.e., considera-se uma janela de tempo de duração 2τ . Essa nova janela começa no tempo em que o processo de chegada violou a taxa média declarada. Compara-se novamente o número acumulado de chegadas durante essa janela de tempo, de duração 2τ , com o número médio de chegadas permitido. Se o número de células ultrapassar o número médio permitido de chegadas, compara-se este valor com o número permitido de chegadas pelo processo envelope BFF e, marca-se o número de células excedente menos as marcadas na janela anterior. Enquanto o número de chegadas ultrapassar o valor médio declarado, continua-se aumentando o tamanho da janela de tempo em τ unidades. Sempre que o número médio de chegadas operar abaixo do valor declarado, reinicializa-se a janela de tempo para τ unidades de tempo e continua-se verificando a taxa média de chegada.

Uma descrição matemática da dinâmica do BFF é a seguinte. Define-se $C(\tilde{t} + n\tau)$ como o número acumulado de células que chegam durante o intervalo $[\tilde{t}, \tilde{t} + n\tau]$ ou seja $C(\tilde{t} + n\tau) = A(\tilde{t} + n\tau) - A(\tilde{t})$ onde $A(\tilde{t})$ é o número de chegadas até o tempo \tilde{t} . Caso $C(\tilde{t} + n\tau)$ exceda o número médio de chegadas durante o intervalo $n\tau$ ($\bar{a}n\tau$), verifica-se se o número de chegadas excede o número de chegadas permitido pelo processo envelope $\lambda(\tilde{t} + n\tau)$ onde $\lambda(\tilde{t} + n\tau) = \hat{L}(\tilde{t} + n\tau) - \hat{L}(\tilde{t})$. Se esse valor é ultrapassado, marca-se o número de células excedente não marcados na janela anterior, i.e., marca-se $C(\tilde{t} + n\tau) - \lambda(\tilde{t} + n\tau) - C(\tilde{t} + (n-1)\tau) + \lambda(\tilde{t} + (n-1)\tau)$.

Aumenta-se a janela de tempo em τ unidades de tempo e repete-se o processo. Uma vez que, a taxa média de chegadas começa a operar abaixo do valor declarado \bar{a} , diminui-se

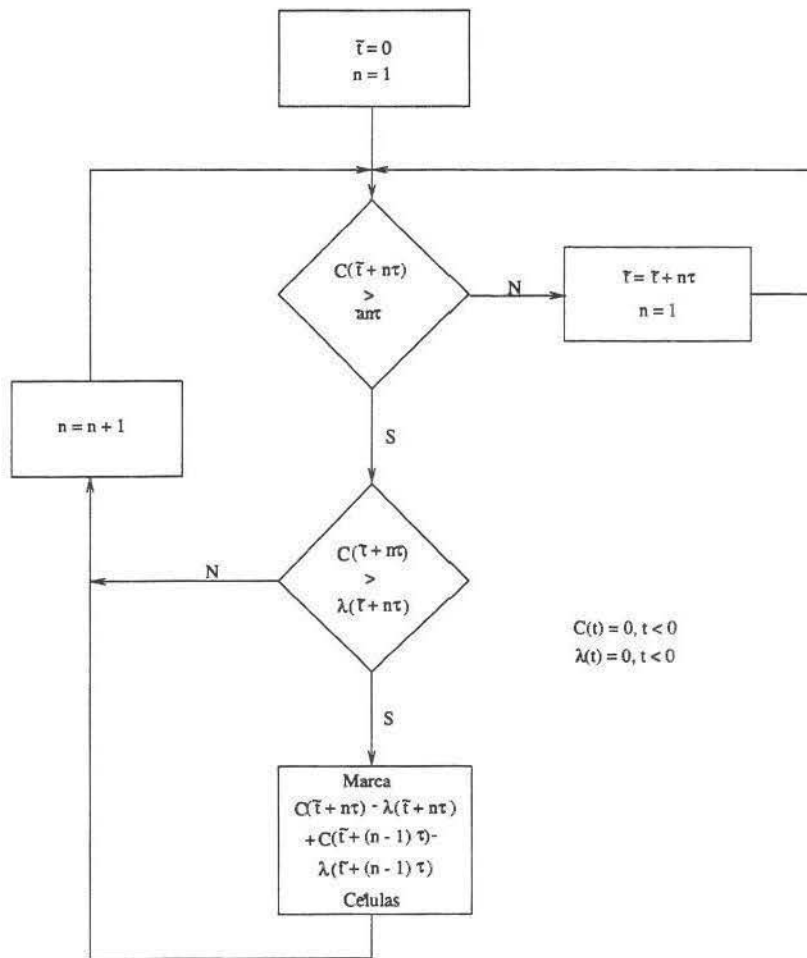


Figura 7.2: O Balde Furado Fractal.

a janela de tempo para o valor de τ unidades de tempo (Figura 7.2).

7.3 Resultados Numéricos

Simulou-se o mecanismo BFF e o BF com o objetivo de comparar seus desempenhos. Nesse primeiro exemplo, usou-se uma seqüência de 20 minutos do vídeo *STAR WARS* como tráfego de entrada. Essa seqüência MPEG apresenta LRD e pode ser caracterizada por um processo envelope mBf. Os parâmetros, \bar{a} e σ^2 , foram obtidos a partir da média e variância da seqüência. Escolheu-se $\kappa = \sqrt{-2 \ln(10^{-6})} = 5.25$ e $H = 0.90$, de forma que o BFF esteja de acordo com o processo envelope da seqüência. Aumentou-se a taxa média de chegada k vezes em seu valor nominal e computou-se a nova probabilidade de violação (PV) para $k \in [1.5 \dots 3]$. Nesse caso, em que a fonte é mal-comportada, a expectativa é

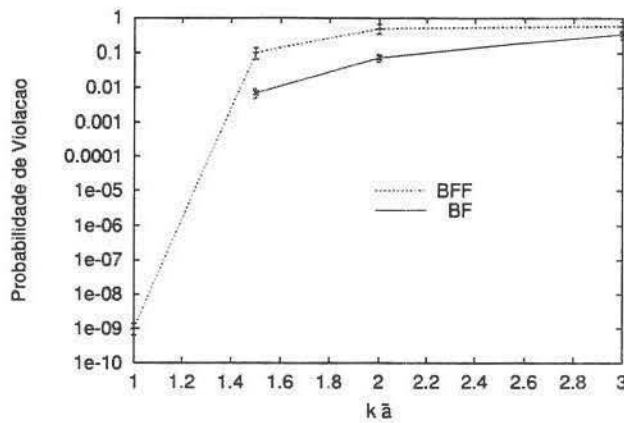


Figura 7.3: Uma comparação entre o BFF e o BF para fontes violando o contrato baseado numa seqüência de fluxo de vídeo do filme *STAR WARS*

que o mecanismo de controle de fluxo deve alcançar uma PV alta. Escolheu-se a taxa de geração de permissão e o tamanho do balde do BF de forma que ambos os mecanismos, tenham a mesma PV na taxa média nominal. A Figura 7.3 mostra que o BFF alcança uma PV maior que a apresentada pelo BF. Pode-se ver que apesar da pequena PV acusada pelo BFF quando a fonte opera na taxa nominal, esse valor torna-se expressivo, acima de 10%, se a fonte viola em 1.5 a sua taxa média nominal.

A Figura 7.4 compara a habilidade do BFF com a habilidade do BF em monitorar fontes bem-comportadas. Nos experimentos de simulação usou-se os procedimentos descritos por Mandelbrot [Man69] para gerar um trace mBf e o método de replicação independente para derivar intervalo de confiança com 99% de grau de confiança. Na Figura 7.4.a mostra-se a probabilidade de violação em função da taxa média para um processo mBf com $H = 0.8$, $\sigma^2 = 1.0$. Para derivar o BFF utilizou-se $\epsilon = 10^{-3}$, i.e., permite-se violação no número de chegadas de no máximo 0.1% do total de chegadas. Pode-se notar que o BF acusa alta probabilidade de violação mesmo com altos valores atribuído à taxa de geração de permissões. Pode-se observar que o BF com parâmetros definidos de acordo com o procedimento descrito nessa seção fornece valores de PV que são aproximadamente da mesma ordem de magnitude da PV do processo envelope. Por outro lado, o BFF fornece uma PV que é diversas ordens de grandeza menor que a PV do BF. Ou seja, o BFF é capaz de monitorar acuradamente um processo mBf. É importante mencionar que para processos SRD e tamanho de balde grande a probabilidade de violação é sensível às altas taxas de geração de permissão. Contudo, isso não é verdade para processos que apresentam LRD uma vez que o BF não é capaz de ajustar-se às longas rajadas existentes nestes processos. Observou-se o mesmo tipo de comportamento ao se variar o parâmetro

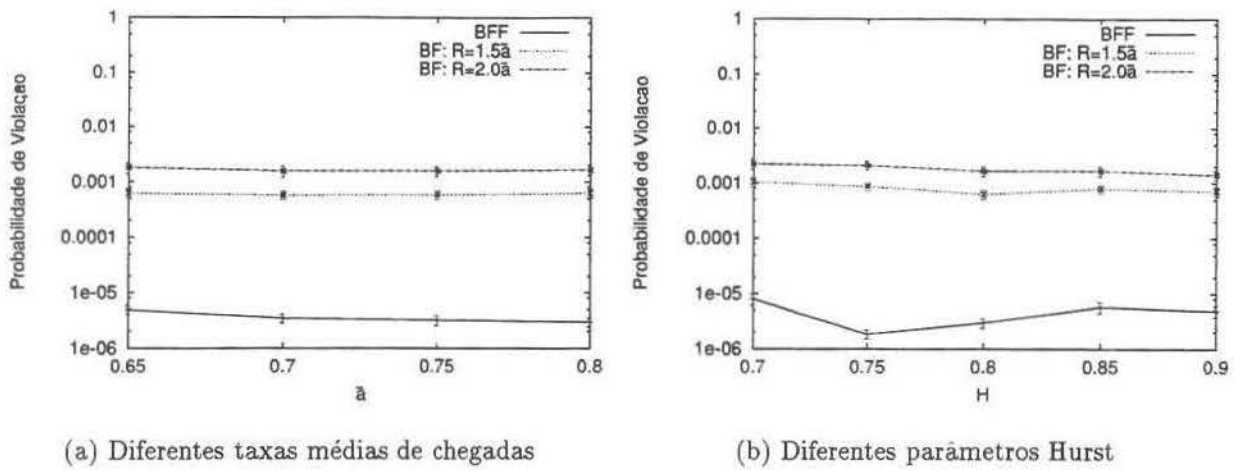


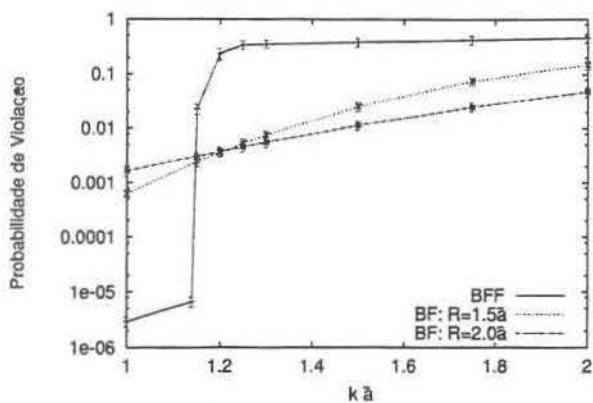
Figura 7.4: Uma comparação entre a probabilidade de violação apresentada pelo BFF e a probabilidade de violação apresentada pelo BF.

H (Figura 7.4.b).

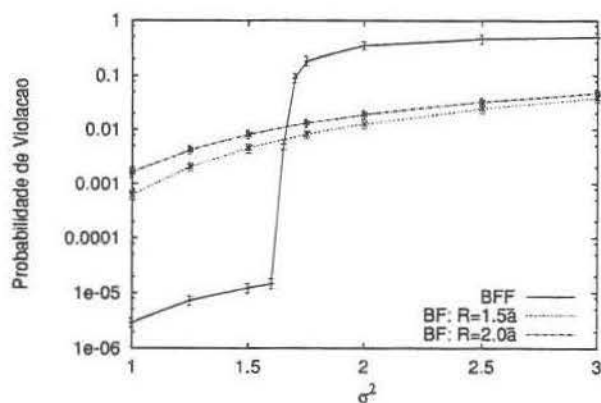
Na Figura 7.5 aumenta-se a taxa média de chegada em k vezes o valor original do processo. Em um mecanismo ideal de policiamento a PV passaria de um valor muito baixo para 1 logo que a fonte começasse a transmitir com taxa média de chegada acima do valor declarado. Nota-se que a PV dada pelo BFF segue um padrão que é similar ao comportamento do mecanismo ideal, exceto que ele alcança a violação com uma taxa de 1.15 maior que o valor original. Essa reação a violação para um mecanismo não ideal é bastante aceitável. Em contra-partida, o BF não faz distinção significativa entre uma fonte em violação e uma fonte bem-comportada. Além disso, a PV do BF não tende para 1 apesar das altas taxas de chegada.

Na Figura 7.5.b manteve-se $\bar{a} = 0.8$, $H = 0.8$ e aumentou-se a variância a partir de seu valor nominal. Novamente, percebe-se que o BF não diferencia as fontes que violam o contrato das fontes que operam de acordo com o contrato. Além disso, a probabilidade de violação do BF não tende para 1 apesar dos altos valores da variância. Ao contrário, o BFF penaliza as fontes que violam o contrato.

O mecanismo BFF compara de tempos em tempos o número acumulado de chegadas com o número de chegadas permitido pelo processo envelope BFF quando a taxa de chegada excede seu valor médio. Investigou-se a sensibilidade dos resultados com relação ao tamanho da janela de tempo (τ) durante a qual verifica-se o número de chegadas, i.e., o tempo decorrido entre duas comparações. Consideram-se janelas de tempo 10, 100 e 1000 vezes maior que a usada nos exemplos anteriores, i.e., n vezes maior que o tempo de transmissão de uma célula. Nota-se que os resultados não são dependentes da duração da janela de tempo. A Figura 7.6 mostra a PV como função da taxa de chegada e como

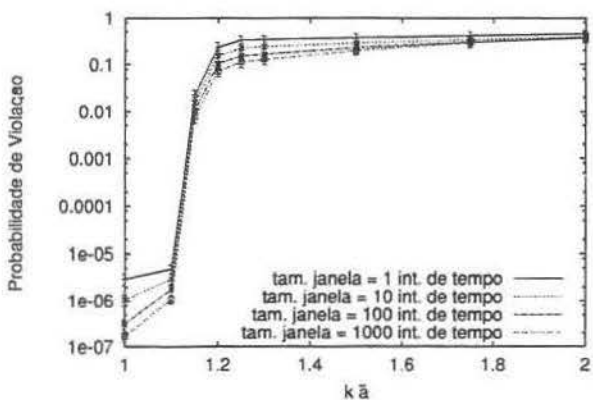


(a) Violando taxa média de chegadas

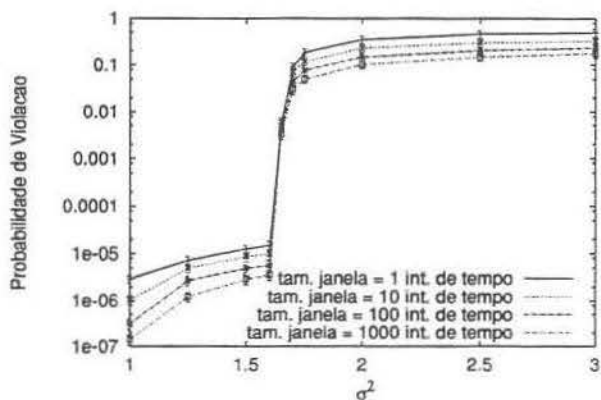


(b) Violando variância

Figura 7.5: Probabilidade de violação apresentada pelo BFF para fontes mal-comportadas.



(a) Violando taxa média de chegadas



(b) Violando variância

Figura 7.6: Sensibilidade da Probabilidade de Violação do BFF para durações diferentes da janela de tempo.

função da variância para diferentes tamanhos de janela de tempo. Tais achados reforçam a robustez do BFF.

Capítulo 8

Conclusões

A utilização de um novo paradigma para a modelagem do tráfego de redes de alta velocidade, definido a partir de modelos auto-semelhantes, tem como consequência a necessidade de mecanismos de controle de tráfego que considerem um alto grau de variabilidade do tráfego, bem como a componente de dependência de longa duração. Nessa dissertação, mecanismos de controle de tráfego foram avaliados sob a luz desse novo paradigma. Estudou-se um novo mecanismo de policiamento, e avaliou-se um conjunto de equações capazes de estimar a quantidade de recursos demandado por um agregado de fontes auto-semelhantes a serem multiplexadas num canal.

A multiplexação de tráfego propicia uma melhor utilização dos recursos de uma rede. Avaliou-se um conjunto de equações capazes de estimar a quantidade de recursos demandados por um agregado de fontes que têm um certo requisito de qualidade de serviço. Utilizou-se o conceito de escala de tempo de interesse, MaxTS . Mostrou-se que quando um agregado de fontes é avaliado levando-se em consideração esse conceito, a quantidade de recursos utilizado é bem menor que a requerida por uma abordagem que desconsidera o conceito. Mostrou-se que esse ganho está relacionado com a variabilidade do tráfego, característica que é considerada pelas equações, e com o número de fontes multiplexadas. Além disso, mostrou-se que essas equações são capazes de derivar regiões de admissão para classes de fontes heterogêneas. Comparou-se o desempenho desse conjunto de equações com as equações de uma abordagem que utiliza a teoria dos grandes desvios. Mostrou-se que os resultados obtidos pela abordagem apresentada nessa dissertação e a abordagem que utiliza a teoria dos grandes desvios são muito parecidos.

Avaliou-se o mecanismo de policiamento Balde Furado quando o fluxo a ser policiado apresenta alta variabilidade. Mostrou-se que em função do reduzido número de parâmetros e do tipo de informação que é considerada, o mecanismo é incapaz de policiar, com precisão, o fluxo gerado por uma fonte. Um novo mecanismo de policiamento, chamado Balde Furado Fractal, foi proposto. Esse mecanismo baseia-se num processo envelope do

movimento Browniano Fractal. As avaliações realizadas mostraram que esse mecanismo é capaz de policiar, com precisão, tráfego auto-semelhante estando a fonte violando ou não o contrato. O comportamento apresentado, em relação a tempo de reação, quando o tráfego gerado pela fonte viola o contrato, é muito próximo do comportamento de um mecanismo ideal. O mecanismo, também, mostrou-se insensível à frequência com que o tráfego gerado pela fonte é avaliado.

8.1 Trabalhos futuros

Propõe-se como trabalhos futuros o estudo de um mecanismo Balde Furado Fractal com *buffer*, bem como a integração deste com mecanismos de controle de fluxo ABR.

Balde Furado Fractal com *Buffer* - Em [BW94] avaliou-se o comportamento do processo de saída de um mecanismo Balde Furado com *buffer*. Esse *buffer* tem tamanho constante definido por $S_T + S_C$, onde S_T é o tamanho do balde e S_C é o tamanho do *buffer* de célula. Mostrou-se que o *buffer* de célula “suaviza” o processo de saída do BF (coeficiente de variação) em uma escala de tempo curta. Numa escala de tempo longa, a influência do *buffer* não é significativa. No entanto, quando ocorre multiplexação de um grande número de processos de saída, o *buffer* de célula pode estabelecer dramática “suavização” ao fluxo que parte pela linha. Essa mesma avaliação pode ser realizada utilizando-se o Balde Furado Fractal, e responder questões como: *Qual o impacto que a adição do buffer pode causar na correlação do processo de entrada quando este passa pelo mecanismo? Ocorre alteração no grau de explosividade do processo quando ele passa por um Balde Furado Fractal com Buffer?*

Policimento de Fluxo ABR - Um controle de taxa baseado em “retorno” capaz de suportar o serviço ABR vem sendo definido. Essa abordagem consiste no estabelecimento, pela rede, em intervalos periódicos, da taxa máxima em que as fontes podem operar. Para que o esquema seja bem sucedido, ou seja, a rede apresente alta utilização e ocorra divisão justa dos recursos, é imprescindível que as fontes não ultrapassem a taxa máxima permitida. Para garantir que os objetivos do serviço ABR sejam alcançados, especialmente em redes públicas, é essencial policiar as fontes e penalizar aquelas que violam o contrato. Em [KRM98], avaliou-se o comportamento de três algoritmos de policiamento, que são variações do Balde Furado, quando a fonte policiada gera seu tráfego de acordo com o serviço ABR. Avaliou-se a vazão da rede e critérios de justiça na divisão dos recursos num ambiente onde as fontes operam em conformidade e, também, violam o contrato de tráfego. Acredita-se que uma variação do mecanismo Balde Furado Fractal possa ser utilizado por um controlador de fluxo do serviço ABR. A manipulação de t , o tamanho da janela, permitirá controlar o volume de tráfego que a fonte pode transmitir.

Bibliografia

- [AEN95] R. G. Addie, A. Elwalid, and T. Neame. Fractal Traffic: Measurements, Modelling and Performance Evaluation. *in Proc. of IEEE INFOCOM*, 1995. Pg. 977-984.
- [AGS93] H. Ahmadi, R. Guerin, and K. Sohraby. Analysis of a Rate-Based Access Control Mechanism for High-Speed Networks. *IEEE Transaction on Communications*, 41:940–950, June 1993.
- [AK94] V. Anantharam and T. Konstantopoulos. Burst Reduction Properties of the Leaky Bucket Flow Control Scheme in ATM Networks. *IEEE Transactions on Communications*, December 1994.
- [AMS82] D. Anick, D. Mitra, and M. M. Sondhi. Stochastic Theory of a Data-Handling System with Multiple Sources. *Bell System Tech. J.*, 61:1871–1894, 1982.
- [And95] A. T. Andersen. *Modelling of Packet Traffic with Matrix Analytic Method*. PhD thesis, Institute of Mathematical Modelling, Technical University of Denmark, 1995.
- [AZ] R. G. Addie and M. Zukerman. An Approximation for Performance Evaluation of Stationary Single Server Queues. *IEEE Transactions on Communications*, 42(12):3150–3160.
- [AZ94a] R. G. Addie and M. Zukerman. Performance Evaluation of Stationary Single Server Autoregressive Queue. *Australian Telecommunications Research*, 28(1):25–32, 1994.
- [AZ94b] R. G. Addie and M. Zukerman. Queues with total recall - application to the B-ISDN. In *ITC 14*, North Holland, 1994.
- [AZN95] R. G. Addie, M. Zukerman, and T. Neame. Fractal Traffic: Measurements, Modelling and Performance Evaluation. In *IEEE INFOCOM'95*, Boston, April 1995.

- [BCT91] M. Butto, E. Cavallero, and A. Tonietti. Effectiveness of the leaky bucket policing mechanism in ATM networks. *IEEE Journal on Selected Areas in Communications*, 9:335–342, April 1991.
- [BD95] D. Botvich and N. Duffield. Large Deviations, the Shape of Loss Curve, and Economies of Scale in Large Multiplexers. *Queueing Systems*, 20:293–320, 1995.
- [Ber92] J. Beran. Statistical Methods for Data with Long-Range Dependence. *Statistical Science*, 7:404–427, 1992.
- [BJ76] G. E. P. Box and G. M. Jenkins. *Time Series Analysis: Forecasting and Control*. 1976.
- [Bol94] Vladimir A. Bolotin. Modeling Call Holding Time Distributions for CCS Network Design and Performance Analysis. *JSAC*, 12(3):433–438, April 1994.
- [Bou91] J. Boundec. An Efficient Solution Method for Markov Models of ATM Links with Long-Range Dependence. *IEEE JSAC*, 9(3):408–417, April 1991.
- [BSTW95] J. Beran, R. Sherman, M. S. Taqqu, and W. Willinger. Long-Range Dependence in Variable-Bit-Rate Video Traffic. *IEEE Transaction on Communications*, 43(2/3/4):1566–1579, 1995.
- [BW92] A.W. Berger and W. Whitt. Traffic Shaping by a Job Buffer in a Token-Bank Rate-Control Throttle. *Communications in Statistics: Stochastic Models*, 8:685–717, 1992.
- [BW94] A.W. Berger and W. Whitt. Pros and Cons of a Job Buffer in a Token-Bank Rate-Control Throttle. *IEEE Transactions on Communications*, 42((2/3/4)), April 1994.
- [CB95] Mark E. Crovella and Azer Bestavros. Explaining World Wide Web Traffic Self-Similarity. *Technical Report TR-95-015*, October 1995.
- [CFZ98] R. Choen, N.L.S. Fonseca, and M. Zukerman. Traffic Management and Network Dimensioning. In M. Tatipamula and B. Kasnabish, editors, *Multimedia Communications Networks: Technologies and Services*. Artech House, 1998.
- [Cha94] C. Chang. Stability, Queue Length, and Delay of Deterministic and Stochastic Queueing Networks. *IEEE Transactions on Automatic Control*, 39(5):913–931, May 1994.

- [CNY73] M. Chi, E. Neal, and G. Young. Practical Applications of Fractional Brownian Motion and Noise to Synthetic Hydrology. *Water Resources Research*, 9:1523–1533, December 1973.
- [Cox84] D. R. Cox. *Long-range dependence: A review*. Ames, IA: Iowa State Univ. Press., 1984. pp. 55-74.
- [Cru91] R. Cruz. A Calculus for Network Delay, part i: Network Elements in Isolation. *IEEE Transaction of Information Theory*, 37(1):114–121, January 1991.
- [CSS97] C. Courcoubetis, V. A. Siris, and G. D. Stamoulis. Comparing Usage-Based Pricing Schemes for Broadband Network. In *IEE Colloquium on "Charging for ATM - the Reality Arrives"*, London, UK, November 1997.
- [CW95] C. Courcoubetis and R. Weber. Effective Bandwidth for Stationary Sources. *Probability in Eng. and Info. Sci.*, 9(2):285–94, 1995.
- [Dah89] R. Dahlhaus. Efficient Parameter Estimation for Self-Similar Processes. *Annals of Statistics*, 17:1749–1766, 1989.
- [DLO⁺95] N. Duffield, J.T. Lewis, N. O'Connell, R. Russel, and F. Toomey. Predicting Quality of Service with Long-range Fluctuations. *Proc of IEEE ICC'95*, pages 473–477, 1995.
- [DO95] N. Duffield and N. O'Connell. Large Deviations and Overflow Probabilities for the General Single Server Queue, with Application. In *Cambridge Phil Soc.*, volume 118, pages 363–74, 1995.
- [ea91] I. Norro et al. The Superposition of Variable Bit Rate Sources in an ATM Multiplexer. *IEEE JSAC*, 9(3):378–387, April 1991.
- [eAM95] A. Adas e A. Mukherjee. On Resource Managemen and QoS Guarantees for Long Range Dependent traffic. *Proc. IEEE INFOCOM*, 1995. Pg. 779-787.
- [E.K96] E.Knightly. H-bind: A New Approach to Providing Statistic Guarantees to VBR Traffic. In *INFOCOM'96*, volume 1, pages 1091–1099, San Francisco, CA, March 1996.
- [EM93] A. I. Elwalid and D. Mitra. Effective bandwidth of General Markovian Traffic Sources and Admission Control of High Speed Networks. *Transactions on Networking*, 1(4):329–343, June 1993.

- [ENW96] A. Erramilli, O. Narayan, and W. Willinger. Experimental queueing analysis with long-range dependence packet traffic. *IEEE/ACM Transactions on Networking*, April 1996.
- [FF98] N. L. Fonseca and M. J. Ferreira. Multiple Class Selective Discard Under a Long-Range Dependent Process. *Proc of IEEE International Telecommunications Symposium 98*, pages 95–101, 1998.
- [Fon94] N. L. Fonseca. *An Aproximate Model for the Output Process of an ATM Multiplexer with Selective Discard Mechanism*. PhD thesis, USC Department of Computer Engineering, 1994.
- [FT86] R. Fox and M. S. Taquq. Large Sample Properties of Parameter Estimates for Strongly Stationary Gaussian Time Series. *Ann. Statist.*, 14:517–532, 1986.
- [FV90] D. Ferrari and D. Verma. A Scheme for Real-Time Channel Establishment in Wide-Area Networks. *IEEE Journal on Selected Areas in Communications*, 8(3):368–379, April 1990.
- [GAN91] R. Guerin, H. Ahmadi, and M. Naghshineh. Equivalent Capacity and Its Applications to Bandwidth Allocation in High-Speed Networks. *IEEE JSAC*, 9:968–81, September 1991.
- [GW94] M. Garrett and W. Willinger. Analysis Modeling and Generation of Self-Similar VBR Video Traffic. In *ACM SIGCOMM*, pages 264–279, September 1994.
- [HDLK95a] C. Huang, M. Devetikiotis, I. Lambadaris, and A. R. Kaye. Fast Simulation for Self-Similar Traffic in ATM Networks. *IEEE ICC*, June 1995. Pg. 438-444.
- [HDLK95b] C. Huang, M. Devetsikiotis, I. Lambadaris, and A. Kaye. Modeling and Simulation of Self-similar Modeling of Variable Bit Rate Compressed Video: A Unified Approach. *Proc. ACM SIGCOM*, 1995. Pg. 114-125.
- [HL86] H. Heffes and D. M. Lucantoni. A Markov Modulated Characterization of Packetized Voice and Data Traffic and Related Statistical Multiplexer Performance. *IEEE J. Select. Areas Commun.*, 4:858–868, 1986.
- [Hos84] J. R. Hosking. Modeling Persistence in Hydrological Time Series Using Fractional Differencing. *Water Resources Research*, 20(12):1898–1908, 1984.

- [HRS96] D. Heath, S. Resnick, and G. Samorodnitsky. Heavy Tails and Long-Range Dependence in On/Off Processes and Associated Fluid Models. *Preprint*, 1996. School of ORIE, Cornell University, Ithaca, NY, 1996.
- [HY96] M. G. Hluchyj and N. Yin. A Second-Order Leaky Bucket Algorithm to Guarantee QoS in ATM Networks. *Proc of IEEE GLOBECOM'96*, pages 1090–1094, 1996.
- [Jaf81] J. Jaffe. A Decentralized Optimal Multiple-User Flow Control Algorithm. *IEEE Transaction on Communications*, (29):954–962, 1981.
- [JDSZ97] S. Jamin, P. B. Danzig, S. Shenker, and L. Zhang. A measurement-based Admission Control Algorithm for Integrated Services Packet Networks. *IEEE/ACM Transactions on Networking*, 5(1):56–70, February 1997.
- [J.K92] J.Kurose. On Computing Per-Session Performance Bounds in High-Speed Multi-Hop Computer Networks. *Performance 92*, June 1992.
- [JSS96] G. S. Mayor J.A. Silvester, N.L.S. Fonseca and S. P. S. Sobral. The Effectiveness of Multi-level Policing Mechanisms in ATM Traffic Control. *Proc. of IEEE International Telecommunications Symposium 96*, pages 98–102, 1996.
- [Kau93] S. Kauffman. *The origins of Order: Self-Organization and Selection in Evolution*. Oxford University, London, England, 1993.
- [Kel96] F. Kelly. Notes on Effective Bandwidth. *Stochastic Networks: Theory and Applications*, 1996.
- [KK80] F. Kamoun and L. Kleinrock. Analysis of Shared Finite Storage in a Computer Network Node Environment Under General Traffic Conditions. *IEEE Trans. Commun.*, 28:992–1003, 1980.
- [Kle76] L. Kleinrock. *Queueing Systems Volume1*. John Wiley and Sons, 1976.
- [KM94a] S. M. Klivansky and A. Mukherjee. On Long-Range Dependence in NSFNET Traffic. *Tech. Rep. GIT-CC-94-61*, 1994. Georgia Tech.
- [KM94b] K.Sohraby and M.Sidi. On the Performance of Bursty Modulated Sources Subject to Leaky Bucket Rate-Based Access Control Schemes. *IEEE Transaction on Communications*, February 1994.
- [Kol40] A. N. Kolmogorov. Wiener'sche spiralen und einige andere interessante kurven im hilbertschen raum. *C.R.(Doklady Acad. Sci. USSR (N.S.))*, 26:115–118, 1940.

- [KRM98] A. Kolarov, G. Ramamurthy, and T. Murase. Comparison of Three Policing Algorithms for ABR Conformance. In *The 7th IEEE Workshop on Computer-Aided Modeling, Analysis and Design of Communication Links and Networks*, volume 1, pages 27–36, August 1998.
- [KT97] M. Krunz and S.K. Tripathi. Exploiting the Temporal Structure of MPEG Video for the Reduction of Bandwidth Requirements. In *IEEE INFOCOM'97*, pages 67–74, April 1997.
- [Kua94] L. Kuang. On the Variance Reduction Property of the Buffered Leaky Bucket. *IEEE Transaction on Communications*, 42(1):2670–2671, 1994.
- [Kue79] P.J. Kuehn. Approximate Analysis of General Queueing Networks by Decomposition. *IEEE Transactions on Communications*, COM-27:113–126, January 1979.
- [KWC93] G. Kesidis, J. Walrand, and C. Chang. Effective Bandwidth for Multiclass Markov Fluids and Other ATM Sources. *IEEE/ACM Trans. Net.*, 1:424–28, August 1993.
- [KZ95] E. W. Knightly and H. Zhang. Traffic Characterization and Switch Utilization Using a Deterministic Bounding Interval Dependent Traffic Model. In *Proc. of INFOCOM'95*, pages 1137–1145. IEEE, April 1995.
- [LA96] C. W. Lee and M. S. Andersland. Consecutive Cell Loss Controls for Leaky Bucket Admission Systems. *Proc of IEEE GLOBECOM'96*, pages 1732–1737, 1996.
- [Lee94] D.C. Lee. Effects of Leaky Bucket Parameters on the Average Queueing Delay:Worse Case Analysis. In *Proc of IEEE INFOCOM'94*, pages 482–489, 1994.
- [LEWW95] W. Lau, A. Erramilli, J. L. Wang, and W. Willinger. Self-Similar Traffic Generation: The Random Midpoint Displacement Algorithm and Its Properties. in *Proc. of IEEE ICC*, 1995. Pg. 466-470.
- [LO86] W. Leland and T. Ott. Load-Balancing Heuristics and Process Behavior. in *PERFORM'86 and ACM SIGM. 1986 Joint Conf. Comput. Perform. Model., Meas., Eval.*, May 1986. Pg. 54-69.

- [LTWW93] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson. On The Self-Similar Nature of Ethernet Traffic. *ACM/SIGCOMM Computer Communications Review*, 23:183–193, September 1993. Proceedings of the ACM/SIGCOMM'93, San Francisco.
- [LTWW94] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson. On the Self-Similar Nature of Ethernet Traffic (Extended Version). *IEEE/ACM Transactions on Networking*, 2(1):1–15, February 1994.
- [LZ] P. Landsberg and C. Zukowski. A Novel Buffer Sharing Method: Complete Sharing Subject to Guaranteed Queue Minimums. in *Proc. of The First Conf. Comp. Commun.*, pages 43–48. Columbia University, New York.
- [Man63] B. Mandelbrot. New Methods in Statistical Economics. *J. Political Economy*, 71(5):421–440, October 1963.
- [Man69] B. B. Mandelbrot. Long-Run Linearity, Locally Gaussian Processes, H-Spectra and Infinite Variances. *International Economic Review*, 10:82–113, 1969.
- [MAS+98] B. Maglaris, D. Anastassiou, P. Sen, G. Karlsson, and J. D. Robbins. Performance Models of Statistical Multiplexing in Packet Video Network. *IEEE Transactions on Communications*, 36(7):834–844, July 1998.
- [May97] G. Mayor. *Performance Modeling and Network Management for Self-Similar Traffic*. Ph. D. Dissertattion, University of Southern California, Faculty of the Graduate School, May 1997.
- [MdV96] M. Montgomery and G. de Veciana. On the Relevance of Time Scales in Performance Oriented Traffic Characterizations. *IEEE Proc. of INFOCOM*, pages 513–520, 1996.
- [MH78] A. I. McLeod and K. W. Hipel. Preservation of the Rescaled Adjusted Range: 1. A Reassessment of the Hurst Phenomenon. *Water Resources Research*, 14(3):491–508, 1978.
- [MN68] B. B. Mandelbrot and J. W. V. Ness. Fractional Brownian Motions, Fractional Noises and Aplications. *SIAM Review*, 10:422–437, 1968.
- [MS95] G. Mayor and J. A. Silvester. The Multi-Level Leaky Bucket Mechanism. In *Proc of the 4th IEEE International Conference on Computer Communications and Networks*, 1995.

- [MS96] G. Mayor and J. Silvester. A Comparative Study of Congestion Detection Mechanisms. *IEEE ITS'96*, 1996.
- [MS98] G. Mayor and J. Silvester. Providing QoS for Long-Range Dependent Traffic. *the 7th IEEE Computer-Aided Modeling Analysis and Design of Communications Link and Networks*, pages 19–28, August 1998.
- [MSA] G. Mayor, J. A. Silvester, and K. Aleronder. Time Scale Analysis of an ATM Queueing System with Long Range Dependence Traffic. *a ser publicado em Proc. of IEEE INFOCON 1997*.
- [MT79] B. B. Mandelbrot and M. S. Taqqu. Robust R/S Analysis of Long Run Serial Correlation. *Proc. 42nd Session ISI*, 1979. Pg. 69-99, Book 2.
- [MW69] B. B. Mandelbrot and J. R. Wallis. Computer Experiments with Fractional Gaussian Noises. *Walter Resources Research*, 5:228–267, 1969.
- [NAZH95] T. Neame, R. G. Addie, M. Zukerman, and F. Huebner. Investigation of Traffic Model of High Speed Data Networks. In *ATNAC'95*, Sydney, December 1995.
- [Nor94] I. Norros. A Storage Model with Self-similar Input. *Queueing Systems*, 16, 1994.
- [Nor95] I. Norros. On the Use of Fractional Brownian Motion in the Theory of Connectionless Networks. *IEEE Journal On Selected Areas in Communications*, 13(6):953–962, August 1995.
- [Pap91] A. Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, 1991.
- [PE95] P. Pruthi and A. Erramilli. Heavy-Tailed on/off Source Behavior and Self-Similar Traffic. *in Proc. of IEEE ICC*, 1995. Pg. 445-450.
- [PF95] V. Paxson and S. Floyd. Wide Area Traffic: The Failure of Poisson Modeling. *IEEE/ACM Transaction on Networking*, 3(3):226–244, June 1995.
- [Rat91] E. Rathgeb. Modeling and Performance Comparison of Policing Mechanisms for ATM networks. *IEEE JSAC*, April 1991.
- [RE96] B. Ryu and A. Elwalid. The Importance of Long-Range Dependence of VBR Video Traffic in ATM Traffic Engineering: Myths and Realities. *in Proc. of ACM SIGCOMM*, 1996.

- [Rob94] P. M. Robinson. Semiparametric Analysis of Long-Memory Time Series. *Ann. Statistic*, (22):515–539, 1994.
- [Ros97] Keith W. Ross. *Multiservice Loss Models for Broadband Telecommunication Networks*. 1997. Chapter 4 (Admission Control).
- [Sch96] Mischa Schwartz. *Broadband Integrated Networks*. 1996.
- [SLG89] M. Sidi, W.Z. Liu, and I. Gopal. Congestion Control Through Input Rate Regulator. *Proc of IEEE Globecom'89*, pages 1764–1768, 1989.
- [SM97] C. Stathis and B. Maglaris. Modeling the Self-similar Behavior of Network Traffic. *IFIP 6th Workshop on Performance of ATM Network*, 1997.
- [ST94] G. Samarodnitsky and M. S. Taqqu. *Stable Non-Gaussian Processes: Stochastic Models with Infinite Variance*. Chapman and Hall, 1994.
- [tAF96] the ATM Forum. Traffic Management Specification. Technical report, ATM Forum, 1996.
- [TG97] Boris Tsybakov and Nicolas D. Georgana. On Self-Similar Traffic in ATM Queues: Definitions, Overflow Probability Bound, and Cell Delay Distribution. *IEEE/ACM Transactions on Networking*, 5(3):397–409, June 1997.
- [THP94] L. Tassiulas, Y. C. Hung, and S. S. Panwar. Optimal Buffer Control during Congestion in a ATM Network Node. *IEEE/ACM Transactions on Networking*, 4(2):374–386, August 1994.
- [TL86] M. S. Taqqu and J. B. Levy. *Using renewal processes to generate long-range dependence and high variability*, volume 11. in Dependence in Probability and Statistics, E. Eberlein and M. S. Taqqu, eds. Boston, MA:Birkhauser, 1986.
- [TT] Murad S. Taqqu and Vadim Teverovsky. Robustness of Whittle-type Estimators for Time Series with Long-Range Dependence. *to appear in Stochastic Models*.
- [WTE96] W. Willinger, M.S. Taqqu, and A. Erramilli. *A Bibliographical Guide to Self-Similar Traffic and Performance Modeling for Modern High-Speed Networks*. Stochastic Networks: Theory and Applications, Oxford, U.K.: Oxford Univ., 1996.
- [WTS97] W. Willinger, M. S. Taqqu, and R. Sherman. Proof of a Fundamental Result in Self-Similar Traffic Modeling. *Computer Commun. Rev.*, 27:5–23, 1997.

- [WTSW97] W. Willinger, M. S. Taqqu, R. Sherman, and D. V. Wilson. Self-Similarity Through High-Variability: Statistical Analysis of Ethernet LAN Traffic at the Source Level. *IEEE/ACM Transactions on Networking*, 5(1):71–86, February 1997.
- [Yaj85] Y. Yajima. On Estimation of Long-Memory Time Series Models. *Australian Journal of Statistics*, 27:303–320, 1985.
- [YS93] O. Yaron and M. Sidi. Performance and Stability of Communication Networks via Robust Exponential Bounds. *IEEE/ACM Transactions on Networking*, 1(3):372–385, June 1993.
- [YSS95] N. Yamanaka, Y. Sato, and K.I. Sato. Performance Limitation of the Leaky Bucket Algorithm for ATM Networks. *IEEE Transacion on Communication*, (43):2298–2230, 1995.
- [ZK94] H. Zhang and E. W. Knightly. Providing End-to-End Statistical Performance Guarantees with Bounding Interval Dependent Stochastic Models. *Proc. ACM SIGMETRICS'94 Conf.*, pages 211–220, May 1994.
- [ZPT94] Z.Liu, P.Nain, and D. Towsley. Exponential Bounds with an Application to Call Admission. Technical report, INRIA, B.P. 93, 06902, Sophia Antipolis Cedex, France, October 1994.
- [ZT97] M. Zukerman and P. W. Tse. An Adaptive Connection Admission Control Scheme for ATM Networks. In *Proceeding of IEEE ICC*, Montreal, Canada, June 1997. Pg. 1153-1157.