

Universidade Estadual de Campinas  
Faculdade de Engenharia Elétrica e de Computação  
Departamento de Comunicações

Modelos Auto-Similares para Tráfego  
de Taxa de Bit Variável em Redes ATM

Albanita Gomes Dantas de Oliveira

Orientador:  
Prof. Dr. Dalton Soares Arantes

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

Campinas, SP - Brasil  
1997

Este exemplar corresponde à edição final da tese  
defendida por Albanita Gomes Dantas de  
Oliveira e aprovada pela Comissão  
Jugadora em 08/05/97  
Dalton S. Arantes  
Orientador

BIBLIOTECA CENTRAL

9708507

UNIDADE	BC
N.º CHAMADA	77 UNICAMP
V.	OL4m
T. Nº 6 /	31153
P. Nº	281/97
C	<input type="checkbox"/> 0 <input checked="" type="checkbox"/>
PREÇO	R\$ 11,00
DATA	24/07/97
N.º CPD	

CM-00099293-1

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

OL4m Oliveira, Albanita G. D. de  
Modelos auto-similares para tráfego de taxa de bit  
variável em redes ATM / Albanita G. D. de Oliveira.--  
Campinas, SP: [s.n.], 1997.

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

1. Telecomunicações - Tráfego. 2. Redes de  
computação. 3. Fractais. I. Arantes, Dalton S.. II.  
Universidade Estadual de Campinas. Faculdade de  
Engenharia Elétrica e de Computação. III. Título.

## AGRADECIMENTOS

À minha família, pelo apoio e paciência em todos os momentos deste trabalho.

Ao meu orientador, Prof. Dalton Soares Arantes, por sua atenção, dedicação e compreensão e por tudo que me ensinou.

Aos Professores João Marcos Travassos Romano, Amauri Lopes, Max Henrique Machado Costa, Hélio Waldman e Roberto A. Lotufo, por valiosas contribuições e sugestões.

Aos colegas Marcelo Menezes de Carvalho e Armando Garcia Weill, por valiosas contribuições a este trabalho.

Aos demais colegas do DECOM, pela colaboração e convivência durante todo o período de minha permanência na UNICAMP.

Ao administrador da rede de computadores do DECOM, Mario Niimi da Cruz, pela sua boa vontade e presteza no atendimento.

Ao colega Wellington D. Felix da Silva, do DCA, pelos esclarecimentos sobre a utilização do *Khoros*.

Aos funcionários da Biblioteca da Área de Engenharia, principalmente à Raquel e à Rose, pela colaboração na pesquisa bibliográfica.

Ao CPqD da TELEBRÁS, pelo apoio e pela cessão das seqüências de vídeo utilizadas neste trabalho.

Ao Laboratório Bellcore, por permitir a utilização dos dados de tráfego de sua rede.

À Marinha do Brasil, pela oportunidade de realizar este trabalho com dedicação exclusiva.

À todos os amigos que me apoiaram e me incentivaram durante a realização deste trabalho.

Acima de tudo, agradeço a Deus, meu Pai, que me sustenta e me guia por caminhos seguros.

*À minha família*

## RESUMO

Este trabalho tem por objetivo a análise de modelos auto-similares para o tráfego de taxa de bit variável em Redes ATM. Os diversos modelos recentemente introduzidos para a caracterização desse tipo de tráfego, são apresentados e exemplificados com dados reais típicos e com dados gerados artificialmente. Esses dados englobam sinais de tráfego reais de uma rede Ethernet e sinais típicos de imagens paradas monocromáticas, obtidas aleatoriamente na Internet. Sinais de vídeo foram também utilizados. Alguns resultados interessantes para a caracterização estatística de sinais de vídeo de taxa de bit variável são apresentados. Uma nítida diferenciação dos processos espaciais (intraquadros) e dos processos temporais (interquadros) de sinais de vídeo, pode ser observada com estes modelos, onde o parâmetro de Hurst foi utilizado como medida do grau de auto-similaridade. Além de sua grande importância no projeto, dimensionamento e gerenciamento das Redes ATM, estes modelos podem ser muito úteis também no projeto e na avaliação de codificadores de vídeo de taxa de bit variável para essas redes. São ainda abordadas, neste trabalho, algumas formas de construção de processos estocásticos auto-similares, o que resultou na implementação de um gerador de séries temporais desse tipo. Todos os programas implementados são listados nos apêndices e as peculiaridades dos vários modelos são enfatizadas. Uma extensa referência bibliográfica é também apresentada ao final deste trabalho.

# Índice

<b>1</b>	<b>As Redes de Faixa Larga</b>	<b>9</b>
1.1.	Introdução	9
1.2.	A Rede Digital de Serviços Integrados de Faixa Larga (B-ISDN)	9
1.2.1.	As Redes ATM	10
1.2.2.	A Comutação ATM e os Multiplexadores Estatísticos	14
<b>2</b>	<b>Os Processos AutoSimilares e o Tráfego da Rede Ethernet</b>	<b>19</b>
2.1.	Introdução	19
2.2.	Os Processos Brownianos	19
2.3.	O Tráfego na Rede Ethernet	22
2.4.	Caracterização Matemática da Auto-Similaridade	23
2.5.	Estimativa do Grau de Auto-Similaridade	24
2.5.1.	Análise da estatística R/S	24
2.5.2.	Método dos Valores Absolutos dos Processos Agregados	25
2.5.3.	Método de Higuchi	25
2.5.4.	Análise das Variâncias dos $X^{(m)}$	26
2.5.5.	Análise do Periodograma	26
2.5.6.	Análise da Estimativa de Máxima Verossimilhança Baseada em Periodograma	27
2.6.	Comprovação da Característica Auto-Similar do Tráfego da Rede Ethernet do Laboratório Bellcore	27
2.6.1.	Diagrama "Pox" da Estatística R/S	31
2.6.2.	Gráfico da Variância	32
<b>3</b>	<b>Sinais com Taxa de Bit Variável em Redes de Faixa Larga</b>	<b>34</b>
3.1.	Introdução	34
3.2.	Limitantes para os Algoritmos de Compressão	34
3.3.	Análise do Tráfego Agregado Gerado por Imagens Monocromáticas	36
3.3.1.	Visualização do sinal agregado em diferentes escalas	39
3.3.2.	O parâmetro de Hurst na agregação semi-paralela	43
3.3.3.	O parâmetro de Hurst na agregação semi-sequencial	47
3.4.	Análise da Auto-Similaridade em Sequências de Vídeo	48

<b>4</b>	<b>Geração de Processos AutoSimilares</b>	<b>57</b>
4.1.	Introdução . . . . .	57
4.2.	Obtenção de Processos fBm a partir da Agregação de Processos Inovações	57
4.3.	Geração do Modelo ARIMA fracionário . . . . .	58
4.3.1.	Modelos Autoregressivo (AR) e da Média Deslizante (MA) . . . . .	59
4.3.2.	O Modelo ARIMA Fracionário . . . . .	60
4.3.3.	O Modelo ARIMA Fracionário de Variância Infinita . . . . .	61
4.4.	Gerador de Séries Temporais Auto-Similares . . . . .	64
<b>5</b>	<b>Conclusões</b>	<b>66</b>
<b>A</b>	<b>A Geometria Fractal</b>	<b>68</b>
A.1.	Introdução . . . . .	68
A.1.1.	A Dimensão Fractal . . . . .	69
A.1.2.	Exemplo: O Movimento Browniano Fracionário (fBm) . . . . .	76
<b>B</b>	<b>Listagem dos Programas Fontes</b>	<b>82</b>
B.1.	Programas para o Gráfico da Variância e “Pox Plot” . . . . .	82
B.2.	Programa para o Cálculo da Variância das Imagens Monocromáticas . . . . .	88
B.3.	Programas para o Cálculo da Variância das Sequências de Vídeo . . . . .	98

# Lista de Figuras

1.1	Arquitetura básica de uma Rede ATM de faixa larga. . . . .	11
1.2	Estrutura da célula ATM. (CGF=controle de fluxo genérico; ITV = identificador de trajeto virtual; ICV = identificador de canal virtual; TCU = tipo de carga útil; PPC = prioridade de perda de célula; VEC = verificação de erro de cabeçalho). . . . .	12
1.3	Sequência de células ATM estatisticamente multiplexadas e sincronizadas a nível de bit. Células vazias podem ocorrer com frequência. . . . .	13
1.4	Sinal ATM agregado em um enlace TDM síncrono a nível de bit. . . . .	13
1.5	Diagrama genérico de um comutador ATM. . . . .	15
1.6	Diagrama esquemático típico de um multiplexador estatístico. . . . .	16
1.7	Variação típica da vazão de entrada ( $V_s$ ) em relação à vazão fixa de saída ( $V_o$ ) em um multiplexador estatístico. . . . .	16
2.1	Resultados do tráfego Ethernet coletados no Bellcore (coluna da esquerda) comparados com o modelo de Poisson (coluna da direita). Escalas de tempo variam de 100 segundos (a) até 0,01 segundos (e). . . . .	29
2.2	“Pox plot” de R/S em função de $n$ (escala $\log - \log$ ) para os dados do Bellcore. A declividade de uma reta interpolada no diagrama corresponde a $H = 0,79$ . . . . .	30
2.3	Diagrama “pox” obtido com as primeiras 10.000 amostras do arquivo pOct.TL do Bellcore. . . . .	31
2.4	Diagrama “pox” para tráfego Poissoniano gerado artificialmente. . . . .	32
2.5	Gráfico da variância para as primeiras 10.000 amostras de tráfego do arquivo pOct.TL do Bellcore (tráfego interno). . . . .	33
2.6	Gráfico da variância para as primeiras 10.000 amostras de tráfego do arquivo OctExt.TL do Bellcore (tráfego externo). . . . .	33
3.1	Curva típica de uma Função Taxa-Distorção. . . . .	35
3.2	Curva típica de compressão para sinais confinados, caracterizada pela Complexidade de Kolmogorov. . . . .	36
3.3	Uma amostra de imagens recortadas que tipicamente são transmitidas na rede Internet. . . . .	37
3.4	Forma semi-paralela para agregação das atividades das imagens. . . . .	38
3.5	Forma semi-sequencial para agregação das atividades das imagens. . . . .	39
3.6	Série temporal completa obtida por agregação semi-paralela (26.624 pontos). . . . .	40
3.7	Série agregada a partir da original, com $m = 64$ . . . . .	40

3.8	Trecho ampliado do trecho anterior, agregado com $m = 32$ .	40
3.9	Trecho ampliado do trecho anterior, agregado com $m = 16$ .	41
3.10	Trecho ampliado do trecho anterior, agregado com $m = 8$ .	41
3.11	Trecho ampliado do trecho anterior, agregado com $m = 4$ .	41
3.12	Série temporal completa obtida por agregação semi-seqüencial (26.624 pontos).	42
3.13	Série agregada a partir da original, com $m = 64$ .	42
3.14	Trecho ampliado do trecho anterior, agregado com $m = 32$ .	42
3.15	Trecho ampliado do trecho anterior, agregado com $m = 16$ .	43
3.16	Trecho ampliado do trecho anterior, agregado com $m = 8$ .	43
3.17	Trecho ampliado do trecho anterior, agregado com $m = 4$ .	43
3.18	Gráfico da variância para o tráfego agregado de 53 imagens (modo semi-paralelo).	44
3.19	Gráfico da variância para o tráfego agregado de 104 imagens no modo semi-paralelo, com reta interpolada em $80 < m < 1.586$ .	45
3.20	Trecho expandido da Figura 3.19, para $252 < m < 1.586$ .	46
3.21	Gráfico da variância para blocos de tamanho $8 \times 8$ .	47
3.22	Gráfico da variância para agregação semi-seqüencial e blocos de tamanho $4 \times 4$ .	48
3.23	Gráfico da variância para as 9 cenas do RPI.	50
3.24	Gráfico “pox” para as 9 cenas do RPI.	50
3.25	Autocorrelação para as 9 cenas do RPI.	51
3.26	Série correspondente às amostras das cenas do RPI e CPqD justapostas.	51
3.27	Gráfico da variância para as cenas compostas do RPI e CPqD.	52
3.28	Gráfico “pox-plot” para as cenas compostas do RPI e CPqD.	52
3.29	Autocorrelação das cenas do RPI e CPqD justapostas.	53
3.30	Série correspondente à cena “lecture”.	54
3.31	Série correspondente às cenas “advertisements”.	54
3.32	Gráfico da variância para a cena MPEG “lecture”.	55
3.33	Gráfico da variância para as cenas MPEG “advertisements”.	55
4.1	Filtragem do modelo AR.	59
4.2	Filtragem do modelo MA.	60
4.3	Gráfico limite do expoente característico ( $\alpha$ ) versus o parâmetro de Hurst ( $H$ ).	64
A.1	Folha de samambaia idealizada (extraída de H. O. Peitgen e D. Saupe [14]).	70
A.2	A curva de von Koch (o floco de neve é composto de três dessas curvas).	71
A.3	A curva estatística de von Koch.	72
A.4	Gráfico $\log - \log$ de $N(r)$ em função de $r$ , para o cálculo da dimensão fractal.	74
A.5	Características fractais de diversas costas marítimas [15].	75
A.6	Sete amostras de diferentes processos fracionários Brownianos (fBm).	77

A.7	Várias amostras de ruído <b>fBm</b> reescaladas, para $H = 0,2$ (esquerda) e $H = 0,8$ (direita) e vários valores de $r$ . Note como o processo é realmente auto-similar. . . . .	79
A.8	Amostras típicas de ruídos fractais que obedecem à lei da potência, para $\alpha = 0$ , $\alpha = 1$ e $\alpha = 2$ . . . . .	80
B.1	Programa elaborado na linguagem <i>Cantata</i> do Ambiente <i>Khoros 2.0</i> para o cálculo da variância dos blocos de $4 \times 4$ ou $8 \times 8$ . O bloco “PROCEDURE” é apresentado na figura abaixo. . . . .	89
B.2	Procedimento para o cálculo da média dos blocos de $4 \times 4$ ou $8 \times 8$ . . . . .	90
B.3	Programa elaborado na linguagem <i>Cantata</i> do Ambiente <i>Khoros 2.0</i> para o cálculo da variância de um bloco de $8 \times 8$ pixels da seqüência <i>mobile</i> . . . . .	99

# Introdução

O crescimento vertiginoso da demanda por serviços de multimídia, tem acelerado de forma significativa o desenvolvimento das redes de faixa larga. Serviços integrados de alta qualidade e de baixo custo estarão disponíveis aos usuários, concomitantemente a um alto grau de utilização dos recursos da rede e elevado faturamento para as empresas de telecomunicações. A recente descoberta da característica auto-similar do tráfego gerado pela rede Ethernet do Laboratório Bellcore [25, 26], bem como por serviços de transmissão de vídeo à taxa de bit variável [26, 29, 21, 62, 63], gerou uma grande apreensão e expectativa na comunidade de redes de faixa larga, em razão de suas sérias implicações sobre o desempenho dessas redes. Em conseqüência, novos caminhos para a modelagem de desempenho de redes de faixa larga estão sendo trilhados, pois os vários modelos tradicionais, baseados na estatística de Poisson, foram colocados em cheque [36]. Torna-se necessário, então, explorar novos modelos que possam caracterizar o comportamento real do tráfego, para permitir o projeto, o dimensionamento e o gerenciamento adequados dessas redes.

Neste trabalho estuda-se os modelos e as características dos vários sinais que deverão se constituir nos principais clientes das redes de faixa larga. Mais precisamente, serão analisados alguns sinais de tráfego de redes locais Ethernet, sinais de imagens típicas comumente transmitidas via Internet, bem como algumas seqüências de vídeo. Este estudo é precedido de uma discussão dos diversos resultados recentemente obtidos na área de processos estocásticos auto-similares (fractais) [6, 8, 9, 15, 25, 26].

Este trabalho está organizado em 5 capítulos e 2 apêndices.

O capítulo 1 apresenta uma visão geral das Redes de Faixa Larga com modo de transferência assíncrono (ATM), abordando o problema da multiplexação estatística na presença de tráfego auto-similar.

O capítulo 2 apresenta a caracterização matemática e métodos para a determinação da auto-similaridade, bem como alguns processos estocásticos dessa natureza. Como ilustração, é apresentada ainda a comprovação da característica auto-similar do tráfego da Rede Ethernet do Laboratório Bellcore, originalmente apresentada por Leland et. al. [25, 26].

No capítulo 3 é realizada uma análise da auto-similaridade do tráfego agregado gerado por imagens monocromáticas e de seqüências de vídeo.

O capítulo 4 trata da construção de processos auto-similares e apresenta um gerador de séries temporais com essa natureza estatística.

As conclusões obtidas deste trabalho são apresentadas no capítulo 5.

O Apêndice A apresenta os conceitos básicos da Geometria Fractal e no Apêndice

B são listados os programas elaborados neste trabalho, inclusive com uma descrição do “workspace” da plataforma KHOROS para a manipulação de imagens monocromáticas e a geração das respectivas séries temporais.

---

# Capítulo 1

## As Redes de Faixa Larga

### 1.1. Introdução

As redes de faixa larga ditas *elásticas*, são capazes de acomodar vários usuários com sinais de taxa de bit variável em meios físicos de transmissão com taxas fixas, tais como nos enlaces SDH (Synchronous Digital Hierarchy). Isto se torna possível graças à utilização de multiplexadores estatísticos estrategicamente inseridos em nós de roteamento dessas redes. Idealmente, a *uniformização obtida por agregação do tráfego* nestes multiplexadores pode oferecer grandes vantagens aos usuários e às empresas operadoras. Conforme será discutido neste trabalho, a hipótese de *uniformização do tráfego* foi inicialmente admitida pelos vários idealizadores das redes elásticas, nas quais a Rede ATM (“Asynchronous Transfer Mode”) se inclui [1, 2, 4, 5]. Esta hipótese é perfeitamente defensável quando se admite que o tráfego agregado é Poissoniano. Estudos recentes, todavia, indicam claramente que o tráfego agregado dessas redes não tem esse comportamento [25, 26, 28, 29, 38, 30], o que implicará em novas abordagens para o problema da Codificação com Taxa de Bit Variável (VBR) e da concepção das Camadas de Adaptação ATM (AAL), bem como em uma possível integração dos processos de gerenciamento reativo da rede e de codificação VBR.

### 1.2. A Rede Digital de Serviços Integrados de Faixa Larga (B-ISDN)

A rede digital de serviços integrados de faixa larga (“Broadband-Integrated Service Digital Network” - B-ISDN) é a evolução das atuais redes de telecomunicações. Está fundamentada nos conceitos de interfaces universais, largura de faixa sob demanda, interconexão de pacotes baseada em hardware e integração de tráfego multimídia. Nessa rede, todos os tipos de tráfego apresentados pelos usuários finais são formados de pequenos pacotes de comprimento fixo. Para isto, necessita-se de segmentação e formatação de blocos fixos de dados. Quanto às interfaces síncronas da rede B-ISDN, estas devem operar inicialmente com velocidades de acesso de 155 Mbits/s e 622 Mbits/s, sendo os dados transportados no modo ATM.

### 1.2.1. As Redes ATM

O modo de transferência assíncrona (ATM) é um conceito motivado, em parte, pela necessidade de se acomodar a grande variedade de serviços oferecidos pela B-ISDN [4, 5].

Na rede ISDN de faixa estreita o modo de transferência é o STM (“Synchronous Transfer Mode”). A regra de subdivisão e alocação de faixa do STM se baseia na designação de “time slots” dentro de uma estrutura recorrente (quadro), para um dado serviço, enquanto durar a chamada. Um canal STM é identificado pela posição do seu “time slot” no quadro. Este modo de transferência tem estrutura rígida, tornando-o ineficiente para serviços de faixa larga com taxa variável, que geram tráfego em surtos (“burst services”).

No modo ATM isto não acontece, pois a capacidade da rede é alocada sob demanda, proporcionando flexibilidade. Seu protocolo explora o ganho estatístico entre serviços com tráfego em surtos e também garante um desempenho aceitável para serviços de taxa de bit constante. Apesar de alguns conceitos fundamentais se basearem na comutação de pacotes, o modo ATM difere significativamente dessa técnica, pois suporta maior variedade de serviços graças à utilização de procedimentos que asseguram melhor desempenho.

O bloco básico de transferência de dados do modo ATM é uma célula de 53 bytes. A arquitetura típica da rede ATM, mostrada na Figura 1.1, é composta basicamente dos seguintes elementos físicos:

- **Equipamento de Usuário (EU)**, cuja função primordial é converter sinais de informação arbitrários em *células ATM sequenciadas*, e vice-versa. A nível de protocolo, esta função é realizada pela *Camada de Adaptação ATM - AAL* (“ATM Adaptation Layer”).
- **Nó de Multiplexação Remota (NMR)**, cujo objetivo é realizar a multiplexação estatística de células ATM oriundas de diversos Equipamentos de Usuários de *baixa taxa de informação*.
- **Nó de Acesso (NA)**, que tem como função concentrar estatisticamente os sinais das NMR’s juntamente com sinais provenientes de outras EU’s de mais alta capacidade.
- **Nó de Comutação Local (NCL)**, que é basicamente uma central comutadora ATM e cujos sinais de entrada são aqueles gerados pelas NA’s, NMR’s e eventualmente EU’s de capacidade muito alta.
- **Nó de Comutação de Trânsito (NCT)**, cuja finalidade é comutar sinais provenientes unicamente das NCL’s. As NCT’s são Centrais de Comutação ATM de grande porte.
- **Enlace de Comunicação Síncrona (ECS)**, que servem para a comunicação física *dedicada e exclusiva* entre as várias NMR’s, NA’s, NCL’s e NCT’s. Estes são enlaces que eventualmente deverão utilizar os padrões SDH (Synchronous Digital Hierarchy) ou SONET (Synchronous Optical Network).

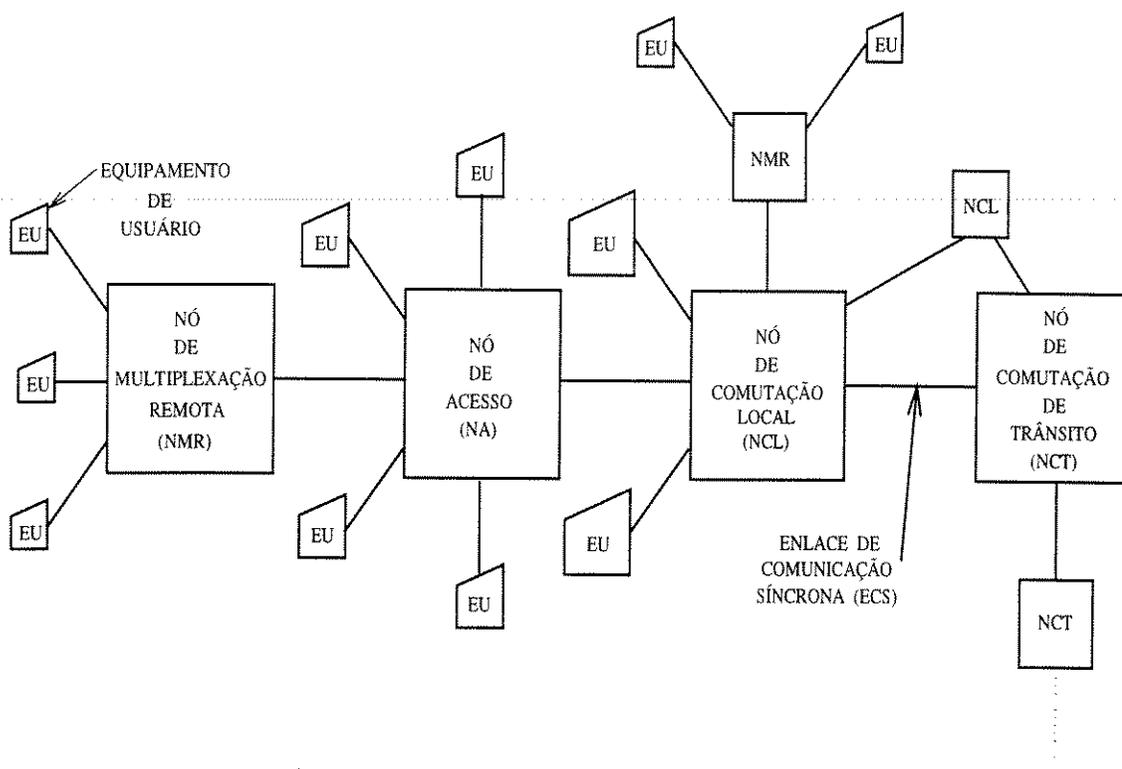


Figura 1.1: Arquitetura básica de uma Rede ATM de faixa larga.

Os usuários da rede são interligados através de *circuitos virtuais orientados a conexão*. Isto significa que não existem recursos físicos alocados exclusivamente para estabelecer a conexão. Vários circuitos virtuais compartilham um subconjunto de recursos de forma estatística, evitando-se, assim, o desperdício de recursos reservados mas nem sempre utilizados. Apesar das células serem transportadas em circuitos virtuais, o serviço oferecido pela rede, visto pelo usuário, pode ser ou não orientado a conexão. O oferecimento desse último serviço é uma característica importante da rede ATM, pois permite a interligação de redes locais tradicionais (LAN's). Com isso, fica plenamente legitimado o potencial mais apreciado da rede ATM, qual seja, o de prover recursos de *serviços integrados*. Deve-se enfatizar, todavia, que a tendência é gradativamente substituir as atuais LAN's por redes locais ATM (ATM LAN), onde estações de trabalho e seus periféricos são conectados diretamente a uma sub-rede ATM de menor porte.

A célula consiste de um cabeçalho de 5 bytes e de 48 bytes de carga útil (dados). O cabeçalho da célula ATM contém informações que permitem encaminhar os dados do endereço do remetente ao do destinatário, bem como dados para o gerenciamento da rede. A Figura 1.2 apresenta a estrutura da célula.

A correção de erros na rede ATM é feita pela rede apenas nos bits do cabeçalho. Proteção de erros na carga útil é deixada para o próprio usuário, que a realiza (ou não) através das camadas superiores do protocolo. Na Figura 1.2 verifica-se a presença de um bit de *priorização de perda de célula*, cuja finalidade é indicar aos multiplexadores estatísticos a prioridade de cada célula na eventualidade de iminente transbordamen-

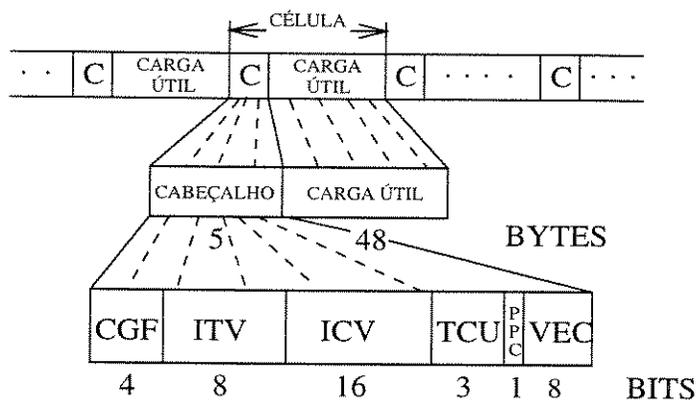


Figura 1.2: Estrutura da célula ATM. (CGF=controle de fluxo genérico; ITV = identificador de trajeto virtual; ICV = identificador de canal virtual; TCU = tipo de carga útil; PPC = prioridade de perda de célula; VEC = verificação de erro de cabeçalho).

to das memórias elásticas. As células de menor prioridade são as primeiras a serem descartadas.

Considerando que as redes ATM serão relativamente sofisticadas e caras, é de se esperar que sejam utilizadas de forma eficiente, isto é, a uma alta vazão. Contudo, a natureza variável da taxa de geração de células das LAN's e dos sinais de usuários, pode forçar a operação da rede com vazões bem inferiores à capacidade máxima. Esta hipótese tem sido levantada em trabalhos recentes, onde se verificou que o tráfego agregado nos nós da rede ATM deverá ter um comportamento estatístico auto-similar (ou fractal) [26]. Neste cenário, fica ainda mais evidenciada a importância do gerenciamento da rede ATM. Ainda é um problema em aberto saber se um gerenciamento inteligente poderá contornar, ou pelo menos amenizar, uma eventual baixa vazão decorrente da auto-similaridade do tráfego. Uma resposta positiva ou negativa a esta questão, talvez só possa ser obtida quando modelos estatísticos mais elaborados forem encontrados. É interessante notar que, caso os usuários fossem de taxa de bit constante (CBR), então a otimização da vazão em um dado instante, onde  $N$  usuários de um nó estivessem ativos, se transformaria em um problema relativamente simples (supondo solução sub-ótima) de alocação de recursos. Neste caso, o problema seria determinístico e flutuações na vazão só ocorreriam quando o número de usuários ativos variasse.

O gerenciamento da rede ATM é basicamente constituído de:

- **Controle de Admissão** - necessário para garantir que novos usuários sejam admitidos na rede somente se não houver perigo de congestionamento em algum dos nós posteriores; este controle depende crucialmente de *descritores estatísticos de tráfego* confiáveis.
- **Policimento de Usuário** - sistema que fiscaliza a estatística dos sinais de usuários, a fim de garantir que estes se enquadrem nas características especificadas no estabelecimento do *contrato de conexão*.
- **Controle de Fluxo** - também conhecido como gerenciamento reativo de tráfego; tem como finalidade detectar possíveis estrangulamentos em nós da rede,

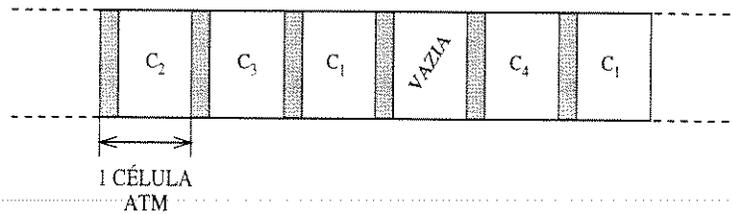


Figura 1.3: Sequência de células ATM estatisticamente multiplexadas e sincronizadas a nível de bit. Células vazias podem ocorrer com frequência.

através da monitoração dos “buffers”; sinais de realimentação são enviados aos nós anteriores, visando impedir a conexão de novos usuários e/ou a utilização de possíveis alternativas de roteamento.

- **Controle de Prioridade** - sistema que visa a otimização da utilização dos enlaces da rede. Baseia-se na informação de prioridade das células; em caso de congestionamento de “buffers”, tenta descartar células de baixa prioridade ou cujos atrasos podem se tornar intoleráveis para o usuário no terminal receptor.

Para que as redes de faixa larga sejam controladas eficientemente é necessária a predição de curto prazo do tráfego baseada em modelos adequados. Como será visto nos capítulos posteriores, os modelos apresentados no presente trabalho fornecem as características de curto e longo prazos desse tráfego.

A Figura 1.3 ilustra uma seqüência de células ATM estatisticamente multiplexadas, vista na saída de um dos vários nós ou comutadores ATM. Embora as células estejam posicionadas de forma *assíncrona*, o sinal agregado correspondente é *síncrono a nível de bit*. Portanto, a interpretação do termo *Asynchronous Transfer Mode* exige um certo cuidado.

Este sinal agregado é, em geral, transmitido entre nós da rede através de sistemas TDM (“time division multiplex”) síncronos a nível de bit, como mostrado na Figura 1.4. Nestes enlaces, mas somente nestes, pode haver *fragmentação de células ATM*, pois em geral o tamanho das células não é um sub-múltiplo do tamanho do quadro TDM. A integridade das células é totalmente recuperada no terminal receptor do enlace TDM. Os sistemas SDH e SONET são os principais concorrentes que disputarão os serviços de enlaces de comunicação (ECS’s) entre nós de redes ATM’s, especialmente para longa distância.

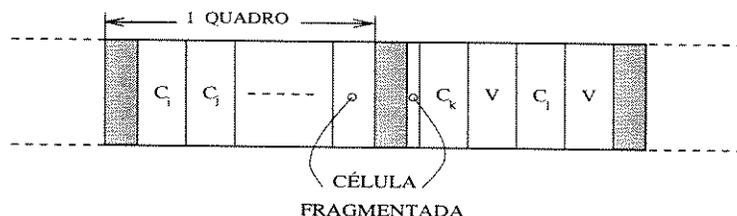


Figura 1.4: Sinal ATM agregado em um enlace TDM síncrono a nível de bit.

Artigos especializados sobre a rede ATM indicam que sua concepção inicial foi fortemente alicerçada na premissa básica da *regularidade estatística do sinal agregado*, que normalmente ocorre quando um grande número de sinais tributários independentes concorrem para um nó ou entroncamento de alta capacidade. Esta suposta *agregação uniforme* seria conseguida essencialmente pelo uso de memórias elásticas e da multiplexação estatística, que suavizariam os sinais de usuários supostos independentes e com características Poissonianas. Em tais ambientes, as memórias elásticas garantem o confinamento dos parâmetros estatísticos do tráfego dentro de limites perfeitamente aceitáveis, onde a rede pode operar de forma *suave, eficiente e previsível*.

Nestas condições, a rede estaria operando de uma forma análoga a uma moderna auto-estrada de alta capacidade e eficiência, onde veículos trafegariam com velocidades e espaçamentos adequados, sem longos congestionamentos, com vias de acesso e saída perfeitamente atendidas, pedágios estrategicamente colocados para uma tarifação justa, sinalização impecável, alta vazão, etc.

Estudos recentes, todavia, indicam que a plena concretização do sonho da filosofia ATM, ainda necessita de uma melhor caracterização da estatística do tráfego na rede. Na verdade, os estudos já realizados em redes locais (LAN's Ethernet), bem como em nós da rede Internet (com protocolo TCP/IP) e em sinais de imagens codificadas com taxa de bit variável (VBR), mostram, de forma insofismável, que a estatística dos pacotes gerados por esses *tributários em potencial* da rede ATM apresentam acentuadas características de auto-similaridade (fractal).

Em outras palavras, a *suavização* prometida pela hipótese de tráfego Poissoniano parece distante da realidade. Em conseqüência, o uso indiscriminado da rede ATM sob a hipótese dos modelos de Poisson, pode levar a vários problemas indesejáveis, tais como uma excessiva taxa de perda de células, grandes atrasos na transmissão de células e, eventualmente, a operação da rede a uma vazão ("throughput") consideravelmente abaixo da vazão máxima idealizada.

### 1.2.2. A Comutação ATM e os Multiplexadores Estatísticos

Um dos principais componentes das Redes ATM é o Comutador de Células ATM, cuja finalidade é encaminhar cada célula para a rota adequada, de forma que o endereço de destino, implícito no cabeçalho, seja atingido. Desta forma, deve também solucionar o problema da contenção que acontece quando duas ou mais células chegam simultaneamente na entrada para serem encaminhadas à mesma saída. Pode-se dizer que um Comutador ATM corresponde a um ou mais Multiplexadores Estatísticos, no que se refere a agregação de células, provenientes de diferentes fontes, em uma única saída com maior taxa de utilização.

A Figura 1.5 mostra um comutador ATM típico, onde os  $n$  enlaces de entrada são comutados para  $q$  enlaces de saída. Cada enlace  $I_i$  de entrada representa o fluxo das células ATM que foram agregadas no nó anterior da rede. Supõe-se que eventuais sinais de controle de multiplexação síncrona (SDH, por exemplo) já foram eliminados à entrada do comutador, daí a representação do fluxo  $I_i$  apenas por células ATM.

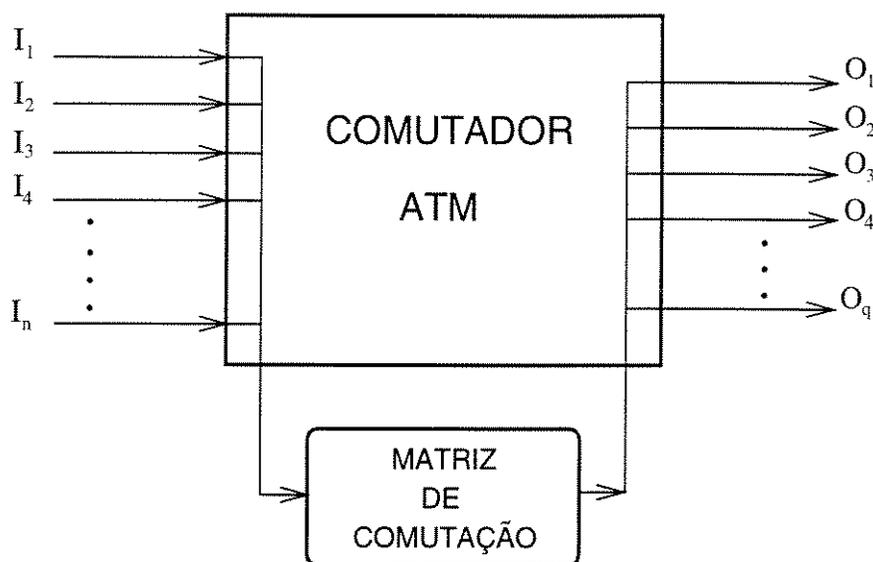


Figura 1.5: Diagrama genérico de um comutador ATM.

A comutação de células de  $I_i$  para  $O_j$  ( $i = 1, 2, \dots, n; j = 1, 2, \dots, q$ ) é realizada por uma *tabela ou matriz de comutação*, como mostrada na Tabela 1.1. Esta é freqüentemente atualizada pelo sistema de controle de admissão da rede, o que ocorre a cada vez que um *canal virtual* é criado ou desativado. Deve-se enfatizar que esta comutação é toda *feita por hardware*, o que viabiliza as altas taxas de transmissão na Rede ATM.

Enlace de entrada	Cabeçalho		Enlace de saída	Cabeçalho
$I_1$	x		$O_3$	k
	y		$O_q$	m
	z		$O_1$	l
.				
.				
.				
$I_n$	y		$O_2$	n
	s		$O_4$	i
	x		$O_q$	g

Tabela 1.1: Matriz de Comutação ATM Típica.

Tendo em vista que o fluxo de células de um certo usuário em cada  $I_i$  é variável e considerando que cada enlace de saída  $O_j$  possui uma capacidade máxima de transmissão de células (enlace SDH), torna-se necessária a utilização de Multiplexadores Estatísticos para agregar e uniformizar o fluxo de todas as células direcionadas ao enlace de saída  $O_j$ . Nota-se que, embora a soma das  $n$  capacidades dos enlaces de entrada seja geralmente igual à soma das  $q$  capacidades dos enlaces de saída, a necessidade dos multiplexadores

estatísticos decorre justamente da variabilidade do fluxo de células destinado a cada enlace de saída.

Seja um enlace de entrada  $I_i$  qualquer. Define-se por  $V_i$  o *fluxo de células* do enlace  $I_i$  que são destinadas (via canal virtual) ao enlace de saída  $O_j$ . Em outras palavras,  $V_i$  representa, em um dado instante, a quantidade de células de  $I_i$  para  $O_j$  por *unidade de tempo*.

Para melhor caracterizar os modelos matemáticos que serão apresentados, admite-se que essa *unidade de tempo* seja igual a  $\Delta T$ . Dessa forma,  $V_1, V_2, \dots, V_n$  representam os fluxos de entrada, conforme mostrado no multiplexador estatístico típico da Figura 1.6, onde  $V_o$  representa a *vazão máxima agregada* suportada por um enlace  $O_j$ , isto é, a quantidade máxima de células por intervalo  $\Delta T$ . O valor de  $V_o$  normalmente é fixado pelas taxas dos enlaces SDH (155 ou 622 Mbits/s, por exemplo).

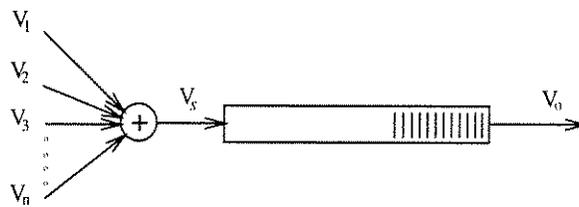


Figura 1.6: Diagrama esquemático típico de um multiplexador estatístico.

Para facilitar a notação, denomina-se por *vazão* tanto os *fluxos*  $V_i$  de entrada quanto a *vazão*  $V_o$  em um dado enlace de saída. Note que  $V_s$  é a soma de todas as vazões de entrada.

Na Figura 1.7 é apresentada uma curva típica da variação da vazão total de entrada em função do tempo. Este gráfico é bastante ilustrativo, pois ele justifica claramente a necessidade das *memórias elásticas* frente ao problema da eficiência de uso dos enlaces. Por exemplo, na hipótese da não utilização dessas memórias, haveria desperdício de recursos de transmissão, isto é, perda de faixa, nos instantes onde  $V_s < V_o$ , ao passo que quando  $V_s > V_o$  haveria interrupção parcial de transmissão de informação (Perda de Células) à taxa de  $V_s - V_o$ .

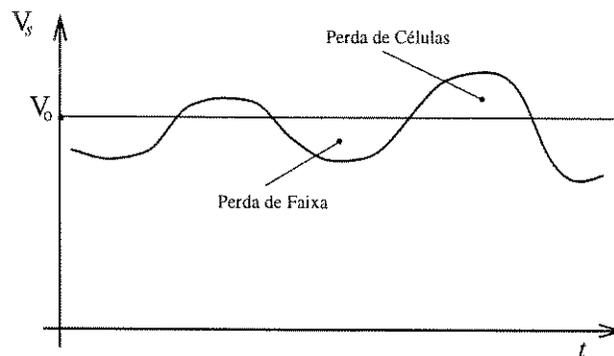


Figura 1.7: Variação típica da vazão de entrada ( $V_s$ ) em relação à vazão fixa de saída ( $V_o$ ) em um multiplexador estatístico.

É justamente a capacidade das memórias elásticas de absorverem estas variações de vazão, tomando seu valor médio, que tornam os multiplexadores estatísticos as peças importantes das redes com usuários de taxa de bit variável (VBR).

Até alguns anos atrás, admitia-se como hipótese realista que os processos estocásticos representativos das vazões (e conseqüentemente de  $V_s$ ) eram processos de Poisson ou similares [25, 26, 38]. Supunha-se que estes processos possuíam *incrementos independentes*.

Chamemos de  $X_s$  o *processo de contagem de células* à entrada da memória elástica da Figura 1.6. Isto é, a partir de um instante  $t = t_0$ ,  $X_s(t)$  representa o número total de células a ser armazenado na memória no intervalo de tempo  $t - t_0$ . Nesse mesmo intervalo, serão lidas ou retiradas da memória uma quantidade de células igual a  $(t - t_0)V_o$ , incluindo-se aí as células eventualmente vazias.

Considerando-se que  $\Delta T$  foi fixado, a vazão  $V_s$  é exatamente representada por

$$V_s(t) = \frac{X_s(t + \Delta T) - X_s(t)}{\Delta T},$$

isto é,  $V_s$  é dada pelos *incrementos normalizados* de  $X_s$ . Nota-se que o processo  $V_s$  é a *derivada* do processo  $X_s$  e, naturalmente, a *integral* de  $V_s$  resulta em  $X_s$ .

Dessa forma, a hipótese de tráfego Poissoniano ou, mais genericamente, de tráfego com *incrementos independentes*, implica em que as amostras de  $V_s$  sejam *independentes*!

Neste caso, a multiplexação estatística estaria operando da forma mais favorável possível, pois os eventuais picos de vazões ( $V_s > V_o$ ) seriam rapidamente compensados pelos seus *vales* subseqüentes ( $V_s < V_o$ ). Isto decorre justamente da independência das amostras de  $V_s$ , que corresponde a um *processo estocástico rápido* à semelhança do ruído Gaussiano branco.

Um problema crucial no desempenho das Redes ATM é o transbordamento de células nas memórias elásticas. Quando isto ocorre, há perda de células com a conseqüente degradação dos serviços de usuário. Na transmissão de dados, a perda de células acarreta pedidos de retransmissão na camada de adaptação, causando atrasos e decréscimo da vazão de dados. Quando se trata de transmissão de vídeo (MPEG, por exemplo), a perda de células implica em degradação da qualidade da imagem e até em perda momentânea de sincronismo de quadro. Além disso, o enchimento freqüente das memórias aumenta o chamado *atraso de fila* em todos os canais virtuais que compartilham os respectivos multiplexadores. Daí a importância da modelagem estatística dos sinais que trafegam pela rede.

Pode-se mostrar que sob a *hipótese de incrementos independentes*, a probabilidade de transbordamento ( $P_d$ ) em um multiplexador com memória de tamanho  $d$  (medido em células, por exemplo), *decrece exponencialmente* com  $d$ , desde que  $E[V_s]$ , o valor médio de  $V_s$ , seja estritamente menor que  $V_o$ . Além disso, o atraso médio de fila tende a um valor constante à medida que  $d$  tende a infinito.

Deve-se notar que este decaimento exponencial significa, na prática, que pode-se conseguir probabilidades de transbordamento desprezíveis com o uso de memórias relativamente pequenas.

Infelizmente, constatou-se recentemente [26, 38] que os principais sinais de taxa de bit variável, que farão uso da rede ATM, possuem *incrementos dependentes*, consistindo

de *processos estocásticos lentos*. Curiosamente, verificou-se também que esses sinais possuem fortes características auto-similares (fractais), com variações significativas em várias escalas de tempo [26].

Estes resultados têm despertado grande interesse de pesquisadores da área de redes de faixa larga e processamento de sinais, pois eles certamente terão grande importância na concepção e operação de redes eficientes, bem como nos algoritmos de compressão de sinais para essas redes.

Note que quando o processo  $V_s$  da Figura 1.7 é lento, o transbordamento da memória pode tornar-se muito freqüente, principalmente quando se deseja operar com  $E[V_s]$  próximo de  $V_o$  (alta eficiência). Na realidade, já foi demonstrado em trabalho recente [37], que para processos auto-similares a probabilidade de transbordamento,  $P_d$ , decai apenas *algebricamente* com  $d$ . Além disso, o atraso médio de fila cresce indefinidamente com  $d$ .

## Capítulo 2

# Os Processos Auto-Similares e o Tráfego da Rede Ethernet

### 2.1. Introdução

Visando o estudo de modelos para o tráfego auto-similar, apresenta-se neste capítulo as características dos processos auto-similares, as definições matemáticas e métodos para estimativa do grau de auto-similaridade. São apresentadas também considerações sobre o comportamento do tráfego da rede Ethernet do Laboratório Bellcore (USA), as quais deram origem ao enfoque auto-similar para os modelos de tráfego em redes de faixa larga.

### 2.2. Os Processos Brownianos

Os modelos mais atuais para a caracterização do tráfego em redes de faixa larga podem ser melhor introduzidos através dos chamados movimentos Brownianos fracionários (**fBm**), definidos por Kolmogorov [10] e por Mandelbrot e Van Ness [9]. As principais características destes processos são discutidos neste capítulo e mais detalhes são apresentados no Apêndice A.

O *processo de contagem* de Poisson é dado pela integral do *processo de incrementos* correspondente. Este, naturalmente, é dado pela derivada do primeiro. Analogamente, o conhecido *Movimento Browniano* (ou *Processo de Wiener*) (**Bm**) é definido como a integral do *Ruído Gaussiano Branco* e este como a derivada daquele. Neste caso também, os processos **Bm** são de incrementos independentes e estacionários, pois estes coincidem com amostras de ruído branco. Deve ser esclarecido que a derivada e, portanto, a integral são interpretadas no sentido de média quadrática e não na forma usual, pois um processo estocástico não é uma função. Para explicar esta interpretação, considera-se  $w(t)$  um processo **Bm** e, para um dado intervalo arbitrário  $[t, t + \tau]$

$$x_\tau(t) = \frac{w(t + \tau) - w(t)}{\tau}. \quad (2.1)$$

Define-se, então, o processo ruído branco como

$$v(t) = \text{l.i.m.}_{\tau \rightarrow 0} x_\tau(t), \quad (2.2)$$

onde *l.i.m.* é o *limite no sentido da média quadrática* que, por sua vez é definido da seguinte forma: diz-se que uma seqüência de processos aleatórios  $x_i$ , ( $i = 0, 1, 2, \dots, \infty$ ), converge *no sentido da média quadrática* para o processo  $x(t)$  se

$$\lim_{i \rightarrow \infty} E [(x(t) - x_i(t))^2] = 0. \quad (2.3)$$

Então, por simples analogia com o limite de uma seqüência de funções, pode-se dizer que o ruído branco  $v(t)$  é a *derivada* do processo **fBm**,  $w(t)$ .

Mandelbrot e Van Ness [9] definiram os processos **fBm** da seguinte forma. Seja  $B(t)$  um Movimento Browniano puro (não-fracionário) e seja  $H$  um parâmetro real satisfazendo  $0 < H < 1$ . Um processo **fBm** de expoente  $H$  é obtido tomando-se a *média deslizante* de sucessivos *incrementos infinitesimais* de  $B(t)$ , isto é, de  $dB(t)$ , ponderados pelo "kernel"  $(t - s)^{H-1/2}$ .

Mais precisamente,  $B_H(t)$  será um processo **fBm** com parâmetro  $H$  se

$$B_H(0) = b_0 \quad (\text{em geral } b_0 = 0), \quad (2.4)$$

$$B_H(t) - B_H(0) = \frac{1}{\Gamma(H + 1/2)} \left\{ \int_{-\infty}^0 [(t - s)^{H-1/2} - (-s)^{H-1/2}] dB(s) + \int_0^t (t - s)^{H-1/2} dB(s) \right\}, \quad (2.5)$$

onde  $\Gamma(x)$  é a função Gama aplicada a  $x$ ,  $B(t)$  é o movimento Browniano ordinário e  $0 < H < 1$ .

Pode-se verificar que  $E[B_H(t)] = 0$  e que para  $H = 1/2$  o processo  $B_H(t)$  coincide com o movimento Browniano  $B(t)$ . É fácil verificar também que  $B_H(t)$  é um processo *não-estacionário*, mas seus incrementos possuem *média zero* e são *estacionários*.

Pode-se mostrar [41, 42] que a função de autocorrelação do movimento Browniano fracionário  $B_H(t)$ , é dada por

$$R(s, t) = \frac{\sigma^2}{2} (|s|^{2H} + |t|^{2H} - |s - t|^{2H}), \quad (2.6)$$

onde

$$\sigma^2 = \text{Var}[B_H(1)]. \quad (2.7)$$

Daí resulta que

$$\text{Var}[B_H(t)] = \sigma^2 |t|^{2H}. \quad (2.8)$$

Como  $B_H(t)$  é um processo não-estacionário, ele não admite uma densidade espectral de potência no sentido usual. Porém, pode-se definir um espectro médio para  $B_H(t)$  [42], que resulta em

$$S_{B_H}(\omega) = \frac{\sigma^2}{\omega^{2H+1}}. \quad (2.9)$$

Como será visto mais adiante,  $B_H(t)$  é um processo auto-similar e cada uma de suas funções amostras é uma *curva fractal* (Apêndice A). Já foi demonstrado [8, 42] que a dimensão fractal dessas curvas é dada por

$$D = 2 - H. \quad (2.10)$$

Tendo em vista que  $0 < H < 1$ , daí resulta que  $1 < D < 2$ .

O processo estocástico  $X = (X_k : k \geq 0)$ , obtido por *derivada* de  $B_H(t)$ , é comumente chamado de *ruído Browniano fracionário*. Pode-se mostrar que  $X(t)$  possui média zero, função de autocorrelação

$$r(k) = \frac{\gamma^2}{2} (|k+1|^{2H} - 2|k|^{2H} + |k-1|^{2H}), \quad (2.11)$$

onde  $\gamma^2 = \text{Var}[X(t)]$ , e densidade espectral de potência

$$S_X(\omega) \sim \frac{1}{\omega^{2H-1}} \quad (2.12)$$

para  $1/2 < H < 1$ .

É interessante observar que

$$S_{B_H}(\omega) \sim \frac{S_X(\omega)}{\omega^2}, \quad (2.13)$$

como era de se esperar, pois  $X(t)$  é a derivada de  $B_H(t)$ .

Nota-se que, para  $H = 1/2$ , o processo  $B_H(t)$  é o próprio movimento Browniano ordinário e  $\{X_k\}$  é o próprio ruído Gaussiano branco e, para  $H > 1/2$ , o processo  $B_H(t)$  possui incrementos *dependentes*, isto é, as amostras  $X_k$  são *dependentes*. O caso extremo  $H \rightarrow 1$  corresponde a uma degeneração, onde  $B_H(t)$  é tão lento que tende a uma linha constante.

Na seção 2.3 será enfatizado o comportamento assintótico ( $k \rightarrow \infty$ ) desses processos, em especial para o caso  $H > 1/2$  que corresponde a processos lentos.

Conforme apresentado no capítulo anterior, a partir do *processo de contagem de células* pode-se obter o *processo de incrementos* que, sem dúvida, melhor caracteriza a vazão de células em algum nó da rede.

Deve-se observar que tanto o “movimento Browniano” quanto o “ruído Browniano” possuem média zero. Mas, considerando que *vazão de células* pressupõe média maior que zero, é necessário adaptar os modelos acima às características do tráfego. Esta adaptação só foi apresentada claramente em trabalho recente de Norros [38], que utilizou o conceito de ruído Browniano fracionário *centrado*.

Norros definiu o *tráfego Browniano fracionário* como o processo estocástico  $A_t$  dado por

$$A_t = mt + \sqrt{am}Z_t, \quad (2.14)$$

onde  $Z_t$  é um processo **fBm** com parâmetro  $H$  ( $1/2 \leq H < 1$ ),  $a > 0$  é um coeficiente de variância e  $m > 0$  é a taxa média de geração de células. Note que  $A_t - mt$  é um processo **fBm** centrado com média zero. Note também que a derivada de  $A_t$  corresponde a um ruído *Browniano fracionário*, ao qual foi adicionado a média  $m$ . Este sim, representa o tráfego sob a forma de incrementos (supondo que ele não assuma valores negativos). Os trabalhos que tratam de modelos de tráfego geralmente não abordam este problema.

Com este modelo, Norros [38] conseguiu interessantes resultados analíticos que confirmam os resultados empíricos encontrados por Leland et. al. [25, 26] no Bellcore.

## 2.3. O Tráfego na Rede Ethernet

A rede local tipo Ethernet utiliza o protocolo CSMA-CD, “Carrier Sense Multiple Access - Colision Detection”, com velocidade de 10 Mbps. Neste protocolo as estações “escutam” a rede para saber se há outra transmissão naquele instante. Caso afirmativo, a estação continua “escutando” até que a rede esteja livre e inicia, então, a sua transmissão. Se outra estação da rede começa a transmitir simultaneamente à primeira, é detectada a colisão dos dados e ambas interrompem as transmissões abruptamente, permitindo, assim, economizar tempo e largura de faixa da rede. Após a colisão ambas as estações aguardam durante um período de tempo aleatório e tentam novamente, assumindo que nenhuma outra tenha iniciado a transmissão nesse intervalo.

As características da rede Ethernet constituem a base do padrão 802.3 do IEEE, o qual engloba também velocidades entre 1 a 10 Mbps em vários meios de transmissão.

O tráfego desse tipo de rede local usualmente é modelado por processos de Poisson, Poisson-Batch, Poisson com modulação de Markov, trem de pacotes e fluxo de fluídos, nos quais supõe-se que o tráfego agregado se torna mais *suave* com o aumento do número de fontes. No entanto, em trabalho recente realizado no Laboratório Bellcore (USA), W. E. Leland et al. constataram que o tráfego real tem comportamento bastante diferente dos modelos citados acima [25, 26]. Foram coletados dados estatísticos importantes sobre o tráfego de pacotes na rede Ethernet (LAN) daquele Laboratório, com alta resolução temporal, entre os meses de Agosto de 1989 e Fevereiro de 1992. Além disso, apresentaram também uma análise estatística sobre os dados coletados, onde se comprovou que o tráfego de pacotes é acentuadamente auto-similar.

As redes locais vêm se tornando em um dos maiores contribuintes para o tráfego de redes de alta velocidade tipo B-ISDN. Outra contribuição significativa vem do serviço de transmissão de vídeo a taxa variável (VBR), cujo tráfego também tem característica de auto-similaridade [29, 61, 62, 63].

Assim, o modelo de auto-similaridade proporciona uma descrição simples, precisa e realista do tráfego e produz sérias implicações no projeto, controle e análise de desempenho de redes de alta velocidade tipo ATM. A seguir são apresentados a caracterização matemática para esses modelos e vários métodos para estimativa do grau de auto-similaridade, bem como mais alguns resultados interessantes, além daqueles já discutidos no capítulo anterior.

## 2.4. Caracterização Matemática da Auto-Similaridade

Um processo estocástico  $B_H(t)$  é dito estatisticamente *auto-similar* se  $\{B_H(at), t \geq 0\}$  possui as mesmas distribuições conjuntas finitas que  $\{a^H B_H(t), t \geq 0\}$  para todo  $a > 0$ . O índice  $H$  assume valores entre 0 e 1, é chamado de *parâmetro de Hurst* ou *parâmetro de auto-similaridade* [9]. Na maioria das aplicações, é suficiente definir auto-similaridade usando-se apenas as estatísticas de segunda ordem, como será feito a seguir.

Seja  $X = (X_t : t = 0, 1, 2, \dots)$  um processo estocástico estacionário no sentido amplo, com média  $\mu$  e variância  $\sigma^2$ , que tem função de autocorrelação da forma  $r(k) \sim ak^{-\beta}$ , quando  $k \rightarrow \infty$ , onde  $0 < \beta < 1$  e  $a$  é uma constante positiva finita. Para  $m = 1, 2, 3, \dots$ , seja  $X^{(m)} = (X_k^{(m)} : k = 1, 2, 3, \dots)$  uma nova série temporal formada por médias de blocos de tamanho  $m$ , não sobrepostos, da série original  $X$ . Assim, para cada  $m = 1, 2, 3, \dots$ , a série temporal  $X^{(m)}$  é dada por

$$X_k^{(m)} = \frac{1}{m} (X_{km-m+1} + \dots + X_{km}), \quad (k \geq 1),$$

e constitui um processo estacionário no sentido amplo, com função de autocorrelação  $r^{(m)}(k)$ .

Define-se o processo  $X$  como auto-similar de segunda ordem, com parâmetro de auto-similaridade  $H = 1 - \beta/2$ , se os processos agregados  $X^{(m)}$  têm estrutura de correlação idêntica à de  $X$ , isto é,  $r^{(m)}(k) = r(k)$  e  $\text{var}[X^{(m)}] = \sigma^2 m^{-\beta}$ , para  $m = 1, 2, \dots$  ( $k = 1, 2, 3, \dots$ ). Desta forma,  $X$  é auto-similar de segunda ordem se os processos agregados  $X^{(m)}$  não se distinguem de  $X$  quanto às estatísticas de primeira e segunda ordens.

Um exemplo de processo auto-similar de segunda ordem é o ruído Gaussiano fracionário, apresentado anteriormente, com parâmetro  $1/2 < H < 1$  [9, 25].

Define-se também  $X$  como assintoticamente auto-similar de segunda ordem, com parâmetro  $H = 1 - \beta/2$ , se  $r^{(m)}(k)$  é igual a  $r(k)$ , para grandes valores de  $m$  e  $k$ .

Um exemplo de processo auto-similar assintótico é o ARIMA fracionário  $(p, d, q)$  (“fractional autoregressive integrated moving-average”) com  $0 < d < 1/2$  e parâmetro de auto-similaridade igual a  $d + 1/2$  [25, 26, 18], que será apresentado no Capítulo 4.

Das definições acima, nota-se que nos processos auto-similares os valores agregados  $X^{(m)}$  possuem uma estrutura de correlação não degenerada quando  $m \rightarrow \infty$ .

A auto-similaridade se manifesta nos seguintes aspectos equivalentes:

- A variância decresce lentamente, isto é,  $\text{Var}[X^{(m)}] \sim bm^{-\beta}$  quando  $m \rightarrow \infty$ , com  $0 < \beta < 1$ , sendo  $b$  uma constante positiva finita;
- A autocorrelação decresce hiperbolicamente ao invés de exponencialmente com o valor de  $k$ , o que implica que  $\sum r(k) = \infty$  (dependência de longo prazo);
- A função densidade espectral de potência  $S(\cdot)$  segue uma lei de potência próximo à origem, isto é,  $S(w) \sim cw^{-\gamma}$ , quando  $w \rightarrow 0$ , com  $0 < \gamma < 1$ ,  $\gamma = 1 - \beta$ , sendo  $c$  uma constante positiva finita.

Na década de 50, H.E. Hurst [39] descobriu importantes propriedades de séries temporais oriundas de fenômenos naturais. Estes resultados levaram à formulação de uma lei empírica que é hoje conhecida como *lei de Hurst* ou *efeito Hurst*. Partindo de uma série de observações  $(X_k : k = 1, 2, \dots, n)$ , com média amostral  $\bar{X}(n)$  e variância amostral  $S^2(n)$ , Hurst definiu a *estatística R/S* (“rescaled adjusted range”) dada por

$$\frac{R(n)}{S(n)} = \frac{1}{S(n)} [\max(0, W_1, W_2, \dots, W_n) - \min(0, W_1, W_2, \dots, W_n)] \quad (2.15)$$

sendo

$$W_k = (X_1 + X_2 + \dots + X_k) - k\bar{X}(n) \quad (2.16)$$

para  $k = 1, 2, \dots, n$ .

Observando inúmeros fenômenos naturais, Hurst descobriu que as séries temporais correspondentes eram bem representadas pela relação

$$E [R(n)/S(n)] \propto n^H \quad (n \rightarrow \infty) \quad (2.17)$$

onde  $H$  (*parâmetro de Hurst*) é tipicamente da ordem de 0,73. Este, portanto, é o caso visto anteriormente de *processo fBm lento* [14, 15].

Por outro lado, Mandelbrot e Van Ness mostraram que para *processos fBm rápidos* (baixa correlação), vale a relação

$$E [R(n)/S(n)] \propto n^{0,5} \quad (n \rightarrow \infty) \quad (2.18)$$

É justamente esta discrepância no valor de  $H$  que é conhecida como *efeito Hurst* ou *fenômeno de Hurst*.

## 2.5. Estimativa do Grau de Auto-Similaridade

O grau de auto-similaridade de um processo pode ser avaliado sob três aspectos diferentes:

- análise no domínio do tempo, baseada na estatística  $R/S$ , nos valores absolutos das séries agregadas e na dimensão fractal (Método de Higuchi);
- análise das variâncias dos processos agregados  $X^{(m)}$ ;
- análise no domínio da frequência, baseada em periodograma.

### 2.5.1. Análise da estatística R/S

A análise da estatística de dados empíricos tem por objetivo inferir o parâmetro de Hurst ( $H$ ) do processo. Constitui uma abordagem gráfica heurística, que tenta explorar ao máximo as informações contidas nos dados [20].

Dado um conjunto de  $N$  observações consecutivas ( $X_k : k = 1, 2, 3, \dots, N$ ), subdivide-se o intervalo em  $K$  blocos não sobrepostos e calcula-se o intervalo reescalado ajustado  $R(t_i, n)/S(t_i, n)$  para cada novo ponto inicial  $t_1 = 1, t_2 = N/(K + 1), t_3 = 2N/(K + 1), \dots$ , que satisfaz  $(t_i - 1) + n \leq N$ . Para o cálculo de  $R(t_i, n)/S(t_i, n)$ , apresentado na seção anterior, utiliza-se  $W_k = W_{t_i+k} - W_{t_i}$  e  $S^2(t_i, n) =$  variância de  $X_{t_i+1}, X_{t_i+2}, \dots, X_{t_i+n}$ .

Então, para  $n$  pequeno obtém-se  $K$  amostras de  $R/S$ , e uma amostra se  $n$  está muito próximo do número total de amostras ( $N$ ) do processo  $X$ .

O gráfico  $\log(R(t_i, n)/S(t_i, n))$  versus  $\log n$  é chamado *diagrama "pox" da estatística  $R/S$* .

Se o parâmetro  $H$  é bem definido, o *diagrama "pox"* apresenta uma região de transiente próxima à origem, referente à característica de dependência de curto prazo do processo. Este transiente diminui à medida que se afasta da origem, e os pontos do diagrama tendem a flutuar em torno de uma reta aproximada.

Assim, pela análise do *diagrama "pox"* pode-se verificar se os dados (amostras) de um processo têm esse comportamento. Caso afirmativo, o parâmetro  $H$  é estimado pela inclinação da reta aproximada que pode ser obtida por mínimos quadrados. Esta inclinação pode assumir valores entre  $1/2$  e  $1$ .

A análise da estatística  $R/S$  tem como principal vantagem a robustez diante de alterações da distribuição marginal do processo [25, 26, 27].

### 2.5.2. Método dos Valores Absolutos dos Processos Agregados

Neste método a série temporal original ( $X_k : k = 1, 2, \dots, N$ ) deve ser dividida em blocos não sobrepostos de comprimento  $m$  [27]. Calcula-se, então, os elementos de  $X^{(m)}$ ,

$$X_k^{(m)} = \frac{1}{m} \sum_{i=(k-1)m+1}^{km} X_i, \quad k = 1, 2, \dots \quad (2.19)$$

A seguir, toma-se a média dos valores absolutos dos elementos da série agregada,

$$\frac{1}{N/m} \sum_{k=1}^{N/m} |X_k^{(m)}|. \quad (2.20)$$

Repete-se este procedimento para diversos valores de  $m$  e traça-se o gráfico do logaritmo das médias das séries agregadas versus o logaritmo de  $m$ . Se a série original apresenta dependência de longo prazo com parâmetro  $H$ , o gráfico deve ser uma reta com inclinação  $(H - 1)$  [27].

### 2.5.3. Método de Higuchi

Este método envolve o cálculo do comprimento de um trajeto e a determinação de sua dimensão fractal ( $D$ ) [27]. É semelhante ao Método dos Valores Absolutos, descrito acima. Dada a série temporal ( $X_k : k = 1, 2, \dots, N$ ) calcula-se os somatórios parciais

$$Y(n) = \sum_{i=1}^n X_i, \quad (2.21)$$

o que equivale à obtenção de um processo movimento Browniano fracionário (**fBm**) a partir de um ruído Gaussiano fracionário. A seguir, subdivide-se a série em blocos não sobrepostos de tamanho  $m$  e determina-se o comprimento normalizado da curva,

$$L(m) = \frac{N-1}{m^3} \sum_{i=1}^m \sum_{k=1}^{\lfloor \frac{N-i}{m} \rfloor} \left( \frac{N-i}{m} \right)^{-1} |Y(i+km) - Y(i+(k-1)m)|, \quad (2.22)$$

com  $m$  representando o comprimento da série temporal e  $\lfloor y \rfloor$  o maior inteiro menor ou igual a “ $y$ ”. Repete-se este procedimento para diversos valores de  $m$ . Como

$$E[L(m)] \sim Cm^D, \quad (2.23)$$

sendo  $D = 2 - H$  e  $C$  uma constante positiva, traçando-se o gráfico do logaritmo de  $L(m)$  versus o logaritmo de  $m$ , deve-se obter uma reta com inclinação  $D = 2 - H$ .

#### 2.5.4. Análise das Variâncias dos $X^{(m)}$

Como mencionado anteriormente, as variâncias dos processos agregados  $X^{(m)}$  decrescem lentamente de acordo com  $bm^{-\beta}$ , quando  $m \rightarrow \infty$ , com  $0 < \beta < 1$ , sendo  $b$  uma constante positiva finita. Nesta análise é feita a subdivisão da série temporal original em blocos não sobrepostos de comprimento  $m$  e calcula-se a média de cada um, de modo idêntico ao do Método dos Valores Absolutos, obtendo-se a série  $X^{(m)}$ . Toma-se, então, a variância de  $X^{(m)}$ . Repetindo-se este procedimento para diversos valores de  $X^{(m)}$  e traçando-se o gráfico  $\log(\text{var}X^{(m)})$  versus  $\log(m)$  obtém-se pontos posicionados linearmente, para  $m$  grande. A seguir faz-se a aproximação da reta por mínimos quadrados, desprezando-se os valores pequenos de  $m$ . Se a inclinação  $-\hat{\beta}$  da reta estimada está entre  $-1$  e  $0$ , o processo é auto-similar com parâmetro (estimado)  $\hat{H} = 1 - \hat{\beta}/2$  [25, 26, 27].

#### 2.5.5. Análise do Periodograma

Consiste em calcular a densidade espectral de potência  $S(f)$  para processos agregados  $X^{(m)}$  e analisar o gráfico  $\log_{10}(S(f))$  versus  $\log_{10}(f)$ . Se na região de baixas frequências os pontos puderem ser aproximados por uma reta com inclinação negativa, então a densidade espectral de potência caracteriza um processo auto-similar. Como a inclinação da reta estimada é  $\hat{\gamma} = 1 - \hat{\beta}$ , pode-se calcular o parâmetro  $\hat{H} = 1 - \hat{\beta}/2 = (1 + \hat{\gamma})/2$ . Em geral, para a interpolação da reta, considera-se frequências de até 10% do total do espectro.

### 2.5.6. Análise da Estimativa de Máxima Verossimilhança Baseada em Periodograma

Esta análise é mais refinada que as descritas acima, pois permite estabelecer o intervalo de confiança para o parâmetro  $H$ . Para processos Gaussianos, a *estimativa de máxima verossimilhança* (MLE) com aproximação de Whittle é amplamente utilizada e tem propriedades estatísticas desejáveis [22].

A combinação da MLE de Whittle com a de agregação das amostras, utilizada nos métodos descritos acima, permite a obtenção do intervalo de confiança do parâmetro  $H$ , através do seguinte procedimento operacional:

- Para uma dada série temporal, considera-se os processos agregados  $X^{(m)}$ , com  $m = 100, 200, 300, \dots$ , onde o maior valor de  $m$  é escolhido de forma que as séries obtidas não tenham menos que 100 elementos aproximadamente;
- Para cada série agregada  $X^{(m)}$ , estima-se  $H$  através do método de Whittle, obtendo-se, assim,  $\hat{H}^{(m)}$  de  $H$  com os intervalos de confiança de 95% da forma  $\hat{H}^{(m)} \pm 1,96 \cdot \hat{\sigma}_H$ , onde  $\hat{\sigma}_H^2$  é dada por um conhecido resultado do Teorema Central do Limite [23];
- Traça-se o gráfico  $\hat{H}^{(m)}$  de  $H$  com os intervalos de confiança de 95% versus  $m$ , cujos pontos, tipicamente, apresentam bastante variação para  $m$  pequeno (baixo nível de agregação) mas depois se estabilizam e flutuam em torno de um valor constante, que é a estimativa final de  $H$  [25, 26].

## 2.6. Comprovação da Característica Auto-Similar do Tráfego da Rede Ethernet do Laboratório Bellcore

Esta seção apresenta os resultados mais importantes do estudo realizado no Laboratório Bellcore [25, 26] e uma comprovação da característica auto-similar do tráfego da rede Ethernet.

A Figura 2.1, apresenta as amostras do tráfego em cinco escalas de tempo diferentes. Para comparação, a Figura mostra o tráfego gerado sinteticamente a partir de um modelo Poissoniano, nas mesmas escalas. Esses gráficos mostram a contagem de pacotes (número de pacotes por unidade de tempo) em função do tempo. O gráfico superior é plotado a uma escala de 100 segundos como unidade de tempo. Os gráficos seguintes são obtidos do anterior aumentando-se a resolução temporal de um fator de 10 e focalizando-se em um sub-intervalo escolhido aleatoriamente (como indicado pelas áreas cinzas). Com isto, a unidade de tempo da escala mais fina (gráfico inferior) é de apenas 10 milissegundos.

Não é necessário fazer uma observação atenta para se verificar que todos os gráficos são estatisticamente semelhantes. Nota-se também que as escalas superior e inferior diferem de um fator relativamente alto ( $10^5$ ). Se a série temporal original  $X$  representar a contagem de pacotes Ethernet em unidades de tempo de 10 milissegundos, mostrado no gráfico (e), então os gráficos de (a) a (d) representam segmentos das séries temporais

agregadas  $X^{(10000)}$ ,  $X^{(1000)}$ ,  $X^{(100)}$  e  $X^{(10)}$ , respectivamente. Como os gráficos são bastante semelhantes, as suas funções de autocorrelação também devem ser semelhantes.

Para confirmar esta auto-similaridade aparente, o trabalho em questão apresenta também vários outros resultados quantitativos interessantes. Por exemplo, na Figura 2.2 é apresentado o “pox plot” da estatística de Hurst, onde o eixo vertical representa o logaritmo de  $R(t_i, n)/S(t_i, n)$  e o eixo horizontal o logaritmo de  $n$ . As duas retas mostradas (inferior e superior) correspondem a declividades de 0,5 e 1,0, respectivamente. O parâmetro de Hurst é facilmente estimado do “pox plot”, sendo da ordem de 0,79.

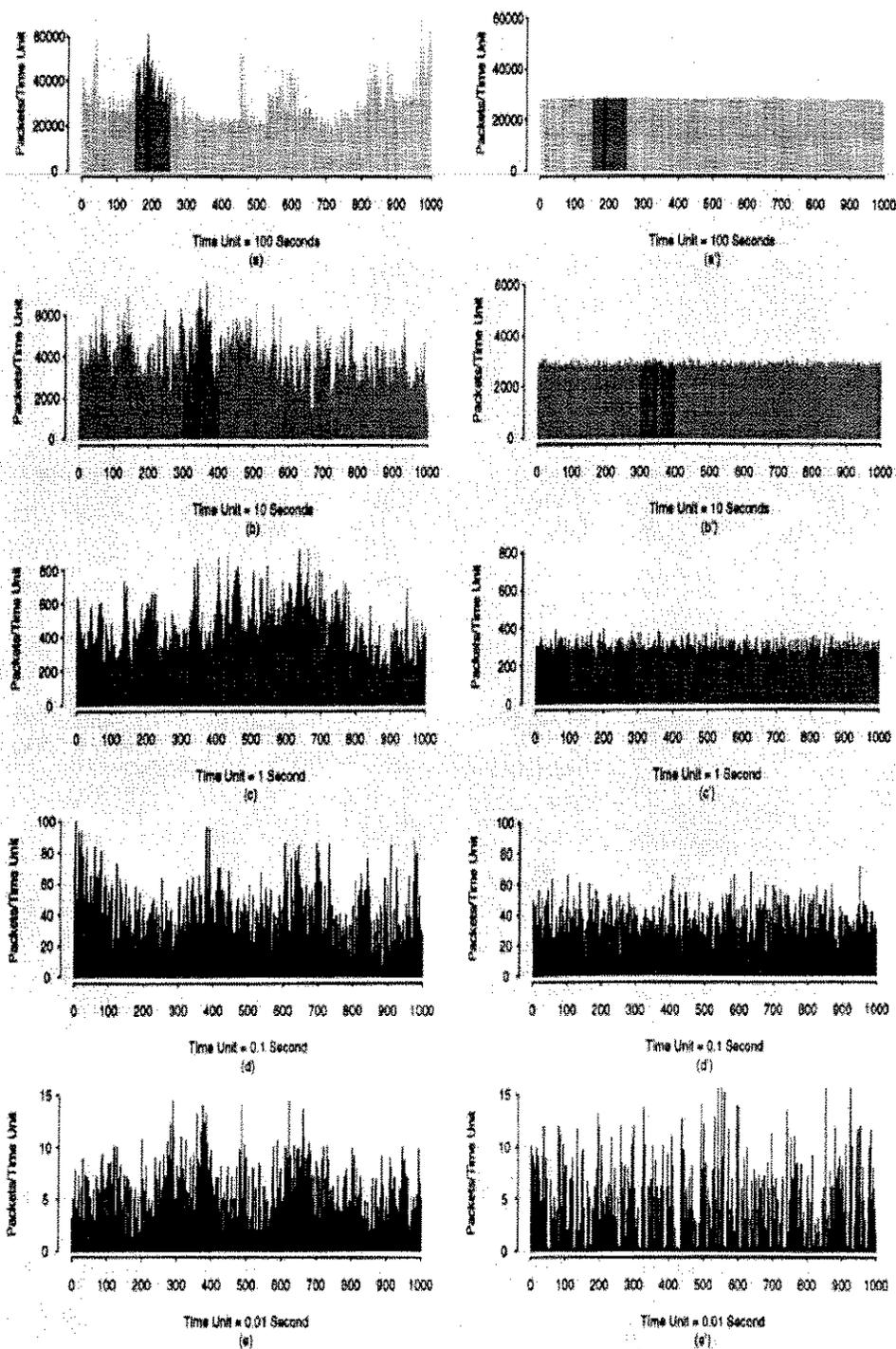


Figura 2.1: Resultados do tráfego Ethernet coletados no Bellcore (coluna da esquerda) comparados com o modelo de Poisson (coluna da direita). Escalas de tempo variam de 100 segundos (a) até 0,01 segundos (e).

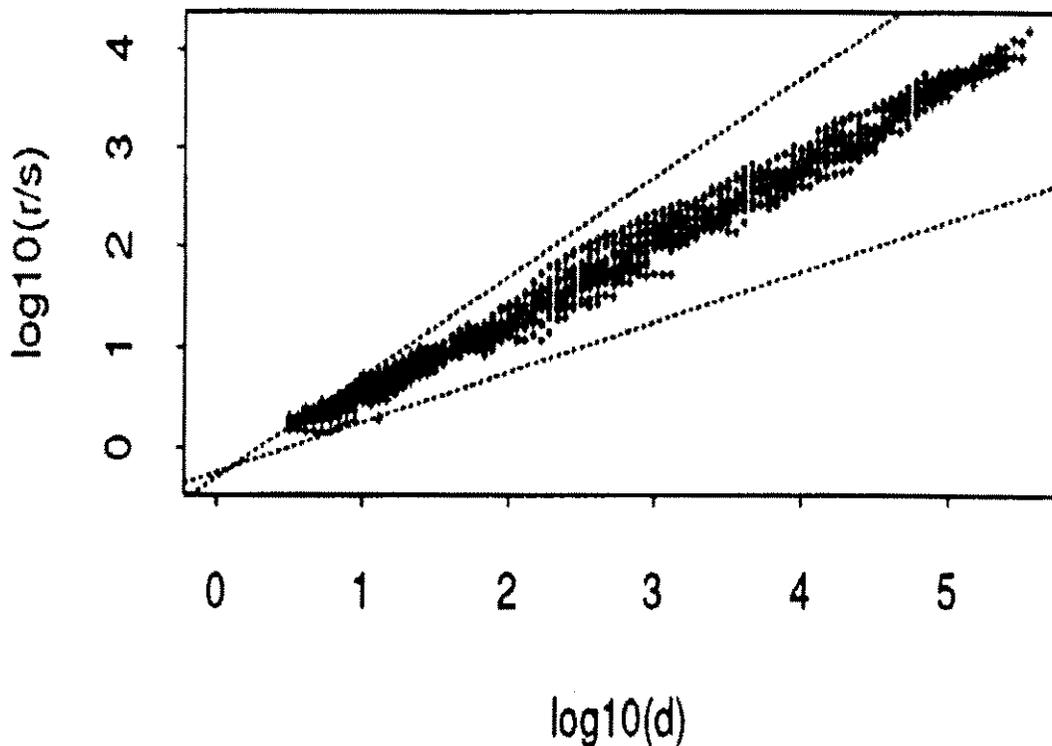


Figura 2.2: “Pox plot” de  $R/S$  em função de  $n$  (escala  $\log - \log$ ) para os dados do Bellcore. A declividade de uma reta interpolada no diagrama corresponde a  $H = 0,79$ .

Dessa forma, fica confirmada a forte característica auto-similar do tráfego na rede Ethernet. Estudo semelhante foi realizado para o tráfego na rede Internet com protocolo TCP/IP, resultando no mesmo comportamento estatístico. Além disso, resultados semelhantes foram também obtidos para o tráfego de sinais de vídeo codificados com Taxa de Bit Variável (VBR). Tendo em vista que estes tipos de sinais devem ser os mais comumente utilizados na rede ATM, como mencionado anteriormente, é de se esperar que o tráfego global agregado nessas redes, seja também altamente granular e com parâmetros de Hurst em torno de  $(0,75)$ .

A divulgação desses resultados nos últimos anos, sem dúvida criou uma grande expectativa na comunidade de redes de faixa larga, pois a premissa básica da regularidade estatística, discutida acima, foi colocada em cheque. A confrontação desses resultados com os modelos clássicos Poissonianos, deve merecer uma atenção especial para que essas redes de alta capacidade possam operar de forma segura e eficiente.

A seguir são apresentados vários resultados obtidos de gráficos do diagrama “pox” e da variância utilizando as amostras de tráfego interno e externo (arquivos pOct.TL e OctExt.TL do Bellcore), referentes a outubro de 89, obtidas via ftp no endereço flash.bellcore.com. O arquivo pOct.TL possui um milhão de chegadas de pacotes no cabo principal da rede, correspondendo a 1.759,62 segundos aproximadamente, monitoradas a partir de 11:00 horas do dia 5/10/89. O arquivo OctExt.TL possui um

milhão de chegadas de pacote para tráfego externo, correspondendo a 122.797,83 segundos aproximadamente, monitoradas a partir de 23:46 horas do dia 3/10/89. Ambos os arquivos têm formato ASCII, sendo que cada chegada de pacote é registrada em uma linha de 20 bytes. Cada linha possui duas entradas; a primeira referente ao tempo em segundos, contado a partir do início da monitoração, e a segunda ao comprimento do pacote em bytes, excluindo-se o preâmbulo, o cabeçalho ou CRC. Os bytes monitorados correspondem a pacotes completos, não incluindo fragmentos de colisões.

### 2.6.1. Diagrama “Pox” da Estatística R/S

Para a elaboração deste diagrama foi utilizado o método descrito na subseção 2.4.1. Os programas, elaborados no Mathematica 2.2 e no Matlab 4.0, são apresentados no Apêndice B. A Figura 2.3 mostra o diagrama obtido para as primeiras 10.000 amostras de tráfego interno.

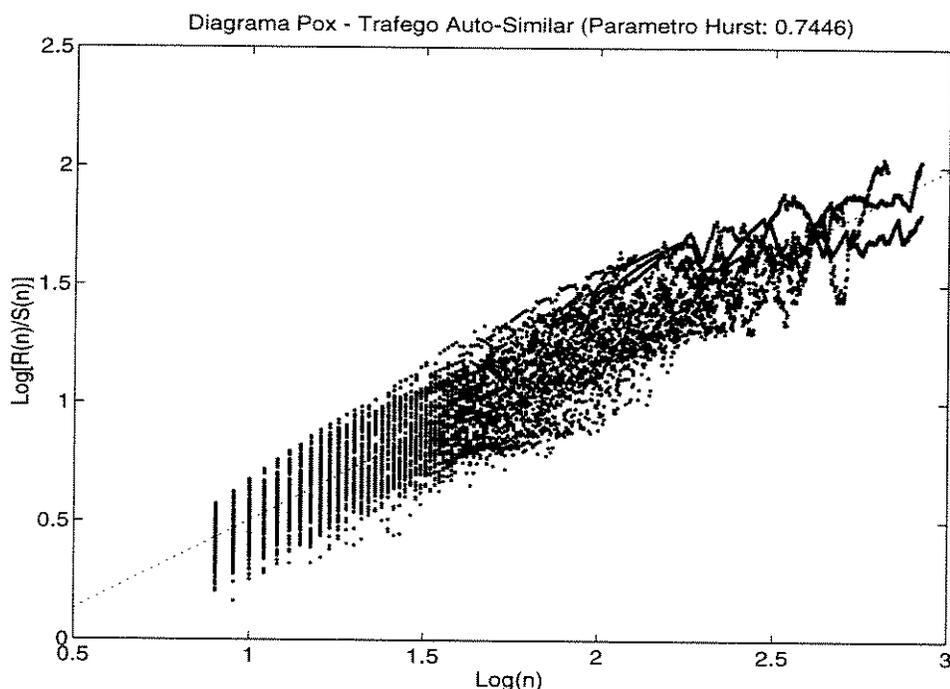


Figura 2.3: Diagrama “pox” obtido com as primeiras 10.000 amostras do arquivo pOct.TL do Bellcore.

Observa-se que o parâmetro de Hurst calculado corresponde realmente a um processo auto-similar, com  $H = 0,7446$ . Para efeito de comparação, a Figura 2.4 apresenta o diagrama “pox” para tráfego Poissoniano gerado artificialmente no MATLAB. O parâmetro de Hurst obtido neste caso está próximo de 0,5, caracterizando um processo com incrementos independentes, bem diferente do tráfego da rede Ethernet do Bellcore.

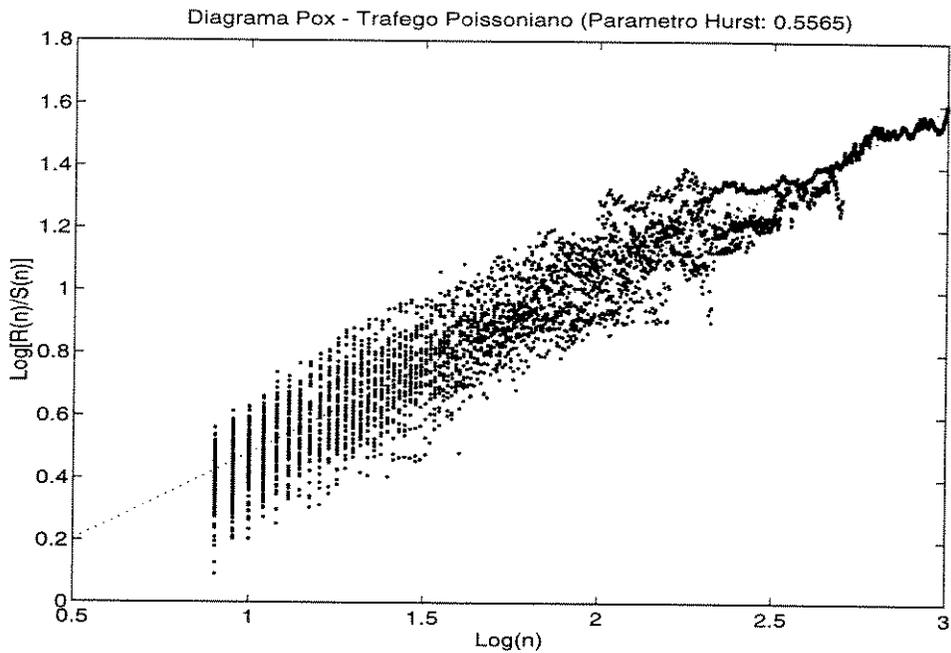


Figura 2.4: Diagrama “pox” para tráfego Poissoniano gerado artificialmente.

### 2.6.2. Gráfico da Variância

As Figuras 2.5 e 2.6 mostram os gráficos das variâncias elaborados para os tráfegos interno e externo do Bellcore, respectivamente, conforme descrito na subseção 2.4.4. O programa implementado no MATLAB 4.0 é apresentado no Apêndice B.

Observa-se que ambos os gráficos dão indícios claros de um processo auto-similar. Para o tráfego interno a reta aproximada tem inclinação  $-\hat{\beta} = -0,6766$  e, portanto, o parâmetro de Hurst estimado é  $\hat{H} = 1 - \hat{\beta}/2 = 0,6617$ . Apesar desse valor ser um pouco menor que o encontrado através do Diagrama “pox” da Figura 2.3, ele é aceitável, pois, na prática, os resultados obtidos por métodos diferentes apresentam pequenas variações [25, 26]. Para o tráfego externo, tem-se  $-\hat{\beta} = -0,2727$  e  $\hat{H} = 0,8637$ .

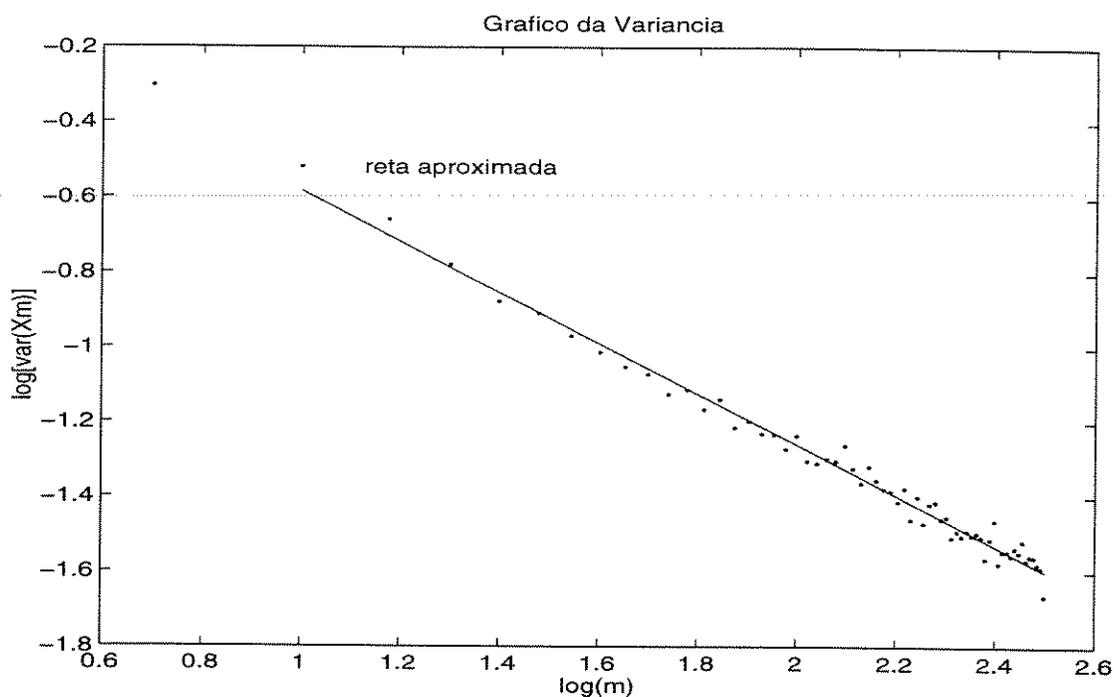


Figura 2.5: Gráfico da variância para as primeiras 10.000 amostras de tráfego do arquivo pOct.TL do Bellcore (tráfego interno).

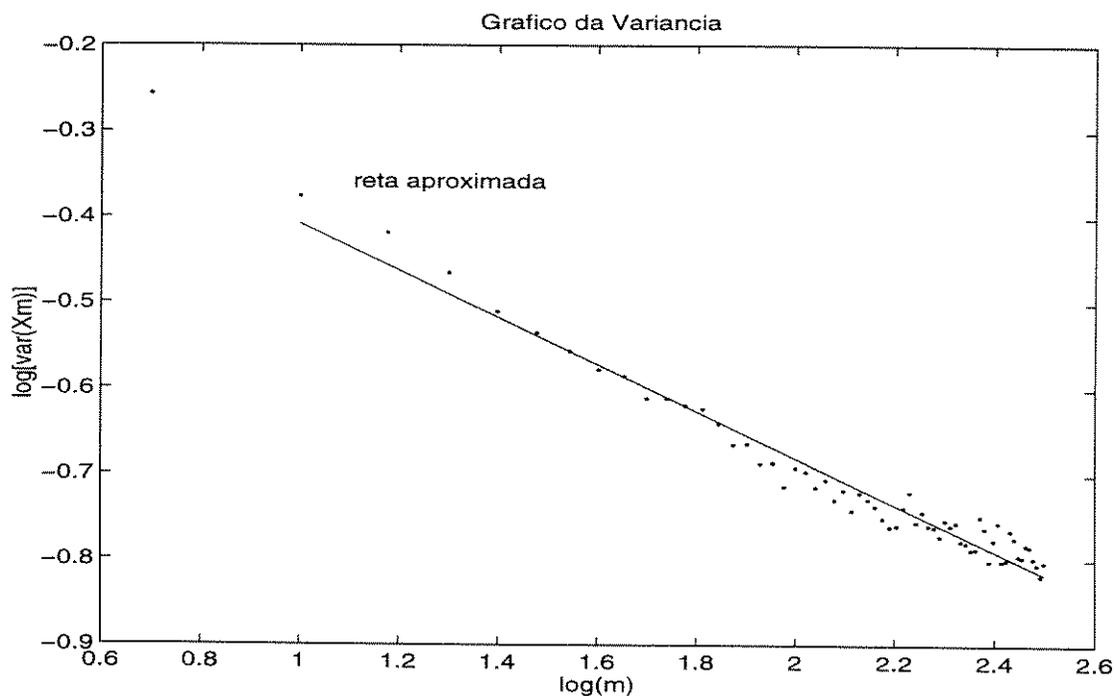


Figura 2.6: Gráfico da variância para as primeiras 10.000 amostras de tráfego do arquivo OctExt.TL do Bellcore (tráfego externo).

## Capítulo 3

# Sinais com Taxa de Bit Variável em Redes de Faixa Larga

### 3.1. Introdução

Este capítulo tem por objetivo verificar o comportamento auto-similar de sinais de imagens paradas e de vídeo produzidas por fontes VBR tributárias em uma rede de faixa larga. Antes, porém, relembra-se alguns conceitos da Teoria da Informação no que diz respeito ao problema da compressão de dados.

### 3.2. Limitantes para os Algoritmos de Compressão

Nesta seção discute-se os limitantes fundamentais para o problema da compressão de dados.

Seja uma fonte de informação estacionária (ou mesmo não-estacionária, mas com uma certa regularidade estatística), que produz um sinal à taxa de  $R^*$  bits/s. Supõe-se que esta fonte é digital ou analógica convertida para digital sem redução de redundância. Isto é, o valor de  $R^*$  representa a taxa bruta de geração de bits. Seja  $H_b$  a entropia binária dessa fonte que, no caso mais geral, pode gerar símbolos correlacionados (fonte com memória) [51].

A curva limitante fundamental para a compressão de uma fonte desse tipo é dada pela Função Taxa-Distorção  $R(D)$  [52], onde  $D$  é a distorção média admitida no processo de compressão. A Figura 3.1 mostra uma curva  $R(D)$  típica para a maioria das fontes práticas.

Nota-se que a compressão *sem perdas* ( $D = 0$ ) é conseguida para valores de taxa  $R(0)$  tais que  $H_b < R(0) < R^*$ . Compressão com taxa  $R(D) < H_b$  só é conseguida se for admitida uma distorção  $D > 0$ , representando uma perda de informação.

Em muitos casos práticos, deseja-se comprimir não a saída contínua (ou quase-contínua) de uma fonte, mas um sinal confinado em um intervalo de tempo ou em uma região de memória (um arquivo de texto ou de imagem, por exemplo). Naturalmente, um sinal desse tipo também pode ser compactado com ou sem perdas.

A Figura 3.2 apresenta os limitantes para sinais deste tipo, onde  $N^*$  representa o

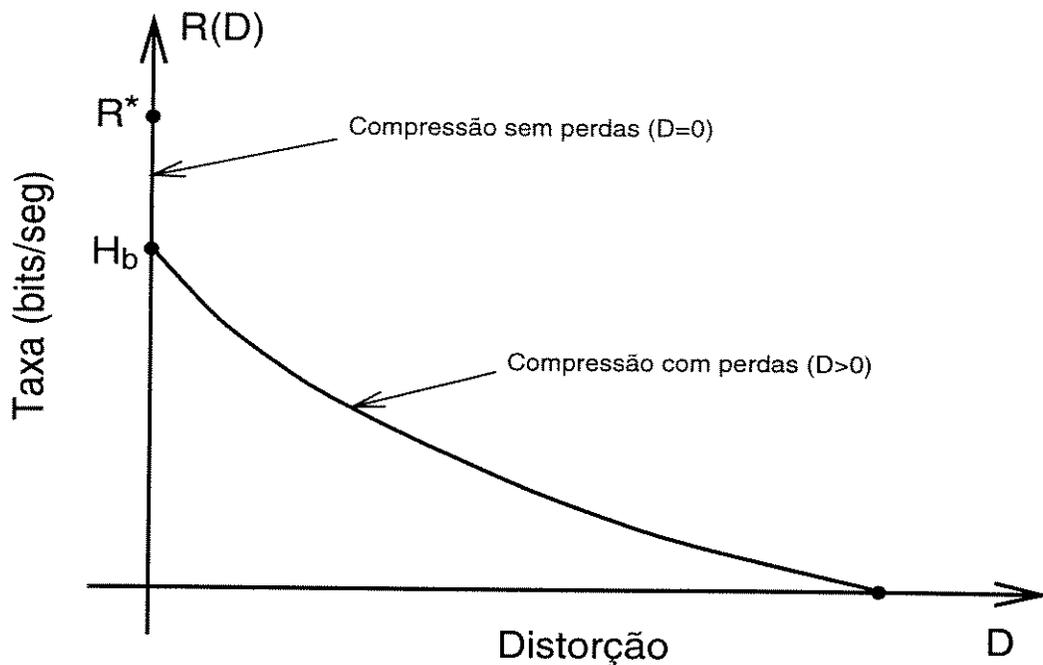


Figura 3.1: Curva típica de uma Função Taxa-Distorção.

tamanho do sinal original,  $N(D)$  representa o tamanho do sinal após a compressão com distorção  $D \geq 0$  e  $K$  é um parâmetro relacionado à chamada *Complexidade de Kolmogorov* ( $K_c$ ) [51]. A complexidade  $K_c$  representa, formalmente, o tamanho do *menor programa binário* que codifica o sinal em questão de forma reversível, isto é, sem perdas. Assim,  $K < N(0) < N^*$  corresponde à região de compressão sem perdas e  $N(D) < K$  à região de compressão com perdas.

As técnicas de compressão sem perdas normalmente usadas nas plataformas de computação, em geral utilizam o algoritmo de Lempel-Ziv [51], com o qual pode-se obter compressão com eficiência próxima do limitante  $K$ .

Na próxima seção será quantificado o conteúdo de informação de um bloco de imagem monocromática ( $4 \times 4$  ou  $8 \times 8$  pixels, por exemplo). Neste caso, a complexidade de Kolmogorov  $K_c$  poderia representar a informação intrínseca desses blocos de imagens. Infelizmente, este parâmetro de complexidade é mais um conceito formal do que propriamente um valor que se possa calcular de forma sistemática, como a entropia de Shannon, por exemplo.

Em razão dessa dificuldade, será utilizada como medida de informação desses blocos de imagens, a *variância dos níveis de cinza* dos respectivos pixels. Este parâmetro tem sido utilizado em processamento de imagens com bons resultados, tendo a vantagem da simplicidade de cálculo. A *entropia dos níveis de cinza*, supostos independentes, é outro parâmetro normalmente utilizado, mas a complexidade de cálculo é maior que a da variância e os resultados não são necessariamente superiores.

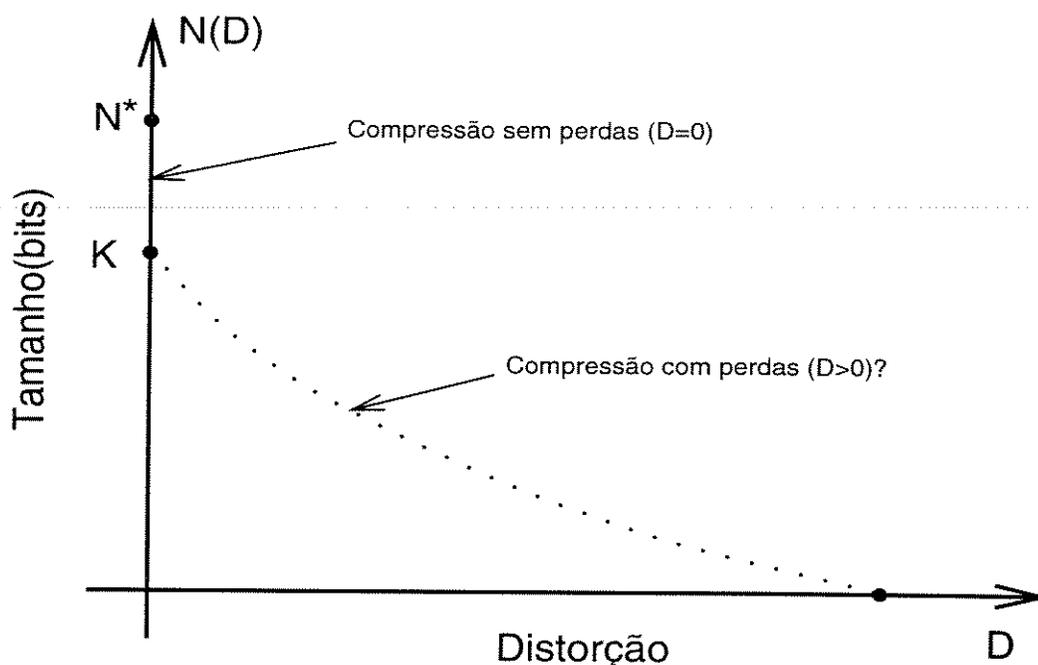


Figura 3.2: Curva típica de compressão para sinais confinados, caracterizada pela Complexidade de Kolmogorov.

### 3.3. Análise do Tráfego Agregado Gerado por Imagens Monocromáticas

O objetivo desta seção é estudar a presença de auto-similaridade no tráfego agregado gerado por imagens monocromáticas isoladas. A preocupação não é tanto simular realisticamente o tráfego de entrada nos multiplexadores estatísticos, mas analisar a *auto-similaridade intrínseca* que, segundo estudos já realizados [29], também está presente nos sinais de vídeo codificados com taxa de bit variável (VBR). Verifica-se que a auto-similaridade do sinal à saída de um codificador VBR provém da própria natureza fractal dos sinais originais. A agregação dos sinais tende a reforçar a auto-similaridade, o que não parece muito intuitivo mas que pode ser observado facilmente.

A fim de realizar uma análise não-polarizada, obteve-se de forma aleatória, via Internet, mais de uma centena de imagens tipicamente transmitidas nessa rede. Todo o estudo aqui apresentado se baseia nesse banco de dados de imagens monocromáticas, recortadas para um tamanho único de  $16 \times 256$  pixels. Apenas como curiosidade, a Figura 3.3 apresenta um subconjunto típico dessas imagens.

Para efeitos práticos, um sinal agregado à saída de um multiplexador estatístico pode ser representado por uma série temporal cujos valores são os fluxos instantâneos de células. Por outro lado, existe uma proporcionalidade entre este fluxo e as atividades dinâmicas (ou conteúdos de informação) instantâneas das imagens que originaram as células. Dessa forma, essas séries temporais serão utilizadas para representar os valores instantâneos da informação em cada ponto da imagem. Estes valores, por sua vez, serão representados pela variância dos pixels nos respectivos blocos de  $4 \times 4$  ou  $8 \times 8$ , conforme

discutido anteriormente.

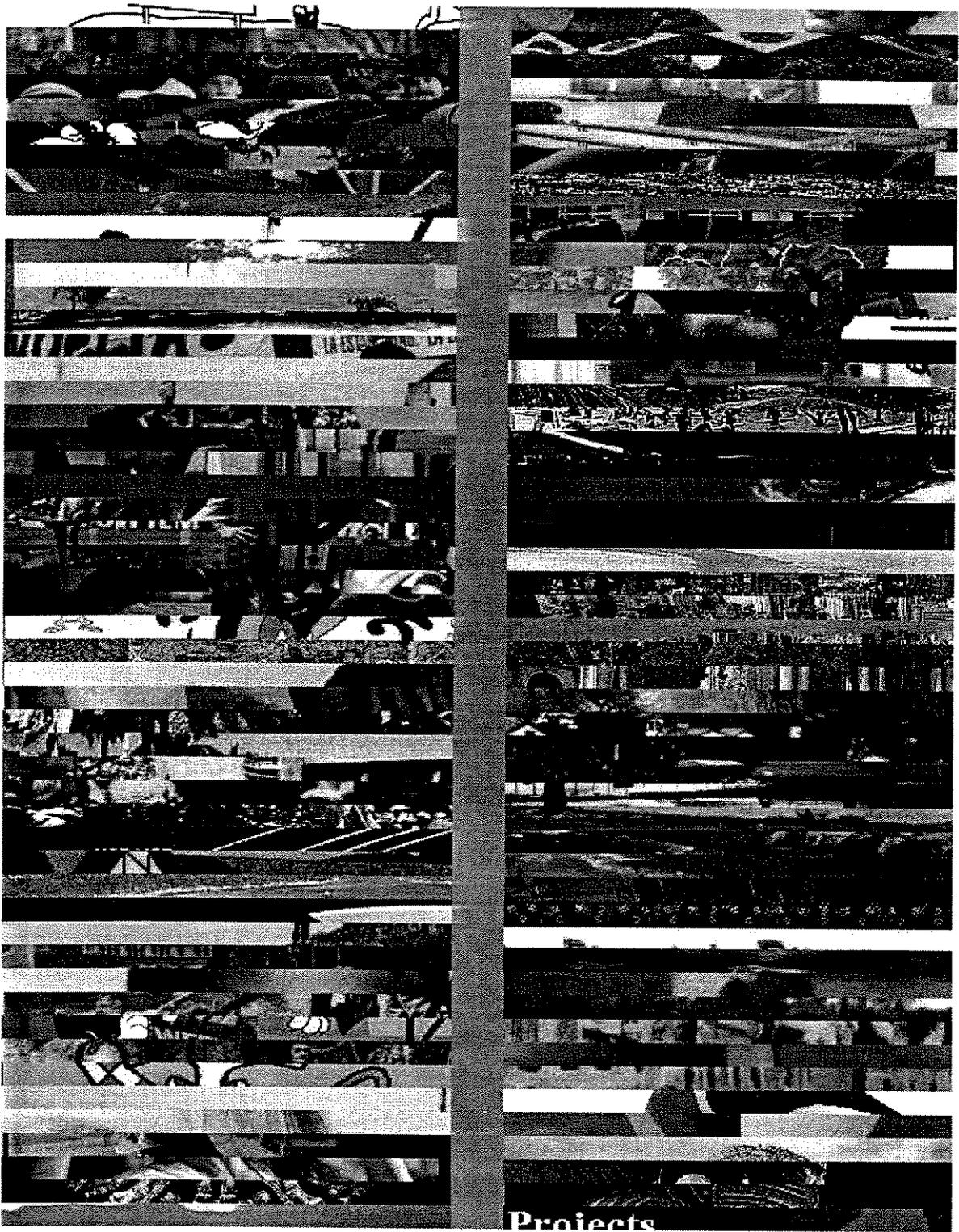


Figura 3.3: Uma amostra de imagens recortadas que tipicamente são transmitidas na rede Internet.

O processo de agregação dos *valores de atividades* das imagens (que produzirá a série temporal desejada na saída do multiplexador estatístico) será simulado de duas formas diferentes, a *semi-paralela* e a *semi-seqüencial*.

Na forma semi-paralela,  $N$  imagens são dispostas lado a lado, e a leitura dos valores de atividade, ou seja, variâncias, já calculadas em etapa anterior (ver Apêndice B), é feita tomando-se um bloco de cada imagem de cada vez, conforme mostra a Figura 3.4. A cada  $N$  blocos lidos, o ponteiro retorna à primeira imagem a fim de ler o seu bloco imediatamente à direita do último bloco lido nessa imagem. Assim sucessivamente, até que seja completada a série temporal com  $N \times 4 \times 64$  valores, para blocos de  $4 \times 4$ , ou  $N \times 2 \times 32$  valores, para blocos de  $8 \times 8$ .

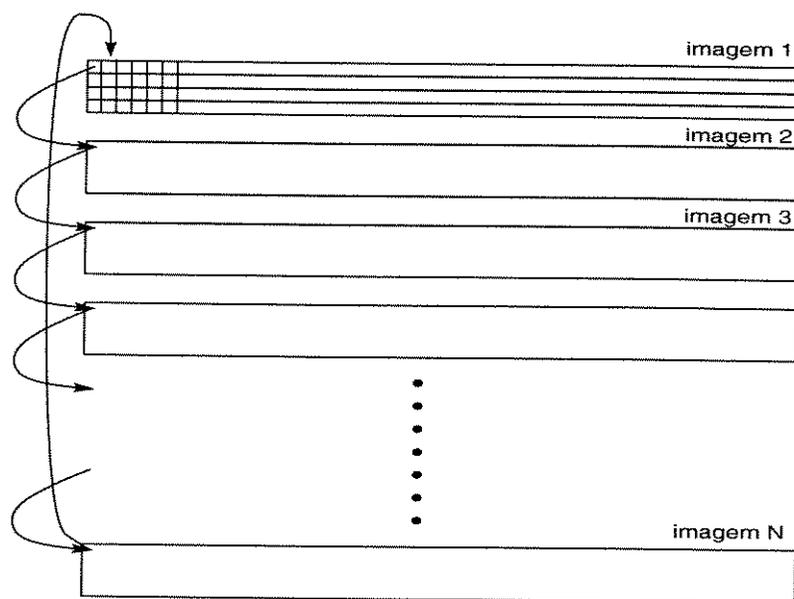


Figura 3.4: Forma semi-paralela para agregação das atividades das imagens.

Na forma semi-seqüencial, a leitura das  $N$  imagens é feita linha por linha, como mostra a Figura 3.5. Ao se esgotar a leitura de uma linha, o ponteiro retorna ao início da próxima linha da imagem seguinte, e assim por diante até que todos os blocos sejam lidos. Neste caso, a série temporal obtida possui o mesmo tamanho da anterior ( $N \times 4 \times 64$  ou  $N \times 2 \times 32$ ).

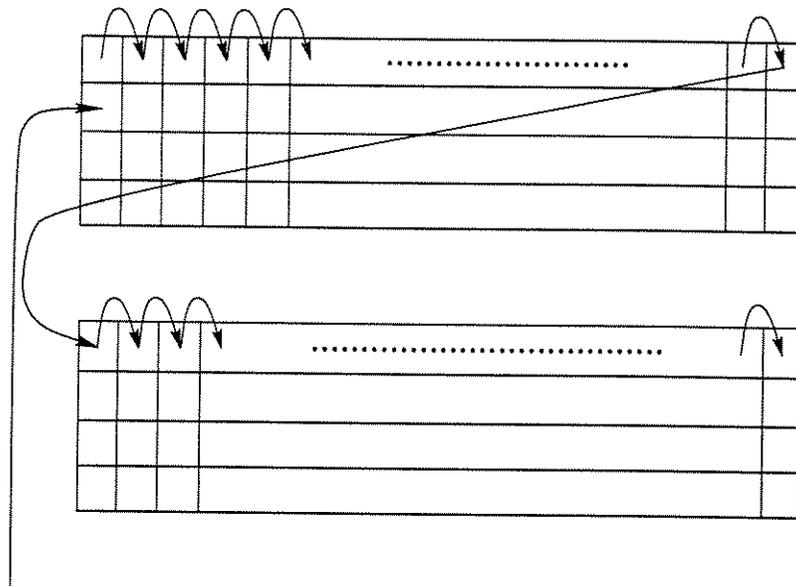


Figura 3.5: Forma semi-seqüencial para agregação das atividades das imagens.

### 3.3.1. Visualização do sinal agregado em diferentes escalas

Inicialmente, é interessante observar, em diferentes escalas, as séries agregadas descritas na seção anterior, tal como foi feito para o tráfego Ethernet do Laboratório Bellcore [25, 26] (ver Apêndice A).

Considera-se um conjunto de  $N = 104$  imagens submetidas ao processo de agregação semi-paralela. A Figura 3.6 mostra todos os  $104 \times 4 \times 64 = 26.624$  pontos obtidos. Nas Figuras 3.7, 3.8, 3.9, 3.10 e 3.11 são apresentadas as séries agregadas em diferentes níveis de resolução. A Figura 3.7 mostra a série obtida da Figura 3.6 para agregação de 64 pontos, isto é, a série original foi agregada com  $m = 64$ . Um trecho entre os pontos 100 e 307 dessa nova série é expandido e plotado na Figura 3.8, correspondendo a uma agregação com  $m = 32$  sobre a série original. Um novo trecho é extraído e agregado entre os pontos 140 e 347 da última série obtida, como mostrado na Figura 3.9. Dois outros trechos sucessivos são novamente expandidos e mostrados nas Figuras 3.10 e 3.11, respectivamente.

Apesar do tamanho relativamente pequeno dessa série, a observação dessas figuras mostra que o fenômeno da auto-similaridade está claramente presente no sinal agregado. O processo de “zoom”, aqui descrito, indica que esse sinal possui detalhes auto-similares em vários níveis de resolução.

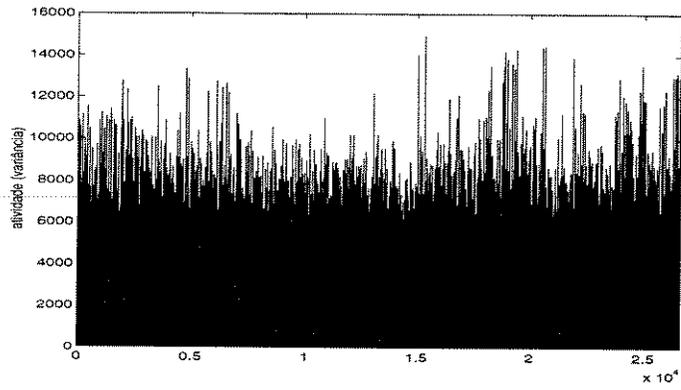


Figura 3.6: Série temporal completa obtida por agregação semi-paralela (26.624 pontos).

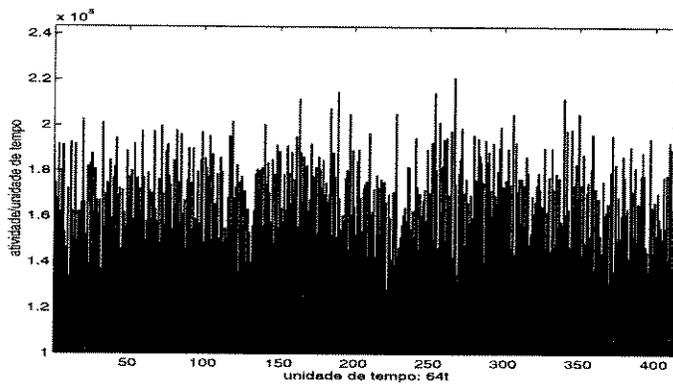


Figura 3.7: Série agregada a partir da original, com  $m = 64$ .

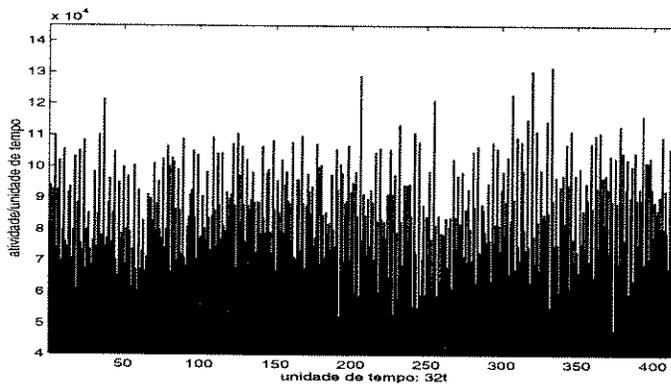


Figura 3.8: Trecho ampliado do trecho anterior, agregado com  $m = 32$ .

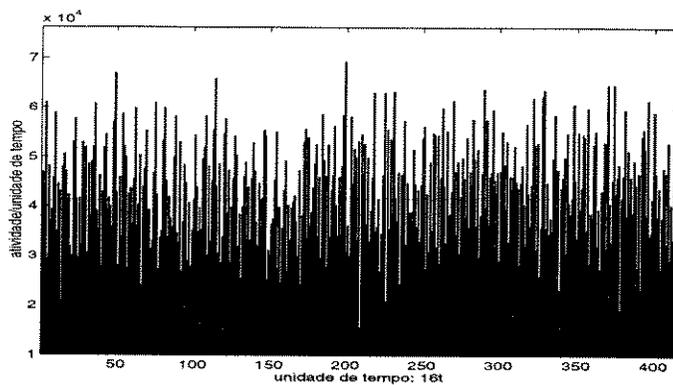


Figura 3.9: Trecho ampliado do trecho anterior, agregado com  $m = 16$ .

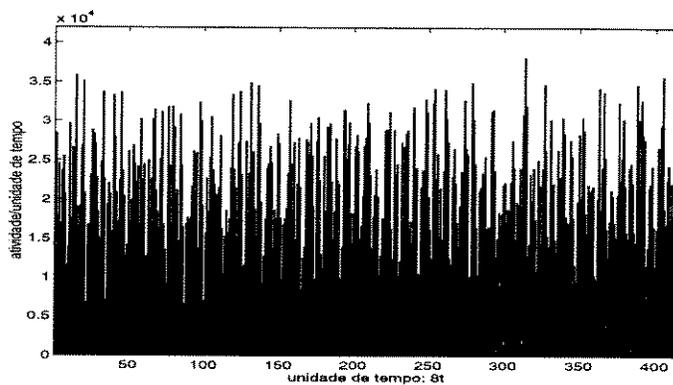


Figura 3.10: Trecho ampliado do trecho anterior, agregado com  $m = 8$ .

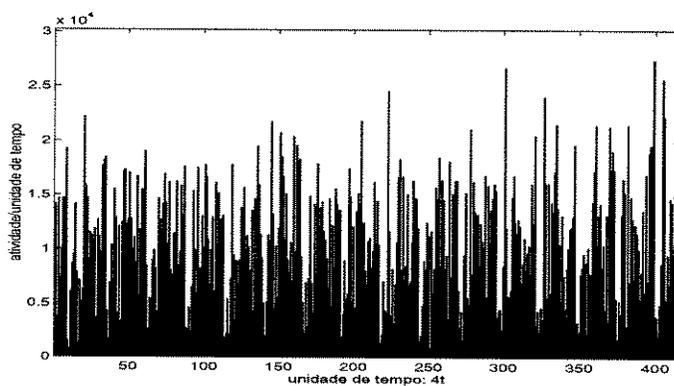


Figura 3.11: Trecho ampliado do trecho anterior, agregado com  $m = 4$ .

O mesmo procedimento foi repetido para o processo de agregação semi-seqüencial, resultando na série completa (não agregada) da Figura 3.12. As Figuras 3.13, 3.14, 3.15, 3.16 e 3.17 mostram os resultados para agregações com  $m = 64, 32, 16, 8$  e  $4$ , respectivamente. Aqui a característica auto-similar também se manifesta claramente.

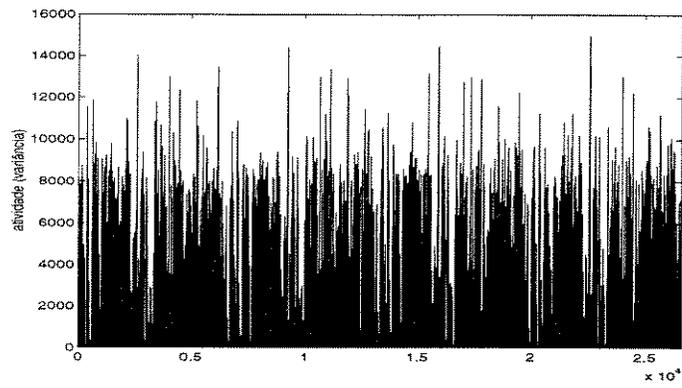


Figura 3.12: Série temporal completa obtida por agregação semi-seqüencial (26.624 pontos).

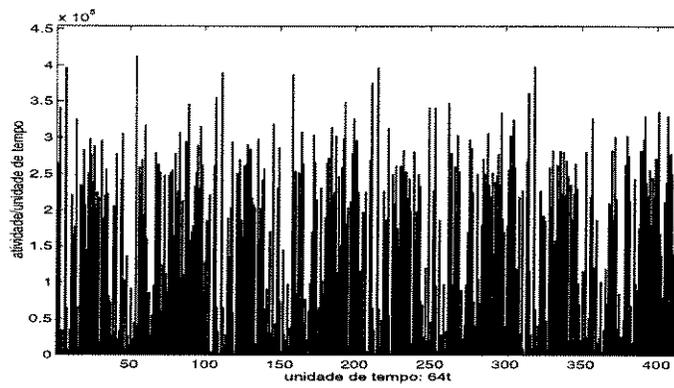


Figura 3.13: Série agregada a partir da original, com  $m = 64$ .

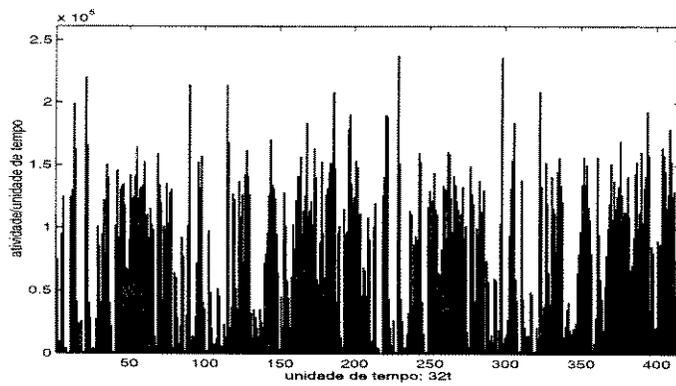


Figura 3.14: Trecho ampliado do trecho anterior, agregado com  $m = 32$ .

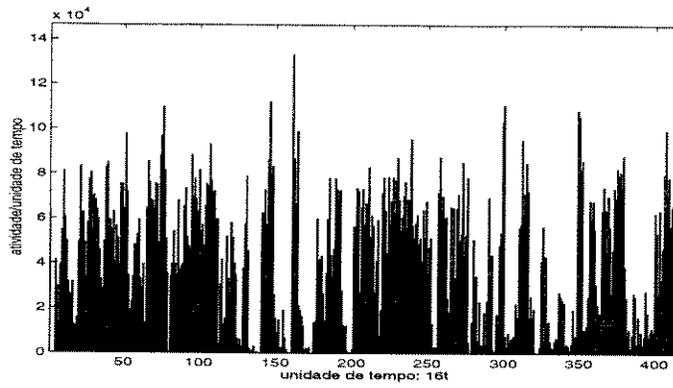


Figura 3.15: Trecho ampliado do trecho anterior, agregado com  $m = 16$ .

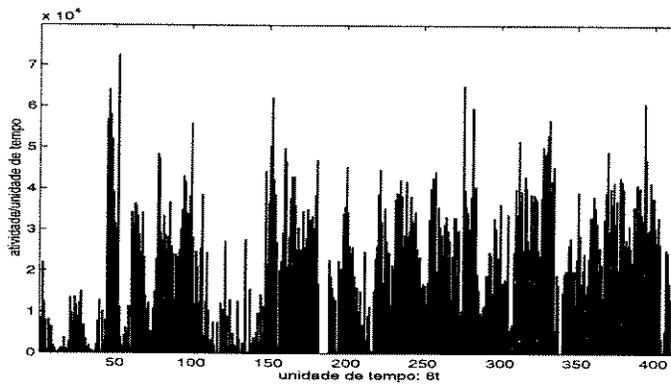


Figura 3.16: Trecho ampliado do trecho anterior, agregado com  $m = 8$ .

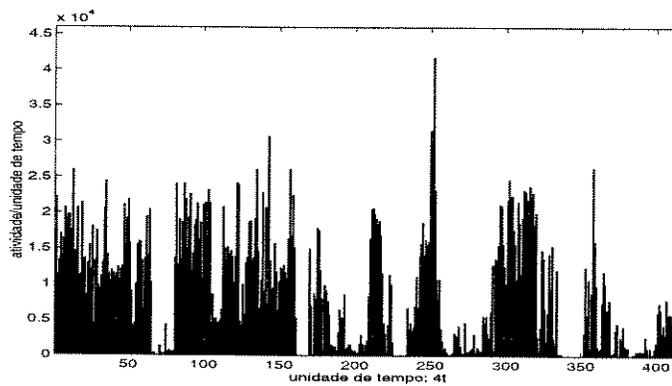


Figura 3.17: Trecho ampliado do trecho anterior, agregado com  $m = 4$ .

### 3.3.2. O parâmetro de Hurst na agregação semi-paralela

Considera-se, inicialmente, um pequeno conjunto com  $N = 53$  imagens e blocos de  $4 \times 4$  pixels, resultando em uma série com 13568 valores ( $53 \times 4 \times 64$ ). Para o cálculo do

parâmetro de Hurst ( $H$ ) da série de agregação semi-paralela, podemos utilizar o método da análise das variâncias, descrito na subseção 2.4.4, que é rápido e simples.

Na Figura 3.18 é apresentado o gráfico da variância para vários fatores de agregação ( $m$ ). Nota-se que este gráfico foi obtido por uma nova forma de agregação realizada sobre a série já agregada no modo semi-paralelo. Pode-se observar que a variância decresce com  $m$ , mas há uma pequena oscilação a partir de  $m = 53$ , que equivale a  $\log(53) \approx 1,72$  no eixo horizontal. Isto pode ser explicado pela própria periodicidade do processo de agregação semi-paralela, que se repete a cada 53 pontos. Obviamente, este fenômeno é mais evidente nas imediações de  $\log(53)$ , mas depois desvanece à medida que  $m$  cresce.

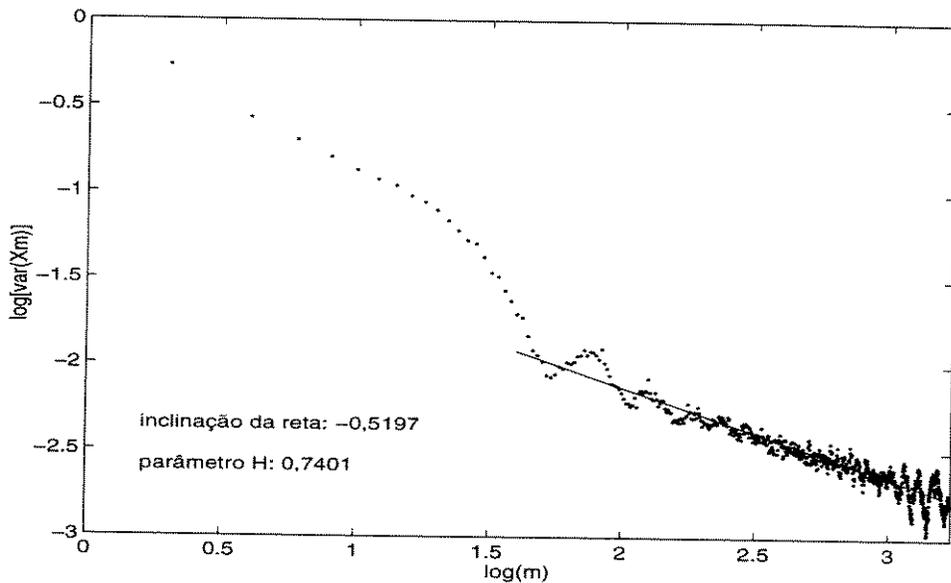


Figura 3.18: Gráfico da variância para o tráfego agregado de 53 imagens (modo semi-paralelo).

O fenômeno de Hurst é captado pela região mais densa desse gráfico. Uma reta interpolada pelo método dos mínimos quadrados nessa região tem declividade  $-\beta = -0,5197$ , que corresponde a um parâmetro de Hurst  $H = 1 - \beta/2 \approx 0,7401$ , indicando uma forte tendência auto-similar.

Observa-se também que fatores de agregação menores que o número de imagens (53 neste caso) levam a  $\log[\text{Var}(X^{(m)})]$  com maiores decréscimos, correspondendo a uma reta interpolada nessa região com declividade  $-\beta \approx -1$ , levando a  $H = 1 - \beta/2 \approx 0,5$ . Mas este valor equivale a um processo com amostras quase decorrelacionadas, o que é coerente, pois neste caso elas pertencem a *imagens distintas e decorrelacionadas* (excetuando-se as fronteiras entre períodos vizinhos).

Resumindo, pode-se constatar que o valor de  $m = 53$  ( $\log(m) \approx 1,72$ ) divide o gráfico da Figura 3.18 em duas regiões bem distintas; uma com dados decorrelacionados ( $H \approx 0,5$ ) e outra com dependência de longo prazo, caracterizada pelo efeito Hurst ( $H \approx 0,74$ ).

Os cálculos acima foram repetidos utilizando-se 51 imagens adicionais, totalizando 104 imagens que levam a uma série com 26.624 valores. A Figura 3.19 mostra o gráfico

da variância para este caso. Aqui também fica bastante evidenciada a dicotomia entre as duas regiões, neste caso em torno de  $\log(104) \approx 2,01$ . A oscilação logo à direita desse ponto também está presente, mas agora bastante atenuada em função do maior número de imagens utilizado. Pode-se verificar também que o período de oscilação aumentou em relação ao caso anterior. Mas isto era de se esperar, pois este período é proporcional ao número de imagens.

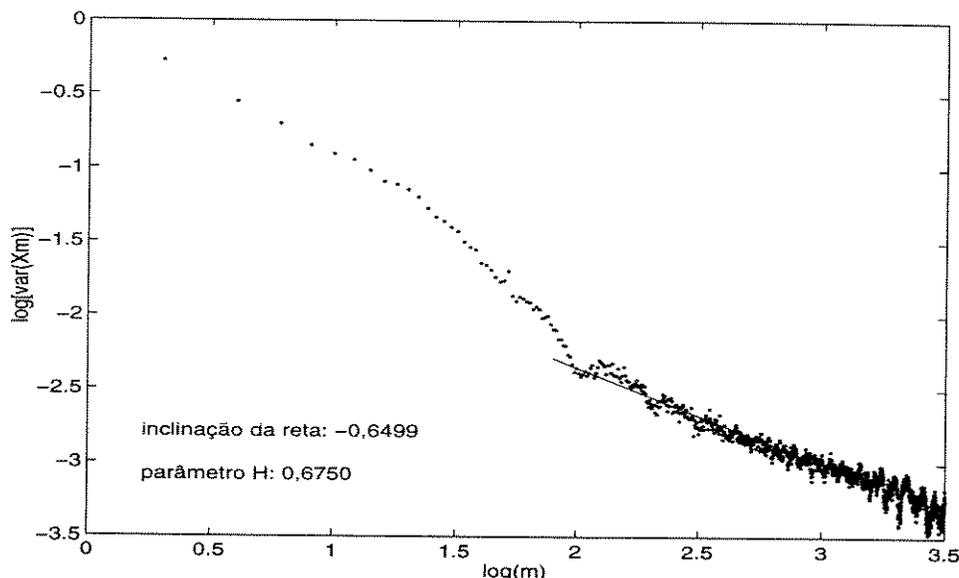


Figura 3.19: Gráfico da variância para o tráfego agregado de 104 imagens no modo semi-paralelo, com reta interpolada em  $80 < m < 1.586$ .

A reta interpolada na região entre  $m = 80$  e  $m = 1.586$  ( $\log(m)$  entre 1,9 e 3,2) tem inclinação de  $-\beta = -0,6499$ , correspondendo ao parâmetro  $H = 0,6750$ . Assim como no caso anterior, a reta interpolada na região onde  $m$  é menor que 100, aproximadamente, corresponde a  $H = 0,5$ , indicando também uma decorrelação entre amostras.

A Figura 3.20 apresenta um trecho expandido da Figura 3.19, para  $m$  entre 252 e 1586 ( $2,4 < \log(m) < 3,2$ ). Esta é a região que melhor capta o efeito Hurst do sinal agregado, pois elimina a transição em torno de  $\log(m) \approx 2$  e outros pontos referentes a fatores de agregação maiores que 1600, aproximadamente, para as quais as séries  $X^{(m)}$  possuem poucos termos (14 ou menos), tornando o gráfico da variância pouco confiável.

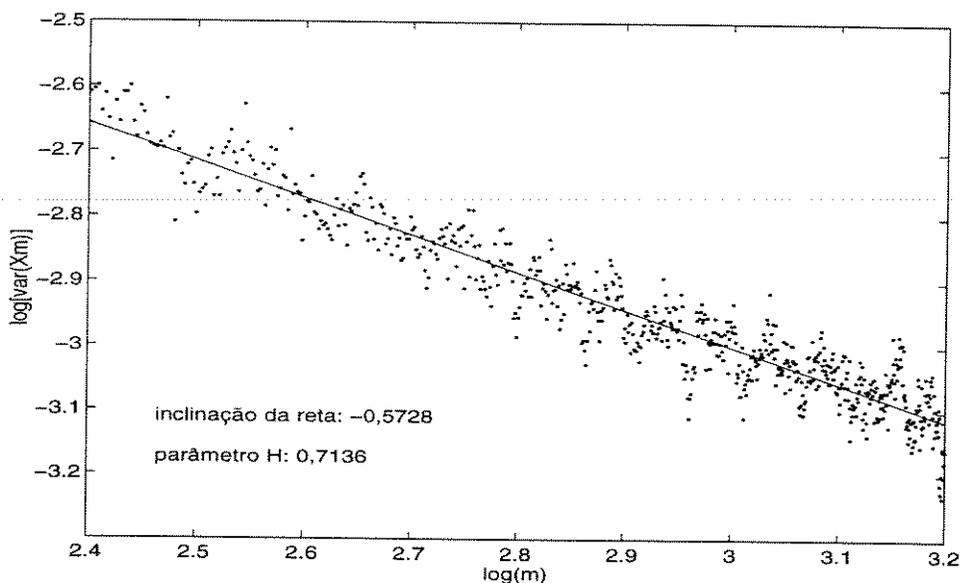


Figura 3.20: Trecho expandido da Figura 3.19 , para  $252 < m < 1.586$ .

A declividade da reta interpolada na Figura 3.20 foi de  $-\beta = -0,5728$ , levando a  $H = 0,7136$ , valor que é mais confiável que o anterior, de  $0,6371$ .

Como exemplo final para o caso de agregação semi-paralela com 104 imagens, os cálculos anteriores foram repetidos tomando-se agora blocos de  $8 \times 8$  pixels. A série resultante tem 6.656 amostras ( $104 \times 2 \times 32$ ).

Os resultados dessa simulação são apresentados na Figura 3.21, onde se pode observar os mesmos fenômenos dos casos anteriores. A reta interpolada na região  $100 < m < 630$  ( $2 < \log(m) < 2,8$ ) corresponde a um valor de  $H = 0,6052$ . Este valor foi menor que o anterior por dois motivos. Primeiro porque a série agregada não é suficientemente longa para permitir uma estabilização no cálculo das variâncias de  $X^{(m)}$ . Como segunda causa, a região de interpolação da reta engloba parte da região de transição e de oscilação e, além disso, não consegue captar a parte onde o efeito Hurst se sobressai. Assim, este exemplo mostra os cuidados que se deve tomar na análise de séries auto-similares.

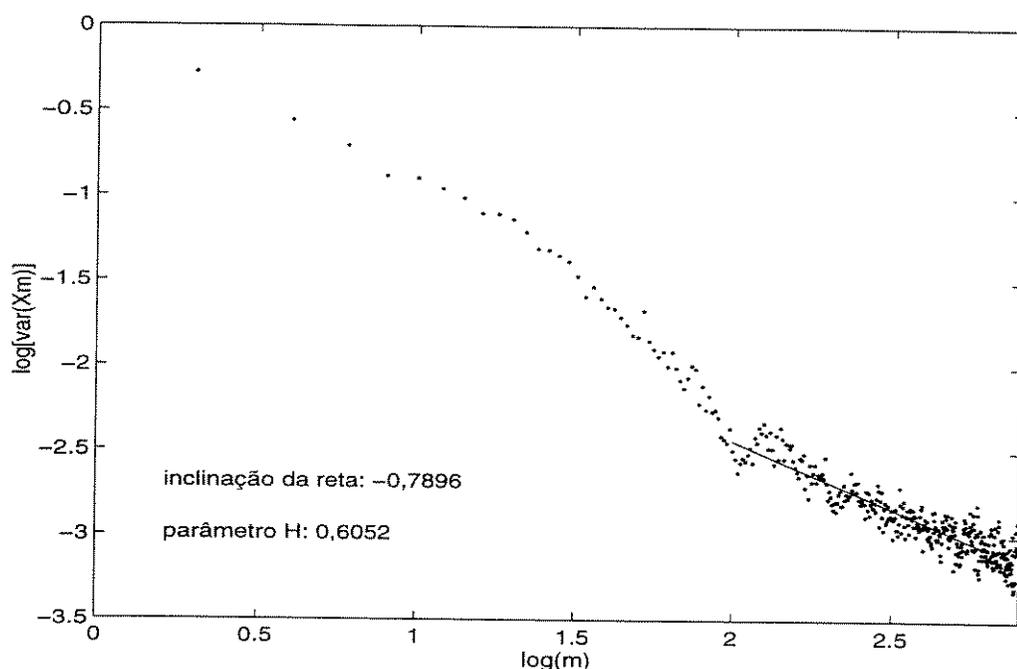


Figura 3.21: Gráfico da variância para blocos de tamanho  $8 \times 8$ .

### 3.3.3. O parâmetro de Hurst na agregação semi-seqüencial

Esta subseção apresenta os resultados de simulação para o caso de agregação *semi-seqüencial*, conforme já mostrado na Figura 3.5. São utilizadas as mesmas 104 imagens da seção anterior e blocos de tamanho  $4 \times 4$  pixels.

O gráfico da variância da série agregada, com 26.624 pontos, é mostrado na Figura 3.22. Pode-se notar uma diferença fundamental em relação à agregação semi-paralela. Trata-se do comportamento do gráfico na região anterior ao ponto crítico que, neste caso, corresponde a  $m \approx 64$  (número de amostras em cada linha das imagens). O parâmetro de Hurst nessa região é próximo de 1,0, representando um comportamento oposto aos casos da subseção anterior, onde  $H = 0,5$ . Isto é perfeitamente justificável, pois nessa região a agregação para o cálculo da variância de  $X^{(m)}$  é efetivamente realizada sobre amostras pertencentes à mesma linha de uma imagem, excetuando-se, obviamente, as regiões de fronteira entre as correspondentes amostras de linhas sucessivas. A simples inspeção visual das imagens ilustradas na Figura 3.3 confirma que a atividade ao longo de uma linha, isto é, dinâmica intraquadros, apresenta uma variação muito lenta (amostras muito correlacionadas). Daí resultam valores de  $H \approx 1,0$ .

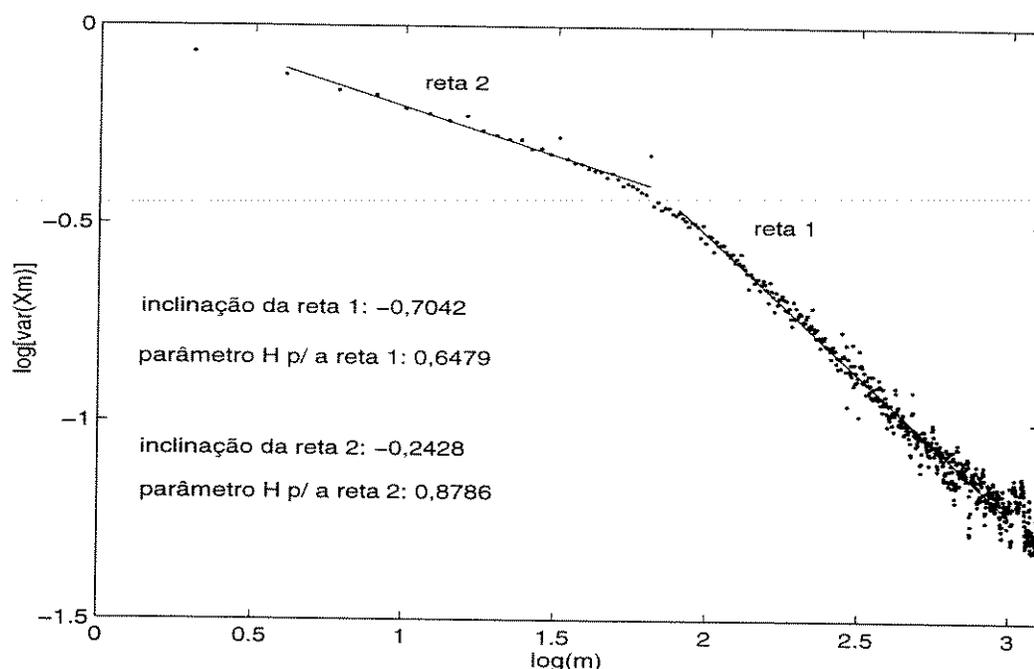


Figura 3.22: Gráfico da variância para agregação semi-seqüencial e blocos de tamanho  $4 \times 4$ .

Na primeira região do gráfico, a reta interpolada para fatores de agregação entre 4 e 64 tem inclinação de  $-\beta = -0,2428$ , correspondendo a um parâmetro  $H \approx 0,8786$ . Na segunda região, considerando fatores entre 80 e 1000, a reta tem inclinação de  $-0,7042$ , levando a  $H \approx 0,6479$ .

### 3.4. Análise da Auto-Similaridade em Seqüências de Vídeo

Nesta seção analisa-se a presença de auto-similaridade no tráfego agregado gerado por seqüências de quadros de cenas de vídeo ao longo do tempo. São consideradas imagens monocromáticas não-codificadas e imagens codificadas no padrão MPEG1 [60].

Utiliza-se, inicialmente, 9 cenas obtidas do Image Processing Lab. do Rensselaer Polytechnic Institute (ipl.rpi), todas em níveis de cinza, com graus de movimento variando entre baixo e médio e com as seguintes quantidades de quadros:

- 1 - *garden* - 99 quadros;
- 2 - *miss.a* - 150;
- 3 - *salesman* - 449;
- 4 - *trevor* - 99;
- 5 - *susie* - 150;
- 6 - *football* - 60;
- 7 - *claire* - 168;
- 8 - *mobile* - 40; e

9 - tennis - 40.

A análise da auto-similaridade realizada sobre o sinal semi-seqüencial da subseção anterior, já representa, de certa forma, a variação espacial (intraquadro) da informação em imagens típicas. É de se esperar que os resultados daquela análise intraquadro possam também ser aplicados à variação espacial da informação em seqüências de vídeo, principalmente se os valores de agregação não forem muito grandes. É bom lembrar que aqueles resultados indicam um valor elevado para o parâmetro de auto-similaridade, pelo menos para aquelas imagens típicas. Portanto, o objetivo aqui é apresentar um novo tipo de análise, para se investigar a auto-similaridade *interquadros*. Neste caso, os resultados indicam o grau de variação local da informação ao longo de quadros sucessivos em certas seqüências de vídeo.

Dessa forma, para cada simulação realizada foi escolhida uma região de 8x8 pixels, aleatoriamente fixada, que serviu de base para a análise estatística ao longo da sucessão de quadros em uma ou mais cenas justapostas. A partir daí calculou-se a variância desses blocos em todos os quadros.

Para as cenas de 1 a 9, listadas acima, obteve-se um conjunto de 1.255 amostras representativas das atividades dos blocos fixados. As Figuras 3.23 e 3.24 mostram, respectivamente, o gráfico da variância e o diagrama "pox" (ou "pox plot") para a série temporal correspondente.

Na Figura 3.23 observa-se que a reta interpolada para fatores de agregação entre 8 e 100 tem inclinação de  $-0,1952$ , correspondendo ao parâmetro de Hurst  $H = 0,9024$ . Para fatores de agregação maiores que 100 aproximadamente, as séries  $X^{(m)}$  têm poucos termos (13 ou menos), tornando o gráfico da variância pouco confiável nesta região.

No "pox plot" observa-se que, após a região transiente inicial, os pontos tendem a flutuar em torno de uma reta com inclinação  $H \approx 0,9722$ .

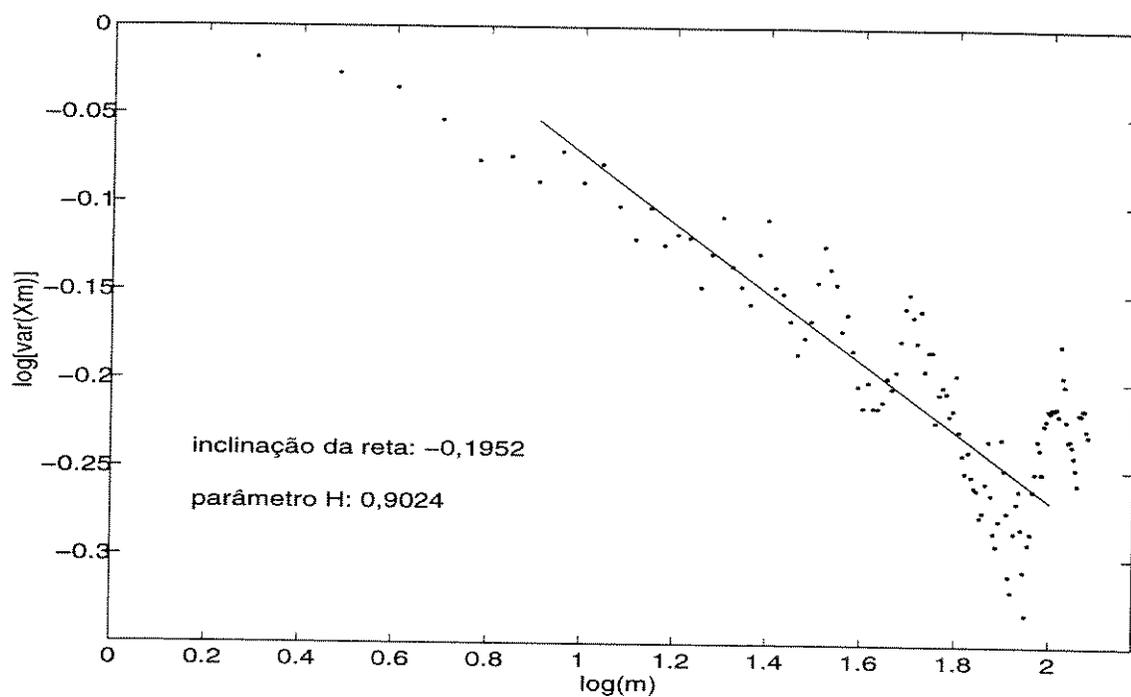


Figura 3.23: Gráfico da variância para as 9 cenas do RPI.

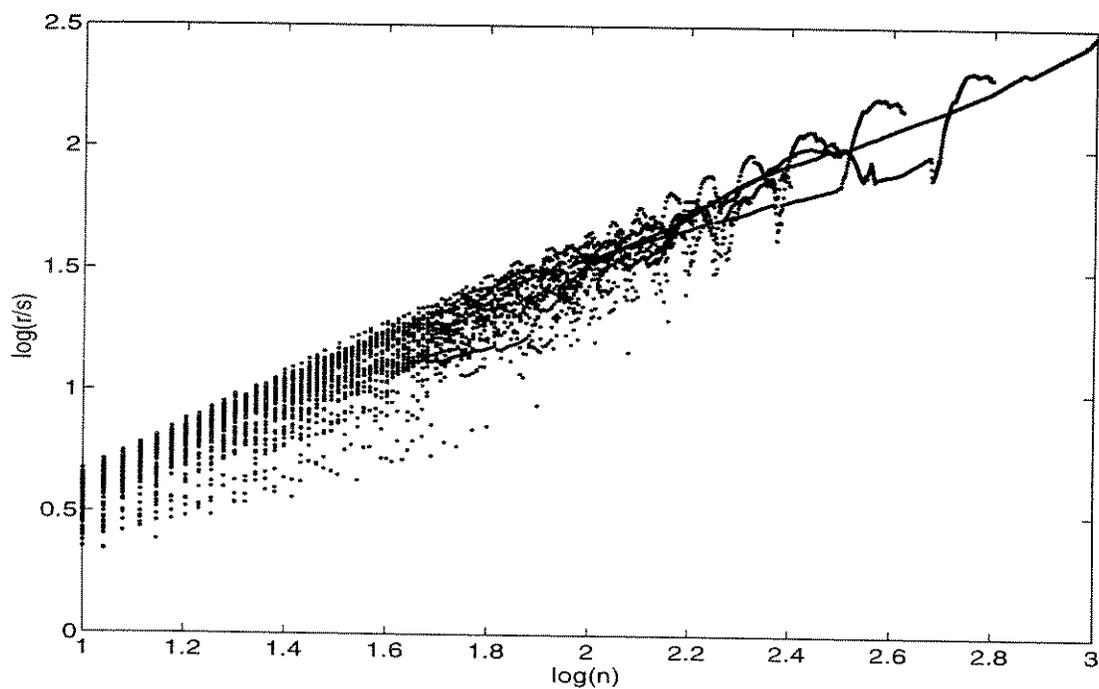


Figura 3.24: Gráfico “pox” para as 9 cenas do RPI.

Pelos valores de  $H$  obtidos, conclui-se que o conjunto de amostras tem grau de auto-

similaridade bastante elevado. Isto pode ser explicado pelo fato das cenas terem baixo ou médio grau de movimento, implicando em quadros bastante correlacionados. A Figura 3.25 mostra a função de autocorrelação dessa série, onde se pode constatar o alto grau de correlação das amostras, mesmo para grandes diferenças de tempo.

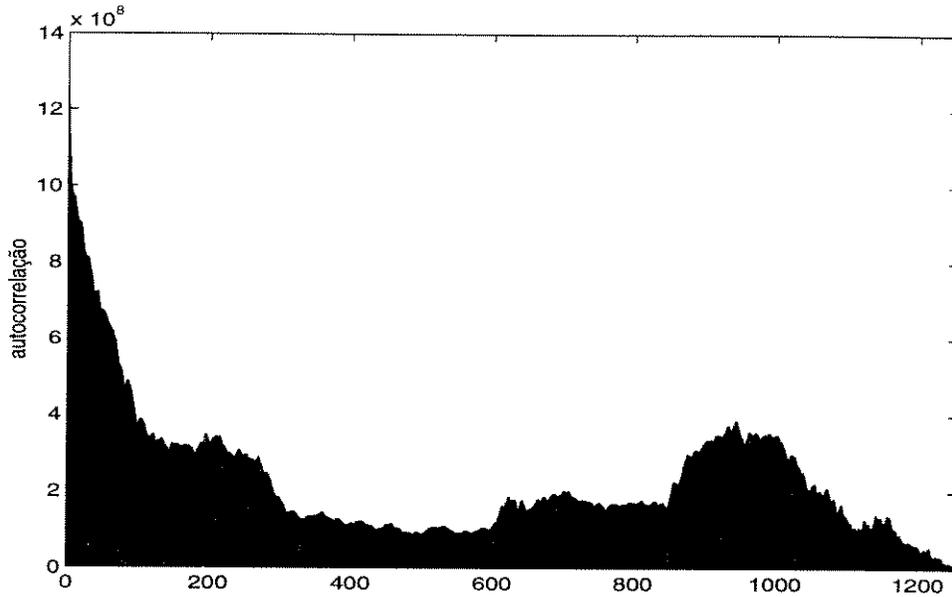


Figura 3.25: Autocorrelação para as 9 cenas do RPI.

A fim de se estudar o comportamento de uma seqüência mais longa, foram anexadas às cenas anteriores novas cenas de vídeo cedidas pelo CPqD-Telebrás. A Figura 3.26 mostra a série temporal das atividades dessas cenas compostas.

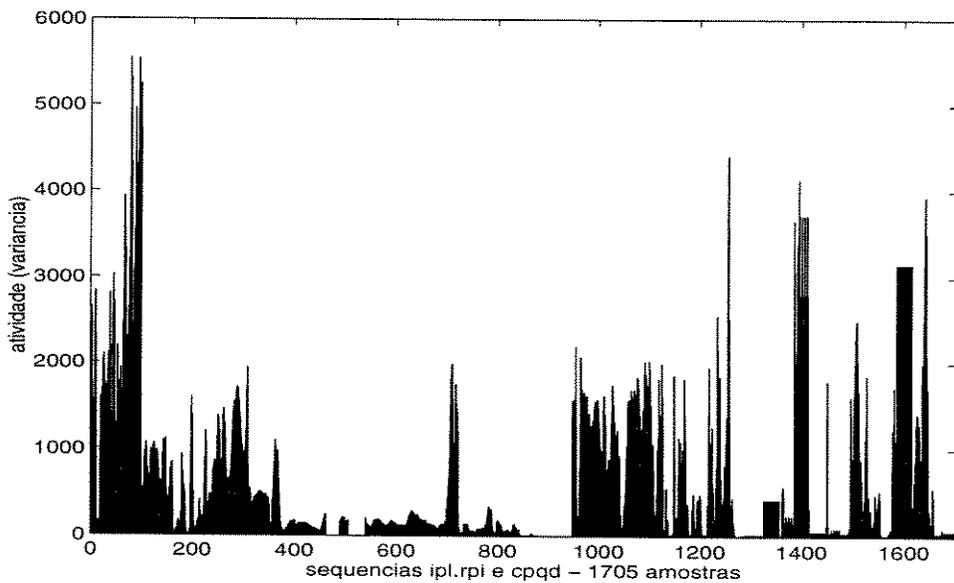


Figura 3.26: Série correspondente às amostras das cenas do RPI e CPqD justapostas.

O gráfico da variância e o “pox plot” obtidos são apresentados nas Figuras 3.27 e 3.28, respectivamente. É interessante observar que o valor de  $H$  para essas cenas agrupadas é menor que o obtido apenas para as cenas do RPI. Isto decorre do fato de que, na realidade, as cenas do CPqD são curtas e encadeadas, portanto com maior atividade média interquadro. A autocorrelação das séries agrupadas (RPI e CPqD), com um total de 1.705 pontos de correlação, é mostrada na Figura 3.29, que também indica dependência de longo prazo.

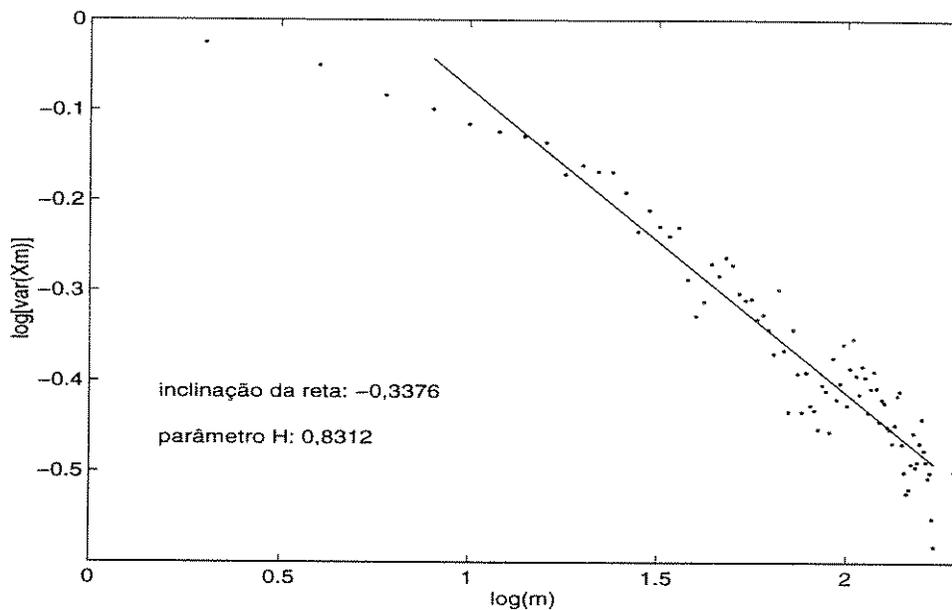


Figura 3.27: Gráfico da variância para as cenas compostas do RPI e CPqD.

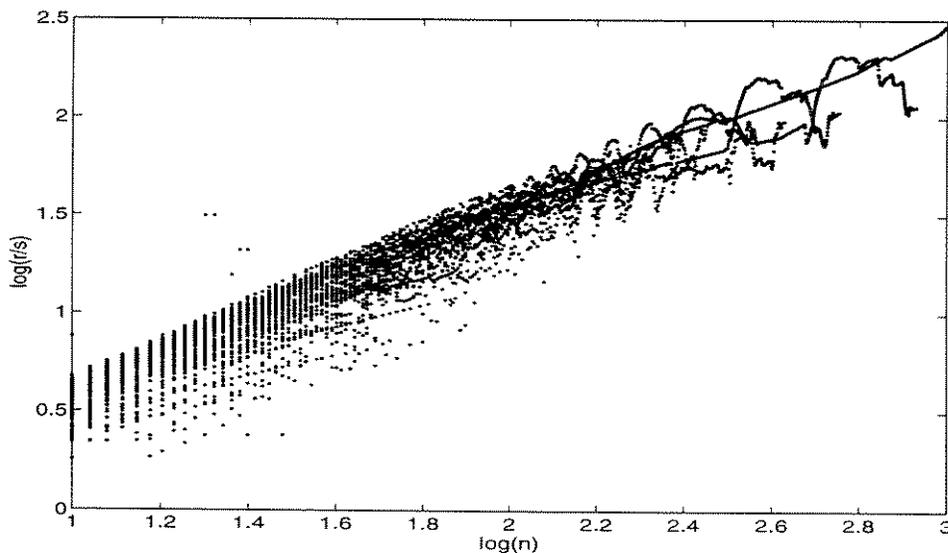


Figura 3.28: Gráfico “pox-plot” para as cenas compostas do RPI e CPqD.

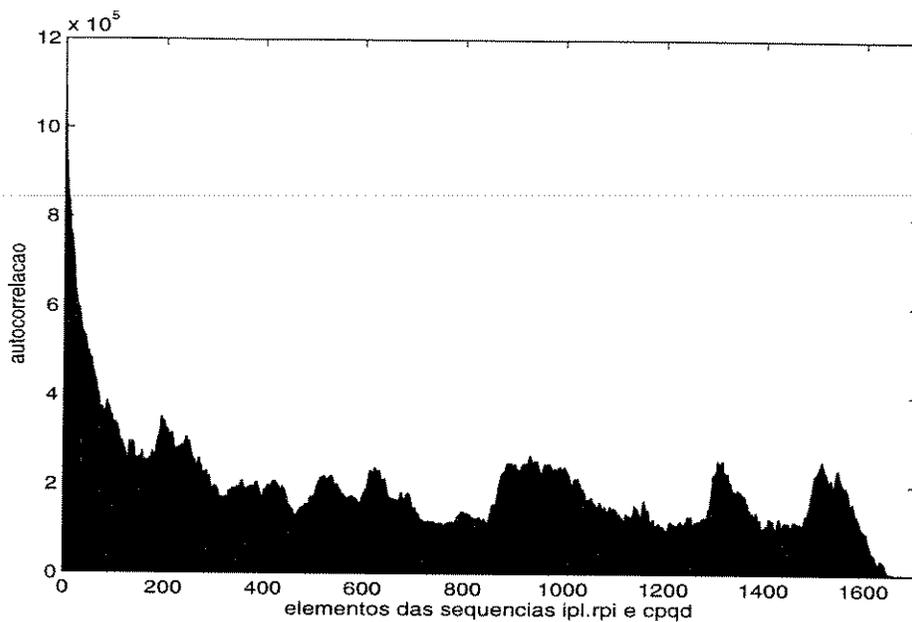


Figura 3.29: Autocorrelação das cenas do RPI e CPqD justapostas.

Além deste estudo com seqüências de imagens não codificadas, foi realizada também uma análise de seqüências de imagens MPEG1 obtidas na Internet. Na realidade, apenas as seqüências de “bits por quadro” é que foram obtidas e utilizadas, o que é suficiente para a análise de auto-similaridade. Neste caso, as séries temporais utilizadas são as próprias séries de números de bits por quadros MPEG1. Foram utilizadas duas seqüências relativas às cenas “lecture” e “advertisements”, obtidas via Internet, no endereço <ftp://tenet.berkeley.edu/pub/dbind/traces>, arquivos do Grupo Tenet da Universidade da Califórnia, em Berkeley, e do Instituto Internacional de Ciência da Computação.

Estas cenas têm 10 minutos de duração, com quadros de dimensões 160x120 pixels. Foram codificadas em MPEG1 à taxa de 30 fps, através de software desenvolvido pela Universidade da Califórnia, em Berkeley. Os frames I, P e B são ordenados na forma IBBPBB.

A cena “advertisements” consiste de vários comerciais mostrando características de produtos, tais como formato, etc. A cena “lecture” corresponde a uma preleção, mostrando o expositor de corpo inteiro, se movimentando, e slides, incluindo “zoom”.

Estas seqüências estão apresentadas nas Figuras 3.30 e 3.31 e os respectivos gráficos da variância são mostrados nas Figuras 3.32 e 3.33, respectivamente.

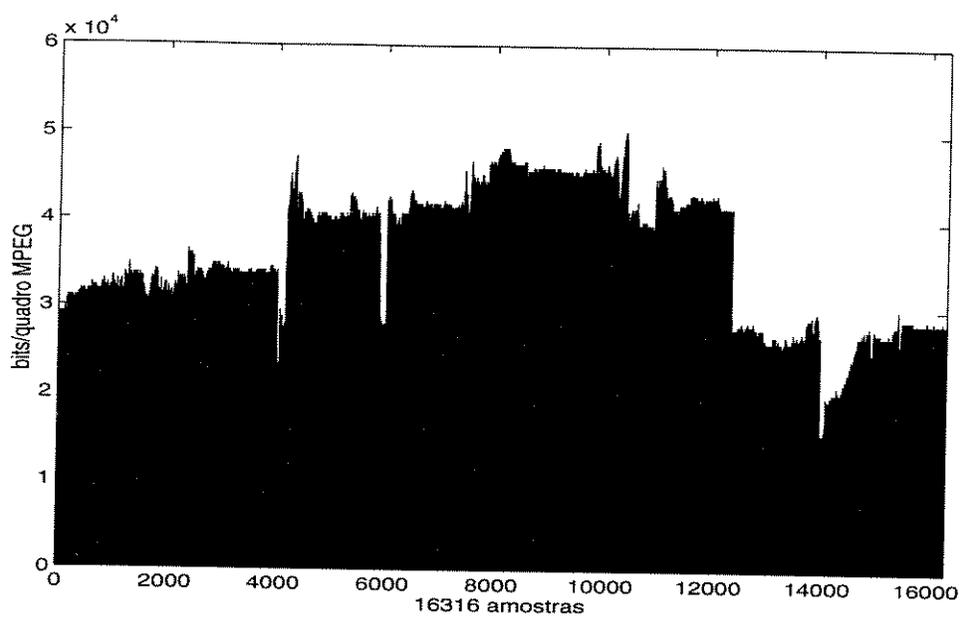


Figura 3.30: Série correspondente à cena “lecture”.

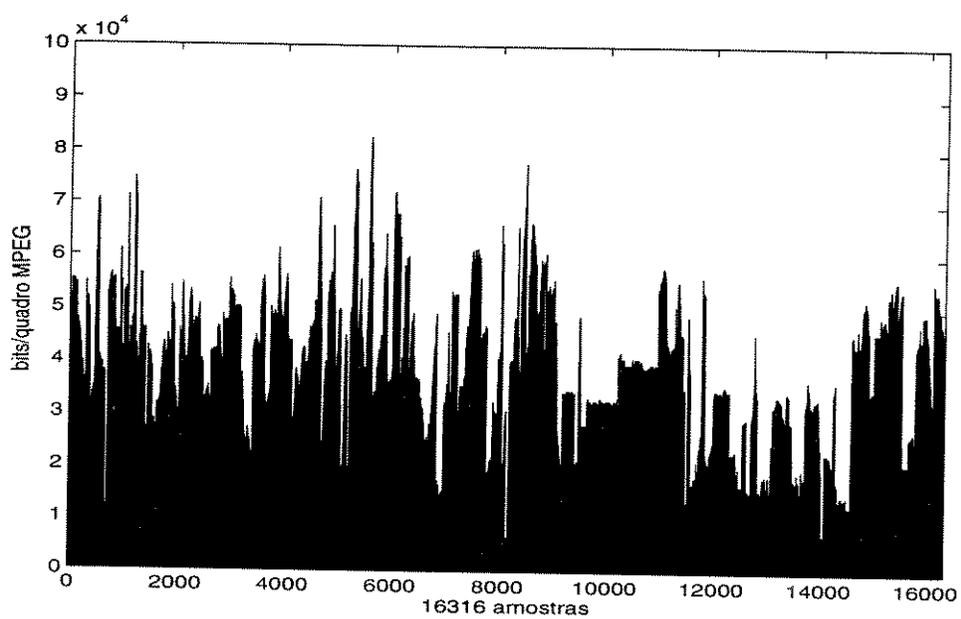


Figura 3.31: Série correspondente às cenas “advertisements”.

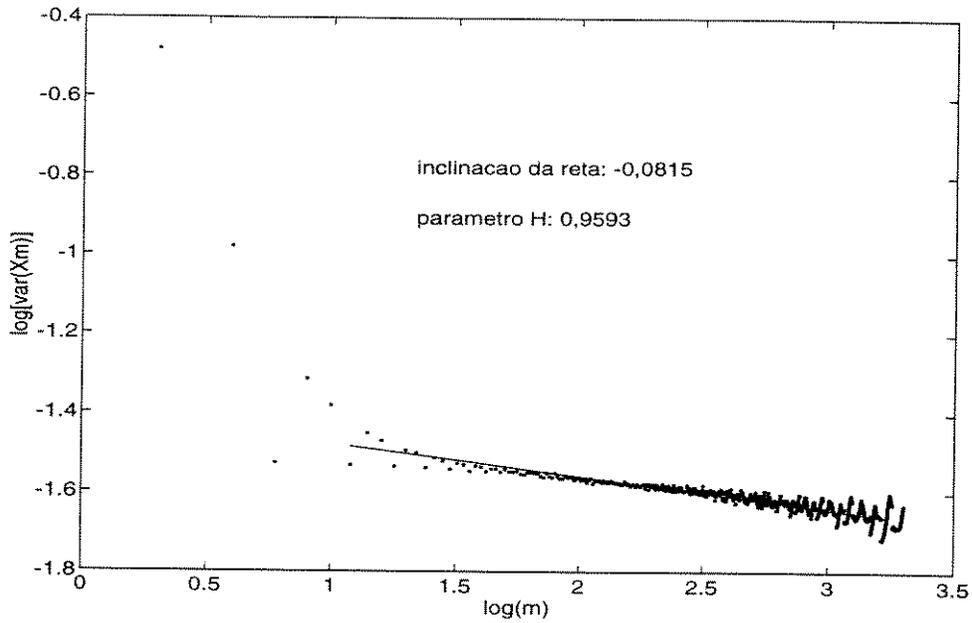


Figura 3.32: Gráfico da variância para a cena MPEG “lecture”.

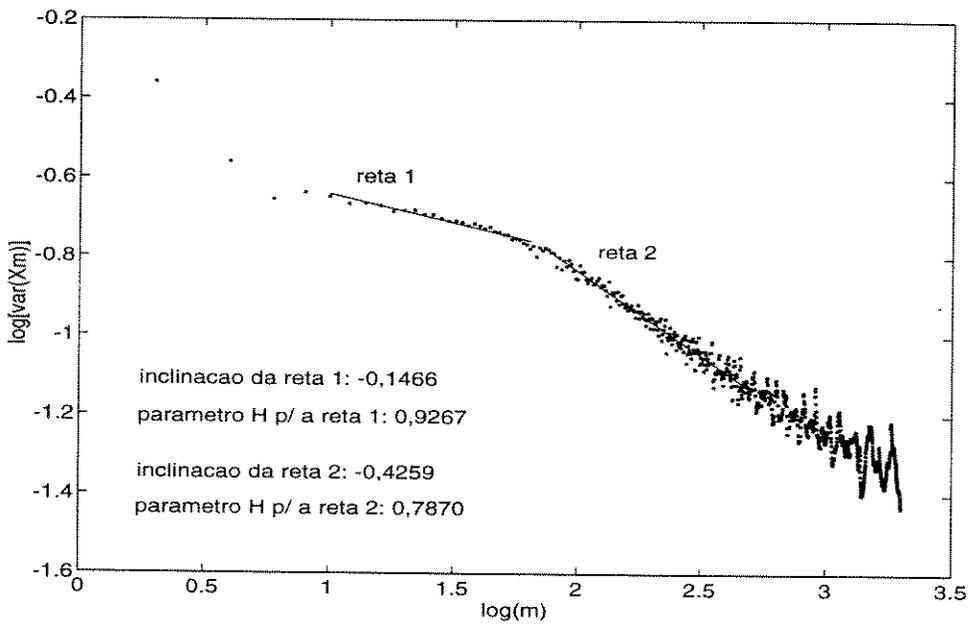


Figura 3.33: Gráfico da variância para as cenas MPEG “advertisements”.

Para a seqüência “lecture” verifica-se que a reta tem inclinação  $-\beta \approx -0,0815$ , correspondendo ao parâmetro  $H \approx 0,9593$  e demonstrando elevada auto-similaridade.

O gráfico da variância para “advertisements”, Figura 3.33, apresenta regiões com inclinações distintas e, portanto, deve ser analisado de maneira diferente da adotada na Figura 3.32. Desprezando-se os primeiros pontos do gráfico, que correspondem a pequenos fatores de agregação ( $m$ ), observa-se uma inclinação bem definida até  $m \approx 63$  ( $\log(m) \approx 1,8$ ). A reta interpolada nessa região apresenta inclinação  $-\beta \approx -0,1466$ , correspondendo a  $H \approx 0,9267$ , indicando auto-similaridade acentuada (amostras bastante correlacionadas). A partir desse ponto ( $m \approx 63$ ) a inclinação aumenta consideravelmente, e, pela interpolação da reta nessa região do gráfico, obtém-se  $-\beta \approx -0,4259$  e  $H \approx 0,7870$  correspondendo a uma diminuição da auto-similaridade. Para se explicar integralmente esse comportamento do gráfico seriam necessárias maiores informações sobre a cena. Pode-se apenas verificar das Figuras 3.30 e 3.31, que a série das cenas “advertisements” caracteriza um processo estocástico bem mais *rápido* que a “lecture”, isto é, a correlação das amostras da primeira seqüência é bem menor que a da segunda, mostrando a coerência dos valores de  $H$  calculados.

# Capítulo 4

## Geração de Processos Auto-Similares

### 4.1. Introdução

Para se estudar o funcionamento de redes de faixa larga com tráfego auto-similar, muitas vezes torna-se necessária a realização de simulações utilizando-se geradores de processos auto-similares. Assim, visando uma contribuição para futuros estudos de tráfego, desenvolveu-se neste trabalho um gerador de séries temporais auto-similares baseado no modelo ARIMA fracionário.

Antes de apresentar o gerador, mostra-se uma forma de se obter processos auto-similares através do movimento Browniano fracionário e, também, o desenvolvimento do modelo ARIMA fracionário.

### 4.2. Obtenção de Processos fBm a partir da Agregação de Processos Inovações

Um processo *inovação* é obtido por uma transformação linear causal, tal como pela filtragem linear de um processo aleatório discreto estacionário, sem componentes periódicas. A transformação inversa gera um sinal com as mesmas propriedades estatísticas do processo original. As amostras de um processo inovação são estatisticamente independentes, cada uma representando informação absolutamente nova, justificando sua denominação. Portanto, esse processo corresponde a um ruído branco.

Um processo do tipo movimento Browniano fracionário (**fBm**) pode ser obtido a partir da agregação de muitos processos inovações, desde que a duração dos intervalos entre períodos de atividade (ou inovações) seja modelado por um processo com variância infinita [21]. Esses intervalos podem ser representados por uma seqüência de variáveis aleatórias independentes e identicamente distribuídas (*i.i.d*), composta por números inteiros ( $U_0, U_1, U_2, \dots$ ) e com função de distribuição de probabilidade caracterizada por

$$P[U \geq u] \sim u^{-\alpha} g(u), \quad u \rightarrow \infty. \quad (4.1)$$

sendo  $g$  uma função que varia lentamente para  $u \rightarrow \infty$ . Esta característica, comumente chamada de “heavy tail”, é satisfeita, por exemplo, pela distribuição de Pareto com

parâmetro  $\alpha$  tal que  $1 < \alpha < 2$ . Isto significa que  $\{U_k\}$  pode assumir valores elevados com alta probabilidade. Considera-se então uma seqüência  $\{S_k\}$  da forma

$$S_k = S_0 + \sum_{j=1}^k U_j, \quad (4.2)$$

onde  $S_k \geq 0$  e  $k \geq 1$ . Pode-se dizer que esta seqüência é um processo inovação e o elemento  $S_k$  representa o instante da  $k$ -ésima inovação (ou período de atividade).  $S_0$  e  $S_1$  devem ser escolhidos de forma a assegurar a estacionariedade de  $\{S_k\}$ . Nota-se que a distribuição de  $(S_k - S_{k-1})$  é igual a do processo  $U$ , para  $K \neq 0$ .

Considera-se também uma seqüência de variáveis aleatórias *i.i.d*  $\{W_k\}$ , com média zero e variância infinita, independente das variáveis  $U_k$ . Define-se o *processo inovação "rewards"* como

$$W = (W(t) : t = 0, 1, 2, \dots),$$

onde

$$W(t) = \sum_{k=0}^t W_k I_{(S_{k-1}, S_k]}(t) \quad (4.3)$$

sendo  $I_{(S_{k-1}, S_k]}(t)$  a função indicadora para o intervalo  $(S_{k-1}, S_k]$ , definida da seguinte forma:

$$I_{(S_{k-1}, S_k]}(t) = \begin{cases} 1, & S_{k-1} < t \leq S_k \\ 0, & \text{fora desse intervalo.} \end{cases} \quad (4.4)$$

Isto significa que

$$W(t) = \begin{cases} W_k, & S_{k-1} < t \leq S_k \\ 0, & \text{fora desse intervalo.} \end{cases} \quad (4.5)$$

Então, agregando-se  $M$  cópias de  $W$  independentes e identicamente distribuídas,  $W^{(1)}, W^{(2)}, \dots, W^{(M)}$ , obtém-se o processo de interesse,

$$W^*(T, M) = \sum_{t=1}^T \sum_{m=1}^M W^{(m)}(t), \quad T = 1, 2, \dots, \quad (4.6)$$

com  $W^*(0, M) = 0$ .

Foi demonstrado por Mandelbrot e Taqqu que no caso de  $T$  e  $M$  muito grandes e  $T \ll M$ ,  $W^*(T, M)$  se comporta como um processo **fBm**, com parâmetro  $H$  tal que  $1/2 < H < 1$ . Portanto, o processo de incrementos de  $W^*$  corresponde, neste caso, ao ruído Gaussiano fracionário.

### 4.3. Geração do Modelo ARIMA fracionário

Esta seção apresenta o modelo ARIMA ("Autoregressive Integrated Moving Average") fracionário e uma simulação elaborada no MATLAB 4.2, com a finalidade de gerar

séries temporais auto-similares a partir desse modelo. Antes, porém, é interessante rever as características dos modelos paramétricos AR (autoregressivo) e MA (“Moving Average” ou modelo da Média Deslizante).

### 4.3.1. Modelos Autoregressivo (AR) e da Média Deslizante (MA)

Estes modelos se baseiam no conceito de *inovação*, apresentado na seção anterior, e são amplamente aplicados em processamento digital de sinais. Consistem na filtragem linear de um ruído branco (processo inovação) de modo a fornecer um sinal aleatório com características estatísticas conhecidas. Seja  $v(n)$  a seqüência ruído branco e  $x(n)$  o sinal aleatório obtido na filtragem. A Figura 4.1 representa a filtragem do modelo AR, onde  $a_1, a_2, \dots, a_M$  são os parâmetros do filtro e  $z^{-1}$  representa o atraso das amostras.

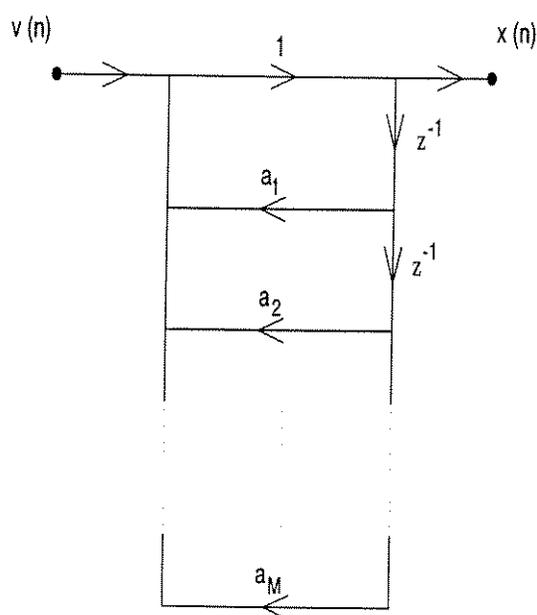


Figura 4.1: Filtragem do modelo AR.

Este filtro é do tipo IIR (“Infinite Impulse Response”), isto é, a seqüência de saída tem comprimento infinito, se a entrada é uma seqüência impulso. Da figura nota-se que a amostra  $x(n)$  é obtida da combinação linear da amostra do ruído branco no mesmo instante  $n$  e de  $M$  amostras passadas de  $x$ . Diz-se, então que  $x(n)$  é um processo AR de ordem  $M$ . A equação a diferenças do modelo AR é

$$x(n) = v(n) + \sum_{k=1}^M a_k x(n - k). \quad (4.7)$$

O modelo MA é o inverso do AR, como mostra a Figura 4.2.

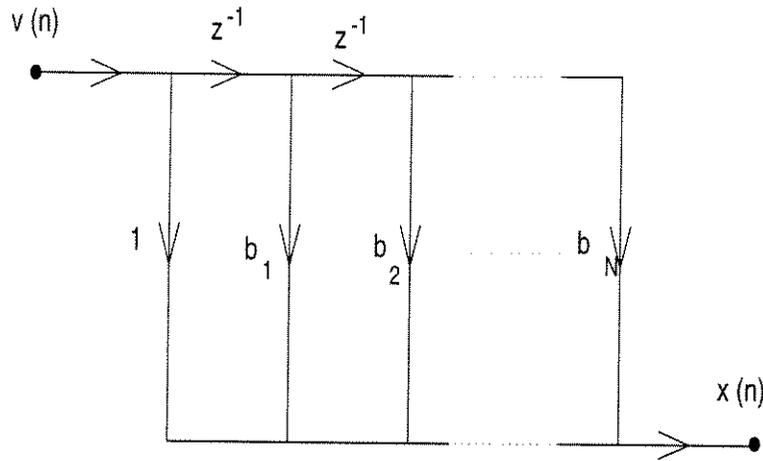


Figura 4.2: Filtragem do modelo MA.

Neste caso o filtro é do tipo FIR (“Finite Impulse Response”); a seqüência de saída tem comprimento finito, pois é obtida da combinação linear das amostras da entrada apenas. Então, com referência à Figura 4.2, o processo  $x(n)$  é dito MA de ordem  $N$ . Sua equação a diferenças é dada por

$$x(n) = v(n) + \sum_{k=1}^M b_k v(n - k). \quad (4.8)$$

A combinação desses dois modelos resulta no modelo ARMA, de caráter mais geral, com a seguinte equação de diferenças

$$x(n) = v(n) + \sum_{k=1}^M b_k v(n - k) + \sum_{k=1}^M a_k x(n - k) \quad (4.9)$$

Os modelos AR, MA e ARMA são perfeitamente adequados para processos com dependência de curto prazo, mas não conseguem captar a de longo prazo. Geram seqüências com funções de autocorrelação que decrescem exponencialmente.

### 4.3.2. O Modelo ARIMA Fracionário

O modelo ARIMA fracionário proposto por Granger e Joyeux [17] e Hosking [18] é uma generalização do ARIMA que, por sua vez, se baseia no modelo ARMA. Tem a vantagem de caracterizar tanto o comportamento de curto prazo como o de longo prazo, sendo utilizado para processos auto-similares. É mais versátil que o modelo do ruído Gaussiano fracionário que retrata apenas o comportamento de longo prazo.

Esta subseção apresenta inicialmente o modelo ARIMA  $(p, d, q)$  proposto por Box e Jenkins [19]. Este modelo tem a seguinte expressão:

$$\Phi(B)x(n) = (1 - B)^{-d} \Theta(B)v(n), \quad (4.10)$$

onde  $\{x(n)\}$  é o processo a ser modelado;  $n$  e  $d$  são números inteiros;  $B$  representa a operação de atraso definida por  $Bx(n) = x(n - 1)$  (equivale a  $z^{-1}$  das Figuras 4.1 e 4.2);

$\Phi(B)$  e  $\Theta(B)$  são polinômios em  $B$  relativos aos modelos AR e MA, respectivamente; e  $v(n)$  é o ruído branco. Os parâmetros  $(p, d, q)$  do modelo são as ordens das partes AR e MA,  $p$  e  $q$ , respectivamente, e o grau de diferenciação,  $d$ . Por exemplo, se  $p$  e  $q$  são nulos e  $d = 1$  (ARIMA(0, 1, 0)), tem-se

$$(1 - B)x(n) = v(n), \quad (4.11)$$

isto é, a primeira diferença de  $\{x(n)\}$  é um processo ruído branco chamado caminho aleatório (“random walk”).

A generalização proposta nas referências [17] e [18] consiste em permitir que o grau de diferenciação ( $d$ ) assumira qualquer valor real e não apenas inteiros. Esta nova classe de processos é chamada ARIMA fracionária. Em particular, para  $d$  pertencente ao intervalo  $(0, 1/2)$ , o processo ARIMA fracionário se torna capaz de modelar a dependência de longo prazo. A função de autocorrelação é positiva e decresce monotonicamente e de forma assintoticamente hiperbólica para amostras afastadas.

O termo  $(1 - B)^{-d}$ , chamado operador diferença fracionária e representado por  $\Delta^{-d}$ , é definido pela série binomial

$$\begin{aligned} \Delta^{-d} &= (1 - B)^{-d} = \sum_{k=0}^{\infty} \binom{d}{k} (-B)^k = \\ &= 1 - dB - \frac{1}{2}d(1 - d)B^2 - \frac{1}{6}d(1 - d)(2 - d)B^3 - \dots \end{aligned} \quad (4.12)$$

Fazendo uma analogia com o processo **fBm**, discutido anteriormente, o ruído Gaussiano fracionário, que é a “derivada” daquele processo, pode ser interpretado como a derivada fracionária de ordem  $(1/2 - H)$  do ruído branco de tempo contínuo. Para  $H = 1/2$ , o ruído Gaussiano fracionário se reduz ao ruído branco. O grau de diferenciação  $d$  é escolhido como  $d = H - 1/2$ , de forma que o processo de tempo discreto  $\{x(n)\}$ , dado por

$$x(n) = \Delta^{-d} v(t), \quad (4.13)$$

com  $v(t)$  representando o ruído branco de tempo contínuo, seja o análogo do ruído Gaussiano fracionário de tempo contínuo. Assim  $\{x(n)\}$  é o processo ARIMA  $(0, d, 0)$ .

Se  $d$  é nulo (ARIMA  $(0, 0, 0)$ ),  $\{x(n)\}$  é o próprio ruído branco. Se  $d$  é tal que  $-1/2 < d < 0$ , o processo ARIMA  $(0, d, 0)$  não tem dependência de longo prazo, ou seja, apresenta “memória curta”. A autocorrelação é negativa, exceto para a origem, e na densidade espectral de potência predominam as componentes de altas frequências, tornando-a uma função crescente e que é nula na origem. Portanto, o interesse nesta seção está voltado para processos ARIMA fracionários com  $d = H - 1/2$  e  $0 < d < 1/2$ .

### 4.3.3. O Modelo ARIMA Fracionário de Variância Infinita

Recentemente, Taqu e Koskoszka [32] propuseram uma alteração no modelo ARIMA fracionário, de forma a incluir a característica de variância infinita, ou “heavy tail”,

apresentada na seção anterior. Assim, além da dependência de longo prazo, o modelo passa a ter também alta variação.

O novo conjunto de equações é dado por

$$\Phi(B)x(n) = \Theta(B)\Delta^{-d}\epsilon(n). \quad (4.14)$$

onde  $\{\epsilon(n)\}$  é o processo inovação (ruído branco) com variância infinita e  $d$  satisfaz as condições estabelecidas acima. Assume-se que  $\epsilon(n)$  são variáveis aleatórias i.i.d  $\alpha$ -estáveis simétricas, simbolizadas por  $S\alpha S$ , com  $\alpha$  tal que  $1 < \alpha < 2$ . Diz-se que as variáveis  $\epsilon(n)$  estão no domínio de atração de uma lei  $\alpha$ -estável, isto é,

$$P(|\epsilon(n)| > x) = x^{-\alpha}L(x), \quad \text{com } x \rightarrow \infty, \quad (4.15)$$

onde  $L$  é uma função que varia lentamente e

$$\frac{P(\epsilon(n) > x)}{P(|\epsilon(n)| > x)} \rightarrow a, \quad (4.16)$$

$$\frac{P(\epsilon(n) > -x)}{P(|\epsilon(n)| > x)} \rightarrow b, \quad (4.17)$$

sendo  $a$  e  $b$  inteiros não negativos, que satisfazem a  $a + b = 1$ . Esta classe de variáveis aleatórias tem função característica da forma  $\varphi(\omega) = \exp(-\sigma^\alpha|\omega|^\alpha)$ , sendo  $\sigma$  o parâmetro de escala e  $\alpha$ , o *expoente característico*, que determina a intensidade do “heavy tail”, isto é, o quanto a calda da distribuição da variável aleatória decresce mais lentamente que a da Gaussiana. Nesta seção considera-se também que as distribuições das variáveis  $\epsilon(n)$   $\alpha$ -estáveis são simétricas.

Foi demonstrado que para  $\epsilon(n)$  com essas características, existe um único modelo MA da forma

$$x(n) = \sum_{j=0}^{\infty} c_j \epsilon(n-j) \quad (4.18)$$

que satisfaz às equações 4.14, desde que os polinômios  $\Phi(B)$  e  $\Theta(B)$  não tenham raízes no interior da circunferência de raio unitário ( $\mathcal{Z} : |\mathcal{Z}| \leq 1$ ), nenhuma raiz em comum e que a desigualdade  $\alpha(d-1) < -1$  seja atendida. Esta desigualdade é necessária para assegurar a convergência da série na equação 4.18, o que equivale a se ter

$$\sum_{j=0}^{\infty} |c_j|^\alpha < \infty, \quad 1 < \alpha < 2. \quad (4.19)$$

Os coeficientes  $c_j$  são definidos por

$$\sum_{j=0}^{\infty} |c_j| \mathcal{Z}^j = \frac{\Theta(\mathcal{Z})}{\Phi(\mathcal{Z})(1-\mathcal{Z})^d}, \quad |\mathcal{Z}| < 1 \quad (4.20)$$

e são assintoticamente proporcionais a  $j^{d-1}$  quando  $j \rightarrow \infty$ . Esta é a razão da condição  $\alpha(d-1) < -1$  para a convergência.

No caso do processo ARIMA  $(0, d, 0)$ , tem-se

$$\begin{aligned} x(n) &= \Delta^{-d} \epsilon(n) \quad \text{ou} \\ x(n) &= (1 - B)^{-d} \epsilon(n) = \sum_{j=0}^{\infty} b_j \epsilon(n - j) \end{aligned} \quad (4.21)$$

sendo  $b_j$  os coeficientes da expansão da série  $(1 - Z)^{-d}$ ,  $|Z| < 1$ , dados por

$$\begin{aligned} b_0 &= 1, \\ b_j &= \frac{\Gamma(1 + d)}{\Gamma(d)\Gamma(j + 1)}, \quad j \geq 1 \end{aligned} \quad (4.22)$$

A referência [31] apresenta uma maneira de estimar os parâmetros do modelo ARIMA fracionário com variância infinita, baseada em periodograma, e, também, uma simulação para gerar processos ARIMA  $(0, d, 0)$ . Nesta simulação a ordem do modelo MA equivalente, equação 4.20, é limitada da seguinte forma

$$x(n) = \sum_{j=0}^J c_j \epsilon(n - j), \quad n = 1, 2, \dots, N, \quad (4.23)$$

sendo  $N$  o comprimento da seqüência  $x(n)$ . Os coeficientes  $c_j$  coincidem com os  $b_j$  da equação 4.21, e podem ser obtidos facilmente pela aplicação da propriedade de recorrência da função Gama, resultando em

$$\begin{aligned} c_0 &= 1 \\ c_{j+1} &= \frac{j + d}{j + 1} c_j. \end{aligned} \quad (4.24)$$

O ruído branco estável e simétrico  $\epsilon(n)$ , por sua vez, é simulado utilizando uma versão do algoritmo de Chambers, Mallow e Stuck [35]. Em síntese, este algoritmo gera uma variável  $\alpha$ -estável por meio de transformações não lineares de duas variáveis aleatórias, uma com distribuição exponencial e outra com distribuição uniforme. Seja  $W$  a variável exponencial, com distribuição de probabilidade

$$P\{W \geq \omega\} = \exp(-\omega), \quad \omega \geq 0, \quad (4.25)$$

e  $Y$  a variável com distribuição uniforme no intervalo  $[-\pi/2, \pi/2]$ . A variável  $\alpha$ -estável,  $S(\alpha)$ , é obtida da seguinte transformação

$$S(\alpha) = \left( \frac{\sin(\alpha Y)}{\cos(Y)} \right) \left( \frac{\cos(eY)}{W \cos(Y)} \right)^{e/(1-e)}, \quad (4.26)$$

sendo  $e = 1 - \alpha$ .

Taqqu e Kokoszka [31] simularam apenas os casos

$$\begin{aligned}
 d = 0,1 & \quad , \quad \alpha = 1,2; \\
 d = 0,1 & \quad , \quad \alpha = 2; \\
 d = 0,2 & \quad , \quad \alpha = 1,5; \quad e \\
 d = 0,2 & \quad , \quad \alpha = 2.
 \end{aligned}$$

Os resultados foram excelentes para  $N = 10000$  e  $J = 1000$ . Para séries curtas ( $N$  pequeno) o resultado é inferior.

#### 4.4. Gerador de Séries Temporais Auto-Similares

O objetivo desta seção é apresentar um gerador de séries temporais com parâmetros de Hurst pré-definidos no intervalo  $(1/2, 1)$ . Este gerador se baseia na simulação descrita na seção anterior, fornecendo, portanto, um processo ARIMA  $(0, d, 0)$  fracionário de variância infinita. Foi implementado no MATLAB 4.2, conforme programa no Apêndice B.

As condições a serem atendidas são

$$\begin{aligned}
 d = H - 1/2, \quad 0 < d < 1/2; \\
 \alpha (d - 1) < -1 \quad \text{ou} \quad \alpha > \frac{1}{\frac{3}{2} - H}; \quad e \\
 1 < \alpha < 2.
 \end{aligned}$$

A Figura 4.3 apresenta o gráfico limite de  $\alpha$  versus  $H$ , onde a região acima da curva corresponde aos possíveis valores de  $\alpha$  para um dado  $H$ .

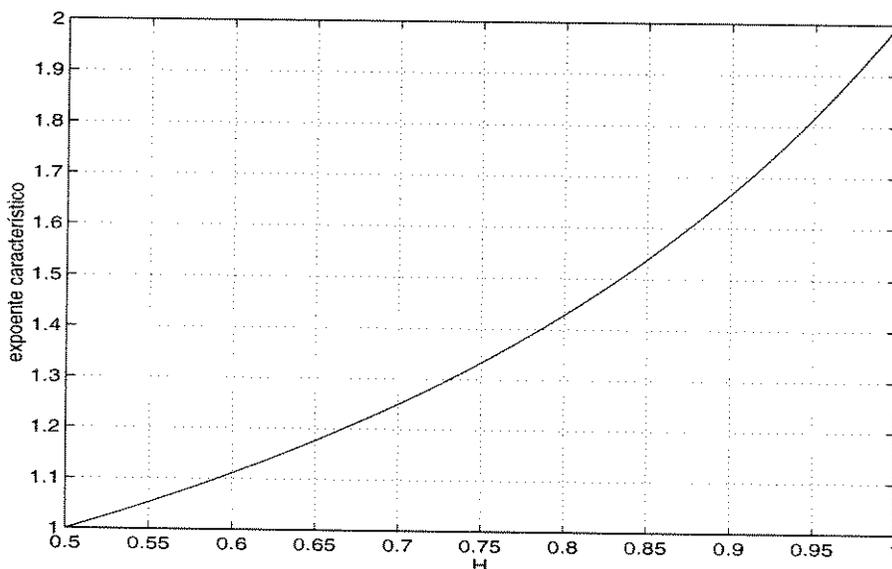


Figura 4.3: Gráfico limite do expoente característico ( $\alpha$ ) versus o parâmetro de Hurst ( $H$ ).

Foram realizadas simulações para  $H = 0,6, 0,7, 0,8$  e  $0,9$ , com diversos valores de  $\alpha$ , algumas variações no comprimento da série e também na ordem do modelo MA

equivalente ( $J$ ). Para se verificar a precisão do gerador, cada simulação foi repetida 50 vezes para se calcular a média, o desvio padrão e a raiz quadrada do erro quadrático médio dos parâmetros  $H$  estimados ( $\hat{H}$ ). Foi utilizado o método da variância para o cálculo de  $\hat{H}$ . A Tabela 4.1 apresenta alguns desses resultados.

No. de elementos	Ordem do Filtro	$H$	$\alpha$	$\hat{H}$ médio	$\hat{\sigma}_{\hat{H}}$	$\sqrt{mse}$
10000	1000	0,9	1,9	0,8415	0,0911	0,1075
10000	1000	0,9	1,8	0,8376	0,0527	0,0813
10000	1000	0,9	1,7	0,8526	0,0389	0,0611
10000	1000	0,9	1,667	0,8544	0,0243	0,0515
10000	1000	0,9	1,6667	0,8484	0,0306	0,0598
10000	1000	0,8	1,9	0,8131	0,0444	0,0458
10000	1000	0,8	1,8	0,8201	0,0299	0,0358
10000	1000	0,8	1,7	0,8097	0,0486	0,0490
10000	1000	0,8	1,6	0,8233	0,0283	0,0365
20000	1500	0,8	1,429	0,7984	0,0909	0,0900
10000	1000	0,7	1,9	0,7637	0,0520	0,0819
10000	1000	0,7	1,8	0,7221	0,1190	0,1198
10000	1000	0,7	1,7	0,7151	0,1225	0,1222
10000	1000	0,7	1,4	0,7220	0,1188	0,1196
20000	1500	0,7	1,252	0,7604	0,0322	0,0684
20000	1500	0,7	1,2501	0,7491	0,0666	0,0822
10000	1000	0,6	1,8	0,7052	0,0856	0,1351
10000	1000	0,6	1,7	0,7256	0,0344	0,1302
10000	1000	0,6	1,5	0,7086	0,0431	0,1167
20000	1500	0,6	1,5	0,7019	0,0948	0,1386
20000	1500	0,6	1,2	0,6966	0,0803	0,1251
20000	1500	0,6	1,12	0,7053	0,0912	0,1387
20000	1500	0,6	1,1112	0,7006	0,0647	0,1193

Tabela 4.1: Alguns resultados obtidos na simulação;  $\hat{\sigma}_{\hat{H}}$  é o desvio padrão de  $\hat{H}$  e  $\sqrt{mse}$  é a raiz quadrada do erro quadrático médio.

Verificou-se que  $\hat{H}$  médio se aproxima bastante do especificado para  $H \geq 0,7$  (série com 10000 elementos), embora não atinja a precisão da simulação de [31]. Isto pode ser devido a diferenças entre os geradores das distribuições uniforme e exponencial utilizados. Observou-se ainda que parâmetros  $\alpha$  intermediários, dentro da faixa permitida para cada valor de  $H$ , fornecem melhores resultados. Para  $H = 0,6$ ,  $\hat{H}$  médio se afasta mais do especificado. Pode-se considerar, então, que o desempenho do gerador é razoável para  $H \geq 0,7$ .

Deve ser ressaltado que a série temporal gerada tem componentes negativos, pois o ruído branco estável,  $\epsilon(n)$ , tem média zero. Logo, para a simulação do tráfego de dados é necessário acrescentar a cada componente um valor maior ou igual ao mínimo encontrado na série. Assim, a série passa a ter média maior ou igual a esse valor mínimo. É bom enfatizar que este detalhe quase não é lembrado nos trabalhos da área.

## Capítulo 5

### Conclusões

O conhecimento preciso do comportamento estatístico do tráfego é essencial para o projeto e gerenciamento adequados de redes de faixa larga. A recente descoberta da natureza auto-similar do tráfego da rede Ethernet, um dos principais clientes das redes de faixa larga, desencadeou muitos trabalhos sobre o assunto, o que demonstra a preocupação dos pesquisadores da área em relação à definição de modelos e aos efeitos da estatística auto-similar no congestionamento das redes de faixa larga. Foi constatado que o modelo de Poisson, comumente utilizado para esse tráfego, não é adequado pois não consegue captar a dependência de longo prazo entre as amostras. Isto significa que o problema é mais complicado do que se pensava, como pode ser percebido pela probabilidade de transbordamento dos “buffers” e pelo atraso das células. Até o momento, contudo, ainda não se chegou a um modelo auto-similar bem definido para aplicação prática no projeto e gerenciamento dessas redes. Os modelos baseados nos processos do ruído Gaussiano fracionário e ARIMA fracionário fornecem boa representação do fenômeno da auto-similaridade.

Foi verificado que o tráfego agregado gerado por imagens monocromáticas aleatórias, não codificadas, é intrinsecamente auto-similar. Este interessante resultado é inédito, considerando a vasta bibliografia pesquisada. Se essas imagens forem codificadas o grau de auto-similaridade poderá variar. Os gráficos da variância apresentam duas regiões distintas. A inclinação da primeira região depende do tipo de agregação considerada.

Na semi-paralela esta região tem inclinação correspondente a  $H \approx 0,5$ , indicando um processo de Poisson, e se estende até o ponto relativo ao fator de agregação  $m$  aproximadamente igual ao número de imagens. Isto é explicado pelo fato das amostras agregadas no método da variância pertencerem a imagens distintas.

Para a agregação semi-seqüencial a primeira região se estende até um valor de  $m$  aproximadamente igual ao número de amostras em cada linha da imagem, e tem inclinação correspondente a  $H$  próximo de 1. A explicação, neste caso, é que as amostras agregadas pertencem à mesma imagem e, portanto, são altamente correlacionadas. Dos gráficos da variância observou-se, ainda, uma periodicidade na segunda região, denotando uma dependência entre pixels adjacentes nas linhas das imagens em decorrência do processo de agregação.

Foi verificado que os processos temporais de seqüências de vídeo não codificadas são intrinsecamente auto-similares, tendo parâmetro  $H$  mais elevado que o das imagens

monocromáticas. Este resultado também é inédito, considerando toda a bibliografia pesquisada. A auto-similaridade, neste caso, é devida a uma dependência *interquadros*, ou seja, temporal, a qual deve apresentar uma variabilidade altamente relacionada com a movimentação das cenas. Este, sem dúvida, deve ser um dos maiores problemas a serem enfrentados em caso de congestionamento das redes.

Da análise de seqüências de bits/quadro de cenas codificadas no padrão MPEG1 foi constatada a auto-similaridade no tráfego gerado por sinais de taxa de bit variável (VBR), conforme demonstrado em vários trabalhos da área. Os elevados valores de  $H$  obtidos mostram que esse tipo de tráfego pode atingir níveis críticos de auto-similaridade, trazendo talvez sérias conseqüências no desempenho da rede.

Processos auto-similares de segunda ordem, do tipo **fBm** podem ser obtidos a partir da agregação de um número  $M$  muito grande de processos inovações cujos intervalos entre períodos de atividade constituem um processo com variância infinita, desde que observados durante um tempo  $T$  muito grande, mas  $T \ll M$ .

Já os processos assintoticamente auto-similares de segunda ordem podem ser construídos pelo modelo ARIMA  $(p, d, q)$  fracionário. Os parâmetros desse modelo podem ser estimados por um método baseado em periodograma.

No presente trabalho foi elaborado um gerador de séries temporais auto-similares com parâmetro de Hurst pré-definido no intervalo entre  $0,5 < H < 1$ , baseado no modelo ARIMA  $(0, d, 0)$  fracionário de variância infinita, que tem precisão razoável para  $H \geq 0,7$ . Apesar da degradação do desempenho desse gerador para  $H \leq 0,7$ , ele ainda é superior a vários outros algoritmos apresentados na literatura.

Em resumo, as principais contribuições deste trabalho são os resultados sobre a auto-similaridade intrínseca das imagens monocromáticas e das seqüências de vídeo e o desenvolvimento do gerador de séries temporais auto-similares. Além disso, o trabalho apresentou, de forma unificada, os vários resultados da literatura sobre este assunto, onde procurou-se aclarar os conceitos mais básicos e importantes desta área emergente.

# Apêndice A

## A Geometria Fractal

### A.1. Introdução

A partir dos trabalhos de Cantor, Hausdorff e outros, Benoit Mandelbrot [6] desenvolveu a Geometria Fractal nos anos 70/80, e com isso conseguiu relacionar inúmeros fenômenos naturais a conceitos matemáticos de grande beleza e simplicidade. O termo *fractal* foi cunhado por Mandelbrot e deriva da palavra Latina *fractus* (fragmentado). Foi o uso de computadores digitais e a disponibilidade de poderosas estações de trabalho gráficas que permitiram a Mandelbrot identificar a *essência matemática* e a *recursividade algorítmica* inerentes à forma de muitos fenômenos naturais. Nuvens, rios, montanhas, costas marítimas, flocos de neve e artérias fractais simuladas, são apenas alguns exemplos de fractais desvendados por Mandelbrot e outros pesquisadores [15]. De lá para cá, a Geometria Fractal tem evoluído de forma expressiva, e sua relação com outras disciplinas, como os Sistemas Dinâmicos Caóticos, tem servido para elucidar e aperfeiçoar a modelagem de vários fenômenos físicos importantes. Invariavelmente, estes modelos fractais são simples e parcimoniosos, isto é, podem ser expressos através de poucos coeficientes ou parâmetros. Melhor ainda, a representação de fenômenos naturais aparentemente complexos com o uso desses modelos simples é em geral conseguida através de algoritmos recursivos rápidos e eficientes.

Formalmente, um fractal é um conjunto de pontos  $\mathfrak{S}$  em um espaço Euclidiano  $\mathbb{R}^n$ , normalmente  $\mathbb{R}$ ,  $\mathbb{R}^2$  ou  $\mathbb{R}^3$ , com a propriedade de que sua *dimensão de Hausdorff* é estritamente maior que sua *dimensão topológica*. Mais adiante estes conceitos serão definidos e exemplificados. Antes porém, serão apresentados alguns exemplos ilustrativos.

Uma folha de papel qualquer (de tamanho carta, por exemplo) imaginada como uma superfície ideal de pontos conectados, flexível e fixada em uma posição qualquer no espaço  $\mathbb{R}^n$ , pode ser representada por um conjunto de pontos neste espaço. A dimensão topológica desse conjunto é igual a dois, pois seus pontos podem ser representados de forma parametrizada no espaço  $\mathbb{R}^2$  (sendo impossível fazê-lo em  $\mathbb{R}^1$ ). Por definição, a dimensão topológica dessa folha não depende da forma como ela é fixada em  $\mathbb{R}^3$ , pois supondo que ela não seja rasgada, a *conectividade* de seus pontos nunca é alterada. Consequentemente, podemos amassar essa folha até o ponto onde ela assume uma *aparência esférica*, sem que ela perca a conectividade e assim mantendo sua dimensão topológica

igual a dois. Nesta nova condição, todavia, sua aparência se tornou semelhante a de uma esfera topológica de dimensão igual a três. Ela foi transformada em um *pseudo-fractal* com dimensão fractal (de Hausdorff) maior que dois e menor que três. Observada em seu interior, essa esfera é toda fragmentada e possui detalhes em várias escalas diferentes.

Exemplo análogo é o de um fino fio metálico maleável, que possui dimensão topológica igual a um, mas que, devidamente amassado e compactado, pode ser transformado em um objeto aparentemente intermediário entre um ponto e uma esfera. Sua dimensão fractal deverá ser estritamente maior que um (sua dimensão topológica).

Os dois exemplos acima indicam que um fractal não possui as características suaves dos objetos topológicos (reta, plano, sólido, etc), mas apresenta detalhes em diferentes níveis de resolução.

Um exemplo clássico de um fractal é a *folha de samambaia ideal* [7], mostrada na Figura A.1. Como pode-se ver, a imagem dessa folha é constituída de réplicas de si própria, devidamente ampliadas, rotacionadas e transladadas. Formalmente, estas operações são conseguidas com *transformações afins*, que são também a essência dos algoritmos de compressão de imagem usando geometria fractal [13]. Note que a imagem matemática dessa folha idealizada possui detalhes em todas as escalas. Porém, as limitações físicas de sua representação no papel impedem que detalhes sejam observados além de um certo nível de resolução, tanto em escalas crescentes quanto decrescentes.

Na prática, porém, o que se necessita para caracterizar um fenômeno fractal é a existência de detalhes em apenas alguns níveis de resolução. Isto é, a repetição do fenômeno de replicação pode cessar a partir de limites inferiores e superiores da escala de representação.

Devido ao efeito de replicação, os fenômenos fractais são também chamados de auto-similares, pois parte da imagem (ou conjunto) é similar a si própria. Na realidade, existem os *fractais determinísticos*, como a folha de samambaia, e os *fractais probabilísticos*, como a folha de papel amassada. Neste último caso, a auto-similaridade ocorre somente no sentido estatístico.

Um outro exemplo clássico de fractal é a curva de von Koch mostrada na Figura A.2. Note que a lei de geração dessa curva é recursiva e a figura gerada é fragmentada, auto-similar e possui detalhes em todas as escalas. A curva fechada de von Koch, na Figura A.2(b), é conhecida como “snowflake”, pois sua estrutura se assemelha à de um floco de neve.

A Figura A.3 ilustra a curva estatística de von Koch, que é obtida com o arremesso de uma moeda para se determinar de que lado da curva deve ser colocado o novo par de segmento de reta. Este fractal aleatório também possui uma estrutura fragmentada, mas a auto-similaridade estrita foi transformada em uma auto-similaridade estatística.

### A.1.1. A Dimensão Fractal

Da discussão acima, pode-se concluir que um conjunto fractal  $\mathfrak{S}$  (ideal) deve possuir as seguintes propriedades:

- $\mathfrak{S}$  possui uma estrutura fina, com detalhes em todos os níveis de resolução.



Figura A.1: Folha de samambaia idealizada (extraída de H. O. Peitgen e D. Saupe [14]).

- $\mathfrak{S}$  é muito irregular para ser descrito na linguagem geométrica tradicional, tanto localmente quanto globalmente.
- $\mathfrak{S}$  possui alguma forma de auto-similaridade, quer seja determinística ou estatística.
- Frequentemente,  $\mathfrak{S}$  é definido de uma forma muito simples (quase sempre recursiva).
- A *dimensão fractal* de  $\mathfrak{S}$  é maior que a sua dimensão topológica.

Esta última propriedade deriva da definição formal usada por Mandelbrot [6, 8, 15] para caracterizar rigorosamente um conjunto fractal. Note que a dimensão topológica de um conjunto é sempre um número inteiro não-negativo. Ela é definida de forma recursiva, partindo do postulado de que a dimensão topológica de um ponto, ou de um conjunto de pontos desconectados, é igual a zero. Por outro lado, a dimensão fractal pode ser definida de modos diferentes, porém, de certa forma, equivalentes para todos os efeitos práticos. As definições mais conhecidas são a *dimensão de caixa* (“box dimension” ou

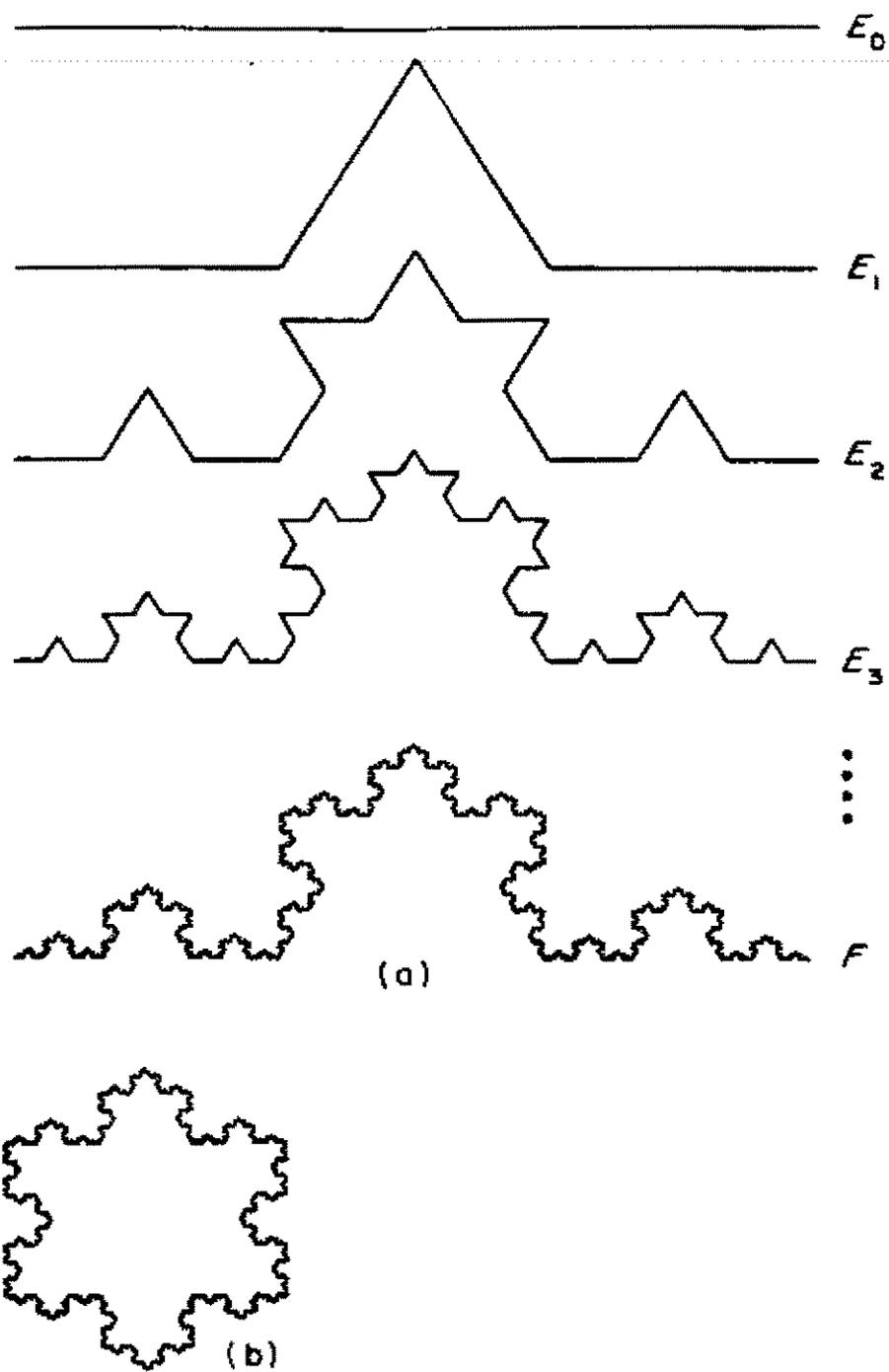


Figura A.2: A curva de von Koch (o floco de neve é composto de três dessas curvas).

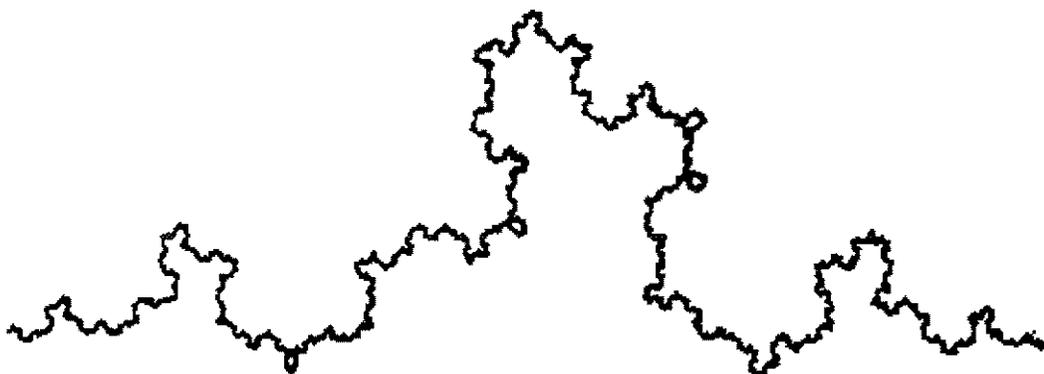


Figura A.3: A curva estatística de von Koch.

Entropia de Kolmogorov) e a *dimensão de Hausdorff*. A dimensão fractal é, em geral, um número fracionário.

Na maioria dos casos práticos, utiliza-se como definição de dimensão fractal uma forma mais simples e mais intuitiva que as duas mencionadas acima. Para entendê-la, considera-se o caso simples de um conjunto  $\mathfrak{S}_1$  correspondente a uma linha (curva ou reta) de tamanho  $L$ . Para medirmos o tamanho da linha, toma-se uma *régua rígida* de tamanho  $r$  e estima-se o menor número  $N(r)$  de régua que cobrem o tamanho  $L$ . Portanto, para  $r \rightarrow 0$  temos que  $L \approx N(r) \cdot r$ . No limite,

$$L = \lim_{r \rightarrow 0} N(r) \cdot r \quad (\text{A.1})$$

Nota-se porém que, para qualquer  $\delta > 0$ ,

$$\lim_{r \rightarrow 0} N(r) \cdot r^{1-\delta} = \infty \quad (\text{A.2})$$

e

$$\lim_{r \rightarrow 0} N(r) \cdot r^{1+\delta} = 0 \quad (\text{A.3})$$

Portanto,  $\delta = 0$  corresponde ao expoente  $D_H = 1$  de  $r^{D_H}$ , para o qual o limite de  $N(r) \cdot r^{D_H}$  converge para um número maior que zero e menor que infinito. Neste caso,  $D_H = 1$  corresponde à dimensão topológica e à dimensão fractal do conjunto  $\mathfrak{S}_1$ . Como estas dimensões são idênticas,  $\mathfrak{S}_1$  não é um fractal.

Analogamente, se  $\mathfrak{S}_2$  for considerado como uma superfície de uma folha de área  $S$ , teremos

$$S = \lim_{r \rightarrow 0} N(r) \cdot r^2 \quad (\text{A.4})$$

e, para  $\delta > 0$ ,

$$\lim_{r \rightarrow 0} N(r) \cdot r^{2-\delta} = \infty \quad (\text{A.5})$$

e

$$\lim_{r \rightarrow 0} N(r) \cdot r^{2+\delta} = 0 \quad (\text{A.6})$$

Neste caso,  $N(r)$  é o menor número de quadrados de lado  $r$  que cobrem a área  $S$ . Aqui as dimensões topológicas e fractais são ambas iguais a dois.

O mesmo raciocínio pode ser aplicado a um conjunto  $\mathfrak{S}_3$  correspondente a um sólido com dimensão topológica (e fractal) igual a três.

A discussão acima sugere a seguinte definição de dimensão fractal:

- **Definição:** A dimensão fractal  $D_H$  de um conjunto  $\mathfrak{S}$  é o número real não negativo para o qual o limite de

$$N(r) \cdot r^{D_H} \quad (\text{A.7})$$

quando  $r \rightarrow 0$ , converge para um valor finito e maior que zero.

Vários autores utilizam esta definição como sendo a da dimensão de Hausdorff [15, 14]. Deve-se enfatizar, todavia, que a dimensão de Hausdorff é definida de forma mais rigorosa e, em muitos casos, fornece valores diferentes daqueles obtidos com a dimensão fractal acima.

Note que, se  $N(r) \cdot r^{D_H} \cong c$  ( $0 < c < \infty$ ), então

$$\log c \cong \log N(r) + D_H \log r \quad (\text{A.8})$$

e

$$D_H = \frac{\log N(r)}{-\log(r)} + \frac{\log c}{\log r} \quad (\text{A.9})$$

Tomando o limite ( $r \rightarrow 0$ ), o segundo termo tende a zero e, então,

$$D_H = \lim_{r \rightarrow 0} \frac{\log N(r)}{\log(1/r)} \quad (\text{A.10})$$

Esta expressão é muito útil para aplicações práticas, pois  $D_H$  pode ser estimado como o gradiente do gráfico log-log plotado em uma região conveniente de  $r$ , como na Figura A.4.

Mandelbrot define então como fractal, todo conjunto que possui uma dimensão de Hausdorff  $D_H$  estritamente maior que sua dimensão topológica.

Utilizando a expressão acima podemos calcular a dimensão fractal  $D_H$  da curva de von Koch na Figura A.2. Para isso, tomando-se  $r = 1/3^n$  para a  $n$ -ésima iteração, temos que  $N(r)$  é proporcional a  $4^n$ . Assim,

$$D_H = \frac{\log(4^n)}{\log(3^n)} = \frac{\log 4}{\log 3} \cong 1,26 \quad (\text{A.11})$$

Como a dimensão topológica da curva de von Koch é igual a um e  $D_H = 1,26 > 1$ , então ela é realmente um fractal segundo a definição de Mandelbrot.

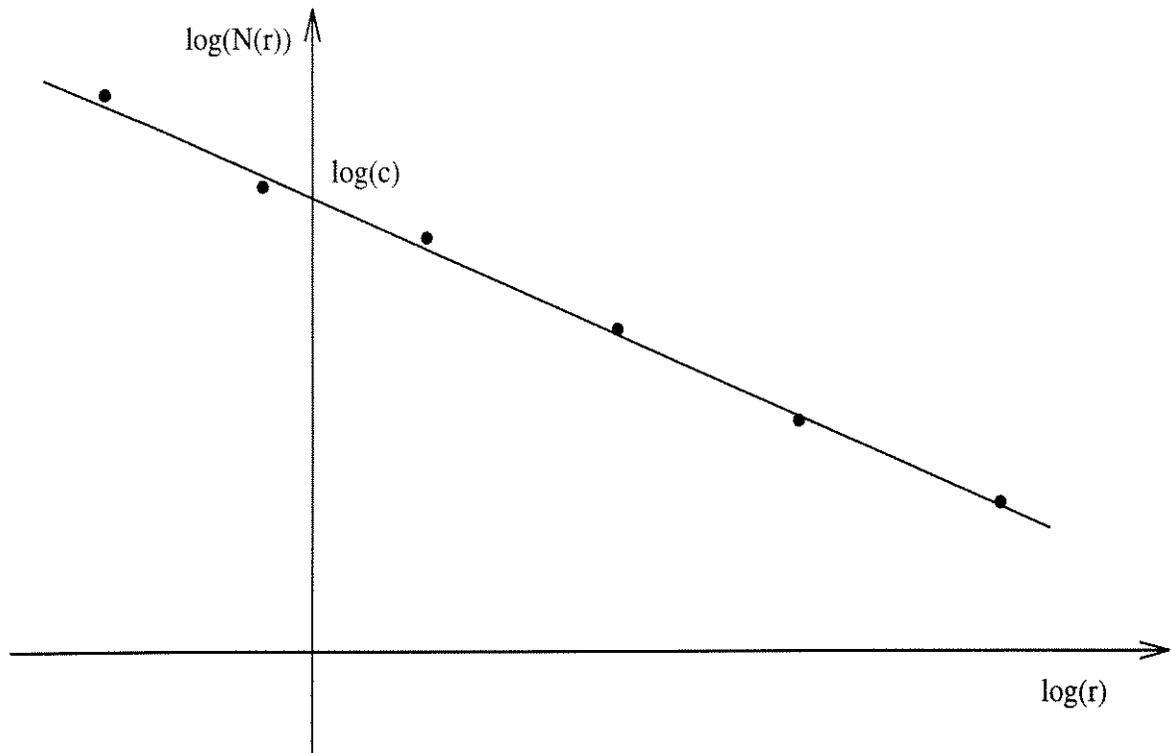


Figura A.4: Gráfico  $\log - \log$  de  $N(r)$  em função de  $r$ , para o cálculo da dimensão fractal.

Outro exemplo muito interessante de fractal surge quando se observa a costa marítima de diferentes países [14]. A Figura A.5 mostra os gráficos  $\log - \log$  do comprimento  $L(r)$  de diversas costas marítimas e fronteiras em função do comprimento  $r$  da régua utilizada. Como  $L(r) = N(r) \cdot r$ , é de se esperar que  $L(r)$  aumente à medida que  $r$  diminui, pois mais e mais reentrâncias do litoral são incorporadas a  $L(r)$  quando  $r$  tende a zero.

Tendo em vista que para  $r$  pequeno

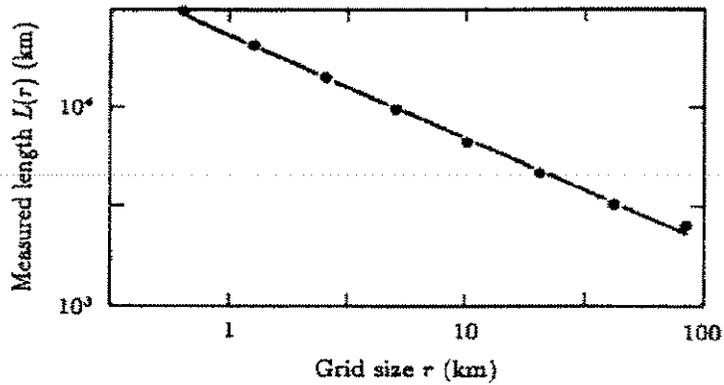
$$N(r) \cdot r^{D_H} \simeq c \quad (\text{A.12})$$

então,

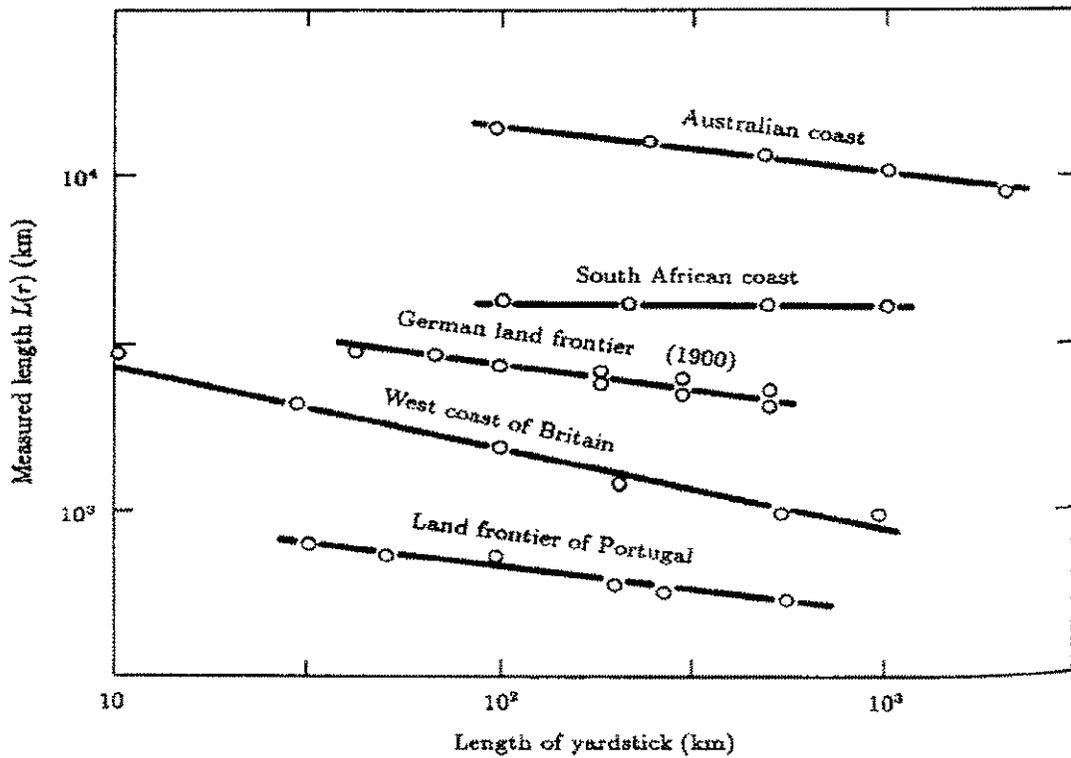
$$L(r) = N(r) \cdot r \simeq c \cdot r^{-D_H} \cdot r = c \cdot r^{1-D_H} \quad (\text{A.13})$$

Em uma escala  $\log - \log$ , temos

$$\log L(r) = \log c + (1 - D_H) \log r \quad (\text{A.14})$$



(a)



(b)

Figura A.5: Características fractais de diversas costas marítimas [15].

Portanto, se  $\xi$  for a declividade de uma reta na Figura A.5, a dimensão fractal correspondente será  $D_H = 1 - \xi$ . Por exemplo, a costa da Noruega possui  $\xi = -0,52$  e portanto uma dimensão fractal  $D_H = 1,52$  (uma dimensão intermediária entre uma linha e uma superfície!).

No extremo oposto de costa fragmentada, situa-se a costa Sul-Africana com  $D_H \simeq 1,0$  (bem próximo da geometria de uma linha topológica). As costas dos outros países possuem dimensões intermediárias.

Mandelbrot estudou também o comportamento de funções iteradas, o que lhe permitiu obter interessantes propriedades dos fractais. Por exemplo, o Conjunto Cardióide de Mandelbrot, chamado conjunto  $M$ , consiste de todos os pontos  $c$  no plano complexo  $Z$ , para os quais a iteração

$$z_{n+1} = z_n^2 + c \quad (\text{A.15})$$

com  $z_0 = 0$ , permanece limitada quando  $n \rightarrow \infty$ . Nesta mesma linha de raciocínio, Michael Barnsley [12] introduziu os sistemas de funções iteradas (IFS), que têm sido utilizados com muito sucesso em algoritmos bastante eficientes de compressão de imagem [13].

### A.1.2. Exemplo: O Movimento Browniano Fracionário (fBm)

Um exemplo simples e muito importante de fractal aleatório é o *Movimento Browniano* (**Bm**) em uma variável, já apresentado no capítulo 2. Se  $X(t)$  é uma amostra desse processo estocástico, então  $X(t_1)$ ,  $X(t_2)$ , etc., são variáveis aleatórias reais nos instantes  $t_i$ 's.

Estes processos possuem média zero e incrementos

$$X(t_i) - X(t_j)$$

que são variáveis Gaussianas independentes sempre que calculadas em intervalos de tempo que não se sobrepõem. Além disso, o valor quadrático médio dos incrementos é proporcional à magnitude dos intervalos de tempo, isto é,

$$E \left[ |X(t_i) - X(t_j)|^2 \right] \propto |t_i - t_j| \quad (\text{A.16})$$

Em vista disso, os incrementos de  $X$  são *estatisticamente auto-similares* no sentido que

$$X(t_0 + t) - X(t_0) \quad e \quad \frac{1}{\sqrt{r}} (X(t_0 + rt) - X(t_0))$$

possuem a mesma distribuição conjunta para quaisquer  $t_0$  e  $r > 0$ .

Para facilitar, podemos tomar  $t_0 = 0$  e  $X(t_0) = 0$  e verificar que os dois processos aleatórios

$$X(t) \quad e \quad \frac{1}{\sqrt{r}} X(rt) \quad (\text{A.17})$$

são estatisticamente indistinguíveis para qualquer  $r > 0$ . Note que o segundo processo é simplesmente o primeiro devidamente reescalado. Observe também que se  $X(t)$  for *acelerado no tempo* por um fator  $r > 1$ , então sua amplitude deve ser *reduzida* de um fator  $\frac{1}{\sqrt{r}} < 1$ .

Uma simulação desse fenômeno de auto-similaridade é mostrado na Figura A.6 [15] para sete valores de  $r$  entre  $1/8$  e  $8$ . Note que as diversas amostras dos sete processos aleatórios são praticamente idênticas do ponto de vista estatístico.

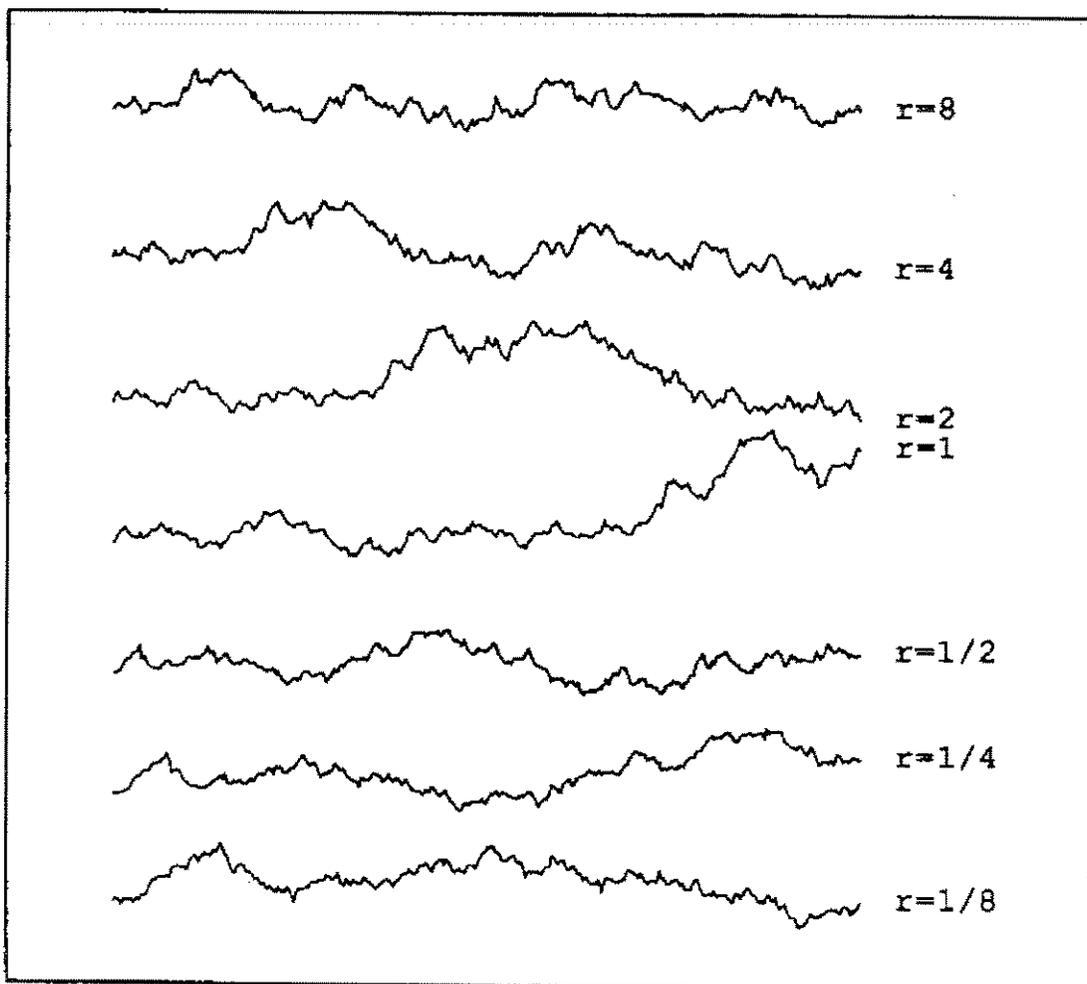


Figura A.6: Sete amostras de diferentes processos fracionários Brownianos (fBm).

Além da *semelhança visual* nas amostras  $X(t)$  da Figura A.6, poderia se perceber também uma *afinidade auditiva* nessas amostras se elas fossem executadas em um toca-fitas. Um ouvinte qualquer muito provavelmente iria dizer que todas as sete amostras provêm de uma mesma fonte aleatória. Neste caso, se o toca-fitas possuísse controle de velocidade e de volume (amplitude), então o ouvinte poderia executar uma das amostras multiplicando a velocidade por um fator  $r$  e *simultaneamente* dividindo o volume por um fator  $\sqrt{r}$ . Estatisticamente, nenhuma alteração na qualidade do som deverá ser notada quando estes dois parâmetros forem variados simultaneamente.

Estes processos não são rigorosamente auto-similares, pois vistos como um gráfico no plano amplitude  $\times$  tempo, essas duas coordenadas não são reduzidas ou ampliadas na mesma proporção quando se deseja *auto-replicar* o gráfico. Por isso, eles são mais

precisamente denominados *auto-afins*. Na literatura, todavia, os dois termos têm sido usados indistintamente.

O processo Browniano, apresentado acima, pode ser considerado um caso particular do Movimento Browniano Fracionário (“fractional Brownian motion - **fBm**”), introduzido por Mandelbrot e Van Ness [9]. Nos processos **fBm** os incrementos também são Gaussianos (mas não independentes) e suas variâncias satisfazem a expressão

$$E [|X(t_i) - X(t_j)|^2] \propto |t_i - t_j|^{2H} \quad (\text{A.18})$$

onde  $0 < H < 1$ . Note que  $H = 1/2$  corresponde ao movimento Browniano puro.

Analogamente, os incrementos de  $X(t)$  dos **fBm**'s são ditos *estatisticamente auto-similares com parâmetro  $H$* , isto é,

$$X(t_0 + t) - X(t_0) \quad e \quad \frac{1}{r^H} (X(t_0 + rt) - X(t_0))$$

possuem as mesmas funções de distribuição conjuntas para quaisquer  $t_0$  e  $r > 0$ . Fazendo novamente  $t_0 = 0$  e  $X(t_0) = 0$ , as duas funções aleatórias

$$X(t) \quad e \quad \frac{1}{r^H} X(rt) \quad (\text{A.19})$$

são estatisticamente indistinguíveis. Assim, um **fBm** acelerado,  $X(rt)$ , deve ser necessariamente reescalado dividindo-se as amplitudes por  $r^H$ .

Já foi visto anteriormente o comportamento de  $X(t)$  quando  $H = 1/2$ . O que ocorre quando  $H \neq 1/2$ ? É interessante observar que para  $H = 0$  ( $r^H = 1$ ), as funções  $X(t)$  e  $X(rt)$  são aparentemente idênticas no sentido estatístico, independentemente da expansão ou contração que se usar no eixo do tempo. Observada em um osciloscópio, uma amostra  $X(rt)$  teria a mesma aparência para qualquer base de tempo! É claro que este comportamento só pode ocorrer pelo fato do gráfico de  $X(t)$  ser denso no plano amplitude  $\times$  tempo. Portanto, é de se esperar que a dimensão fractal dessas amostras seja igual a 2. Na realidade, pode-se demonstrar que os gráficos de amostras de processos **fBm** possuem dimensão fractal igual a  $2 - H$ , valor que coincide também com a dimensão de Hausdorff. Assim, o movimento Browniano puro ( $H = 1/2$ ) possui dimensão fractal igual a 1,5.

No outro extremo tem-se o caso onde  $H = 1$ . Aqui as funções  $X(t)$  e  $\frac{1}{r} X(rt)$  são realmente auto-similares, pois a contração ou expansão é realizada igualmente nos dois eixos. Neste caso as amostras são bastante suaves, o que condiz com o fato de sua dimensão fractal ser igual a  $D_H = 2 - H = 1$ . Processos como este variam lentamente no tempo e, pelo fato de  $D_H$  ser igual a um, eles não são fractais no sentido estrito (possuem auto-similaridade mas não apresentam granularidade). Note que o parâmetro  $H$  indica o grau de *aspereza ou granularidade* das funções em pequenas escalas.

Os processos Brownianos fracionários podem ser divididos em três categorias importantes, correspondentes a  $H < 1/2$ ,  $H = 1/2$  e  $H > 1/2$ . Para  $H = 1/2$  temos o processo Browniano puro, onde os incrementos são independentes e porisso descorrelacionados. Para  $H > 1/2$  existe uma correlação positiva entre os incrementos; se o gráfico de  $X(t)$  estiver crescendo para algum  $t_0$ , então a tendência é ele continuar crescendo para  $t > t_0$ . Por outro lado, para  $H < 1/2$ , a correlação é negativa e a tendência das curvas é oscilar

caoticamente. Para complementar os gráficos da Figura 3.12 (caso  $H = 1/2$ ), as Figuras A.7 (a) e (b) mostram outra série de amostras para os casos  $H = 0.2$  e  $H = 0.8$ . Os comportamentos discutidos acima são bem ilustrados nessas figuras.

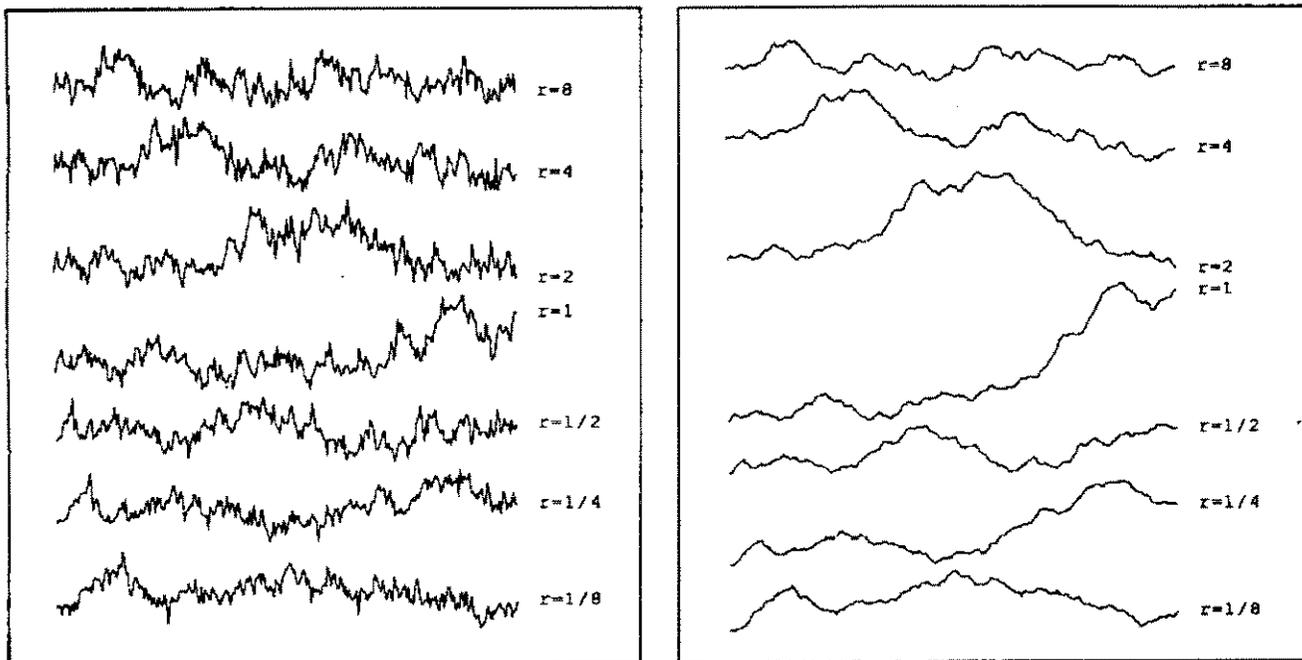


Figura A.7: Várias amostras de ruído **fBm** rescalonadas, para  $H = 0,2$  (esquerda) e  $H = 0,8$  (direita) e vários valores de  $r$ . Note como o processo é realmente auto-similar.

Pode-se mostrar que os processos **fBm** possuem uma função de densidade espectral de potência dada por uma *lei de potência*, isto é,

$$S(f) \propto 1/f^\alpha \quad (\text{A.20})$$

onde  $\alpha = 2H + 1$ . Na região de interesse  $\alpha$  toma valores entre 1 e 3, pois  $0 < H < 1$ . Na Figura A.8 são apresentadas as densidades espectrais de potência e amostras típicas para  $\alpha = 1$  e  $\alpha = 2$ , bem como para o caso  $\alpha = 0$  que corresponde ao ruído Gaussiano branco. O processo obtido com  $\alpha = 1$  é denominado *ruído de cintilação* (“flicker noise”) e corresponde a inúmeros fenômenos físicos interessantes [15]. Por exemplo, descobriu-se recentemente que quase todos os tipos de música (clássica, popular, etc), possuem características muito próximas das do ruído de cintilação [15, 14]. Músicas de “rock” não se enquadram na categoria  $1/f$ !

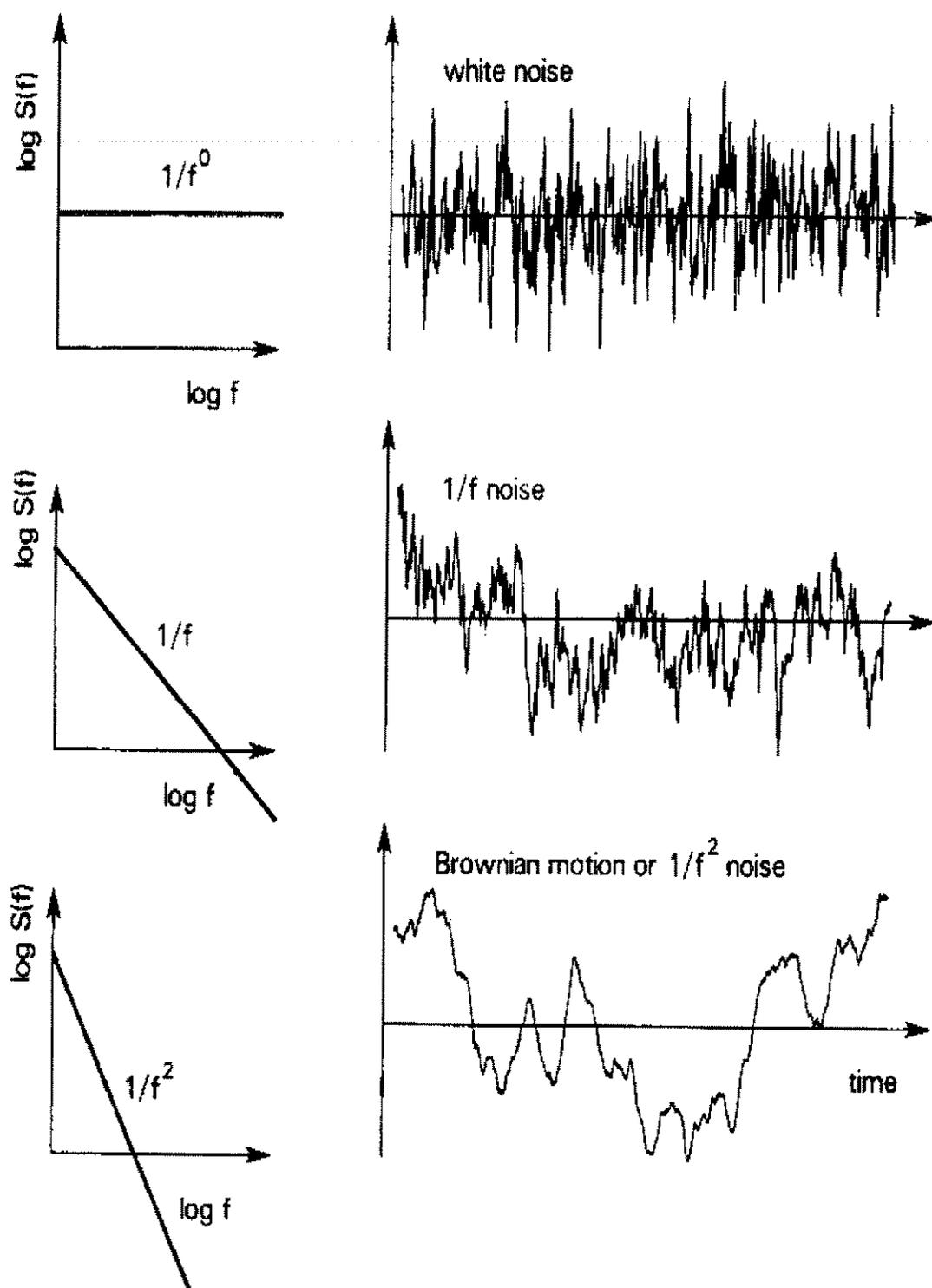


Figura A.8: Amostras típicas de ruídos fractais que obedecem à lei da potência, para  $\alpha = 0$ ,  $\alpha = 1$  e  $\alpha = 2$ .

Curiosamente, já foi observado por vários pesquisadores que séries temporais de

fenômenos naturais (secas, enchentes, variações de temperaturas, etc.), quando devidamente reescaladas, seguem as propriedades dos ruídos **fBm** com valores de  $H$  próximos de 0,75. Mas note que estes valores correspondem a processos estocásticos lentos e com baixo valor de dimensão fractal. É exatamente esta *persistência temporal* que implica em calamidades climáticas de longa duração. Dados convincentes sobre enchentes nas margens de grandes rios (Nilo e outros) são apresentados por alguns autores e parecem explicar até certas passagens bíblicas. Por estas e outras razões, os vários ruídos acima são frequentemente chamados de *ruído cor-de-rosa* ( $\alpha = 1$ ), *ruído marron* ( $\alpha = 2$ ) e *ruído negro* ( $\alpha \simeq 3$ ), além obviamente do *ruído branco* ( $\alpha = 0$ ).

# Apêndice B

## Listagem dos Programas Fontes

Neste apêndice são apresentadas as listagens de todos os programas implementados para a análise de processos auto-similares. Os programas foram implementados principalmente no MATLAB 4.0 para Windows; alguns foram também elaborados no MATHEMATICA 2.2, em versão para UNIX. São descritos ainda os programas para o cálculo da variância dos pixels das imagens monocromáticas e dos quadros das seqüências de vídeo, utilizadas no Capítulo 3. Espera-se que os comentários introduzidos possam facilitar o trabalho de eventuais usuários desses programas.

### B.1. Programas para o Gráfico da Variância e “Pox Plot”

```
%***** Grafico da Variancia *****
%
%               **MATLAB 4.0**
%
%*** agregacao das amostras e calculo da variancia *****
%
% dados -sequencia de amostras
% em    -vetor composto dos fatores de agregacao (m) cujo valor
%        maximo e' igual a metade do numero total de
%        amostras, para que log[var(Xm)] nao seja
%        indeterminado (var(Xm)= 0)
% k     -numero de subconjuntos de amostras para um dado grau
%        de agregacao do vetor x
% A     -matriz cujas colunas de m elementos sao subconjuntos
%        de amostras do vetor x
% xm    -vetor composto das medias das colunas da matriz A
% varm  -vetor composto das variancias referentes aos fatores
%        de agregacao normalizado pela variancia do conjunto
%        total de amostras
```

```

%

clear;
load -ascii dados
y=dados;

% ***** Calculo da Variancia *****

em = [2:2:(length(y)/2)];
for i = 1:length(em),
    clear A;
    k = length(y)/em(i);
    l = 1;
    p = em(i);
    for j = 1:k,
        A(1:em(i),j) = y(l:p)';
        l = l+em(i); p = p+em(i);
    end;
    xm = mean(A);
    varm(i) = cov(xm)/cov(y);
end;

% *** interpolacao da reta por minimos quadrados *****
% pol - vetor composto dos coeficientes da reta
% pol(1) - inclinacao da reta
% ii - indice correspondente ao inicio da reta
% jj - indice correspondente ao final da reta

em2 = em(ii:jj);
varm2 = varm(ii:jj);
pol = polyfit(log10(em2),log10(varm2),1);
reta = pol(1).*log10(em2)+pol(2);

% *** plotagem do grafico *****

plot(log10(em),log10(varm),'.',log10(em2),reta,'b-');
xlabel('log(m)');
ylabel('log[var(Xm)]');
title('grafico da variancia');

```

```

% Programa para o Calculo do "Pox Plot" da Estatistica R/S
% de um Conjunto de Amostras *
%
%
%           ***MATHEMATICA  2.2***
%
%
% dados -arquivo contendo duas colunas, sendo a primeira
%        o instante de tempo e a segunda a amostra de trafego
% en     -vetor composto dos fatores de agregacao "n"
% x      -vetor composto do conjunto de amostras a serem
%        analisadas
% A      -matriz de agregacao -as colunas de "n" elementos sao
%        subconjuntos nao sobrepostos do vetor x
% total -numero total de amostras
%
%*** O vetor "en" utilizado e' apenas um exemplo *****

clear;
load -ascii dados;
x = dados(1:total,2);
en = [10:1:10000];

%*** Calculo de R/S para cada fator de agregacao *****

for i = 1:length(en),
    clear wk;
    k =length(x)/en(i);

% *** construcao da matriz de agregacao (A) *****

    l = 1;
    p = en(i);
        for j = 1:k,
            A(1:en(i),j) = x(1:p)';
        end;
    clear j;
    clear l;
    clear p;

% ** calculo da media e do desvio padrao das colunas de A
%   referentes ao fator de agregacao en(i)

    med = mean(A);
    des = std(A);

```

```

% ** calculo de  $W(l)=[X(1)+X(2)+\dots+X(l)]-[l E(X)]$ ,  $l=1,2,\dots$ 
%   ..,en(i), para cada subconjunto (colunas da matriz A)
%
% ** w e' uma matriz com numero de colunas igual ao de A,
%   ou seja, k

    for b = 1:k,
for r = 1:en(i),
    w(r,b) = sum(A(1:r,b))-(r*med(b));
        end;
    end;
clear A;
clear med;
clear r;
clear b;

% ** calculo de  $W_k(l) = W(l)-W(1)$ ,  $l=2,\dots,en(i)$  e  $W_k(1)=0$ ,
%   para cada subconjunto
%
% ** wk e' uma matriz com numero de colunas igual ao de A

    for a = 1:k,
wk(1,a) = 0;
        for c = 1:(en(i)-1),
            wk(c+1,a) = w((c+1),a)-w(1,a);
        end;
    end;
clear a;
clear c;
clear w;

% ** calculo da estatistica  $R/S = 1/S[\max(W_k)-\min(W_k)]$  ***

    mm = max(wk)-min(wk);
    for z = 1:k,
rs(z,i) = log10((1/des(z))*mm(z));
        end;
    clear z;
    clear des;
end;

% ***** plotagem do diagrama *****

```

```

plot(log10(en),rs,'r. ');
xlabel('log(n)');
ylabel('log(r/s)');
title('"PoX Plot" da Estatistica R/S');
hold;

```

(\* Este programa tem por finalidade gerar os pontos do diagrama "poxplot" de uma serie temporal qualquer. Basicamente, o programa trata de calcular a estatistica R/S em particoes da serie fornecida. O numero de amostras consideradas na particao aumenta de acordo com o valor do incremento "step" ate' o nivel de agrupamento desejado. O programa assume que o arquivo de entrada esteja em formato ASCII e que os pontos da serie estejam dispostos em uma unica coluna. A saida gerada encontra-se na forma de uma "lista", conceito utilizado pelo Mathematica (Lists), na qual esta' a colecao de pares de pontos obtida.\*)

(\* Programa Principal \*)

(\* Definicao do diretorio de trabalho e leitura do arquivo de entrada "arq" (a serie temporal) \*)

```

SetDirectory["/tmp"];
arquivo=ReadList["arq",Number,RecordLists->True];

```

(\* Determinacao do numero total de elementos da serie existentes em "arquivo". Este valor e' guardado na variavel "nt" \*)

```

d=Dimensions[arquivo];
nt=d[[1]];

```

(\* Definicoes da media amostral e do desvio padrao \*)

```

mean[list_]:=Apply[Plus,list]/Length[list];
std[list_]:=Sqrt[mean[(list-mean[list])^2]];

```

(\* Definicao do incremento do valor da particao,

```

    inicializacao do vetor de saida "xn" e valor inicial da
    particao *)

step=1;
xn={};
n=8;

(* Definicao do tamanho maximo da particao. No caso, optou-se
    por um decimo do comprimento total da serie para efeito de
    numero de pontos amostrais no ultimo estagio de
    agrupamento.*)

nmax=Round[nt/10] //N;

(* Inicializacao do laco que determina o comprimento da
    particao a ser considerado *)

While[n<=nmax,

(* ti=posicao inicial da particao em questao *)

ti=1;

(* Laco para determinacao da particao a ser considerada *)

While[ti<=(nt-n+1),

(* xp=vetor temporario de amostras da particao *)

xp={arquivo[[ti]]};
Do[xp=Append[xp,arquivo[[ti-1+1]],{1,2,n,1}];

(* Calculo da media e desvio padrao das amostras da
    particao *)

x=mean[xp] //N;
s=std[xp] //N;
Clear[xp]; (* Apaga vetor temporario *)

(* Calculo da estatistica R/S *)

w={0}; (* Inicializacao do vetor "w" (vide definicao
    dada no calculo da estatistica R/S) *)
soma=0;
Do[

```

```

soma=soma+arquivo[[ti-1+1]];
w=Append[w,soma-1*x],
{1,n}
];

(* Calcula os valores maximo e minimo da sequencia "w" *)

maximo=Max[w];
minimo=Min[w];
Clear[w];
If[s!=0,
rs=(maximo-minimo)/s //N; (* estatistica R/S *)
xn=Append[xn,{n,rs}],
a=1 (* variavel auxiliar, devido a sintaxe do If *)
];
ti=ti+n
];
n=n+step
];

(* Transformacao para escala logaritmica *)

xn=Log[10,xn] //N;
xn >> saida; (* Salva o vetor de pontos em "saida" *)

```

## B.2. Programa para o Cálculo da Variância das Imagens Monocromáticas

O cálculo da variância dos blocos de  $4 \times 4$  e  $8 \times 8$  foi realizado através da linguagem de programação visual *Cantata* do ambiente *Khoros 2.0*, como mostrado na Figura B.1. baseia-se na estimativa da variância de um conjunto  $X$  de  $M$  amostras que é dada por

$$\sigma^2(x) = E[x^2] - \bar{x}^2$$

sendo

$$E[x^2] = \frac{1}{M} \sum_{i=1}^M x_i^2 \quad \text{e}$$

$$\bar{x}^2 = \left( \frac{1}{M} \sum_{i=1}^M x_i \right)^2$$

Esse cálculo foi feito para cada imagem de  $16 \times 256$  pixels de maneira a se obter as variâncias de todos os blocos de  $4 \times 4$  (ou  $8 \times 8$ ) simultaneamente. Assim, a imagem na saída teve dimensões de  $4 \times 64$ , para blocos de  $4 \times 4$ , ou  $2 \times 32$ , para blocos de  $8 \times 8$ , e

a intensidade dos pixels correspondeu à variância dos blocos (amostras de atividade da imagem original). A seguir fez-se o “empilhamento” dessas imagens de saída. Obteve-se assim uma imagem de  $(N \times 4) \times 64$  (ou  $(N \times 2) \times 32$ ) pixels. A matriz contendo as intensidades dos pixels foi agregada nas formas semi-paralela e semi-sequencial, conforme apresentado no Capítulo 3.

A média foi implementada pela convolução da imagem por uma matriz (imagem) quadrada tipo *impulso unitário* dividida pelo número de pixels do bloco (16 ou 64); seguida da aplicação de uma “máscara”, visando a seleção dos valores correspondentes a blocos não sobrepostos; e por último, uma redução da imagem para compensar a expansão causada pela convolução. A intensidade dos pixels desta imagem resultante correspondeu à média dos blocos. A média quadrática foi calculada de forma semelhante, sendo que as intensidades dos pixels foram elevadas ao quadrado antes da convolução. A Figura B.2 corresponde ao cálculo da média.

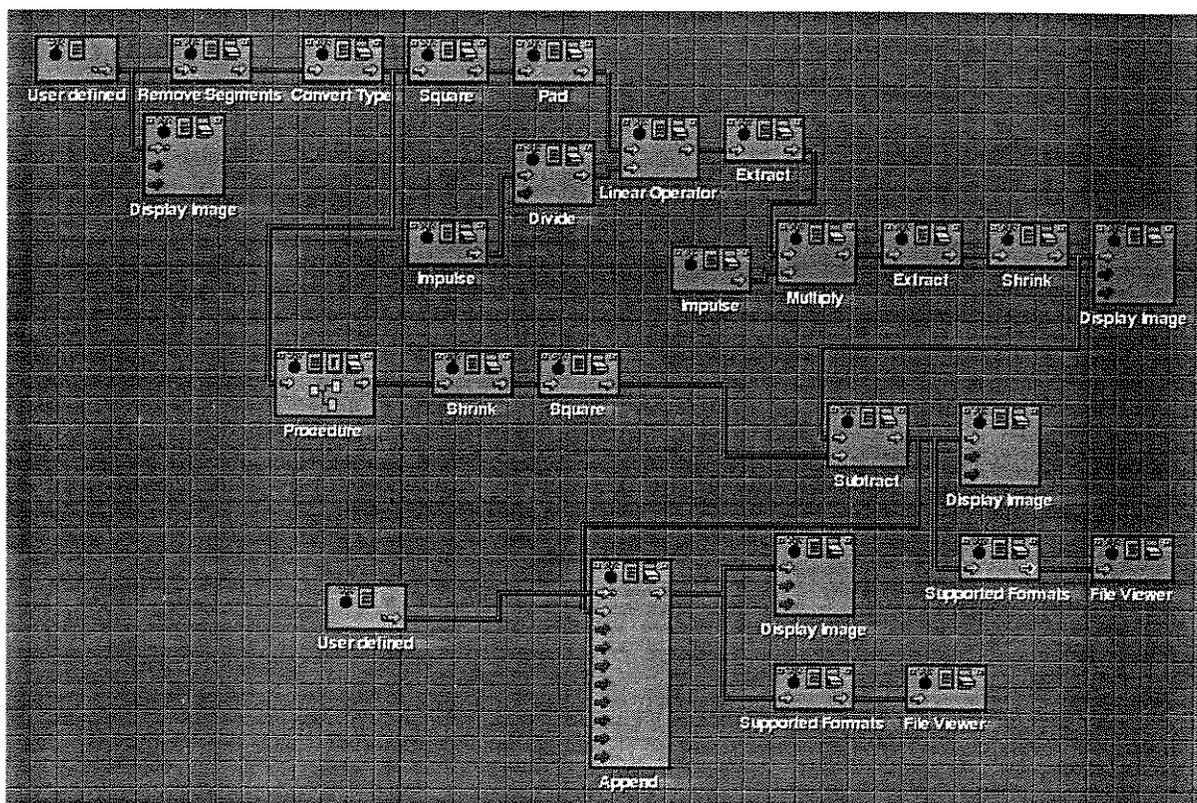


Figura B.1: Programa elaborado na linguagem *Cantata* do Ambiente *Khoros 2.0* para o cálculo da variância dos blocos de  $4 \times 4$  ou  $8 \times 8$ . O bloco “PROCEDURE” é apresentado na figura abaixo.

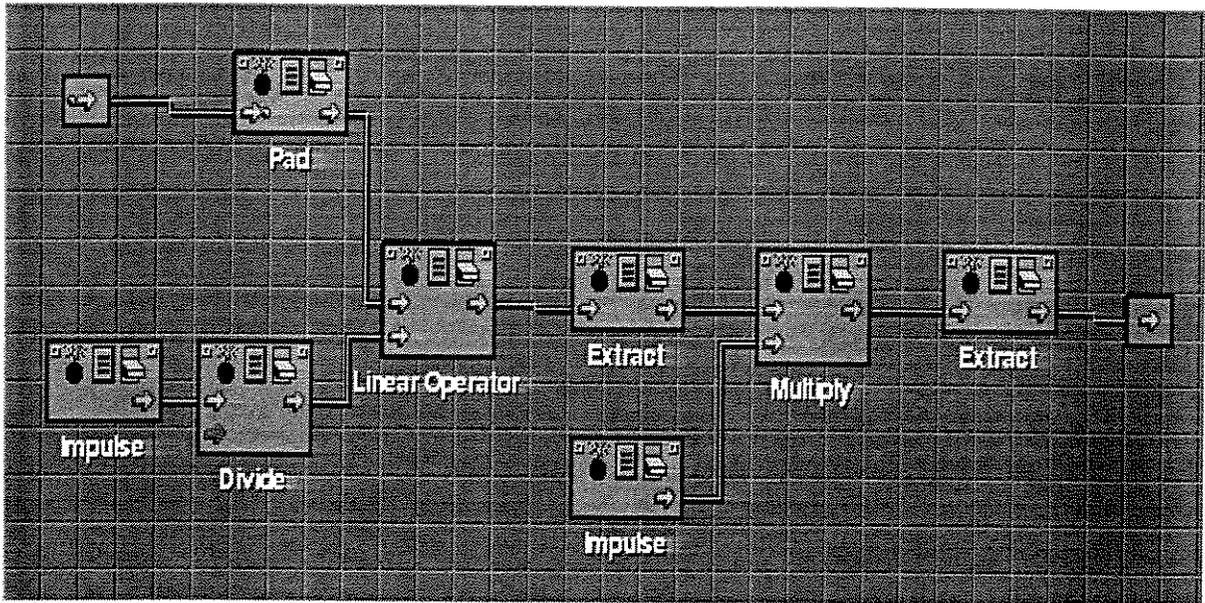


Figura B.2: Procedimento para o cálculo da média dos blocos de  $4 \times 4$  ou  $8 \times 8$ .

A seguir é apresentada a listagem correspondente ao programa da Figura B.1.

```
# Khoros Visual Programming Workspace
#
# cantata workspace file (/.../conv.wk) was created
# on Wed Mar 12 13:45:40 1997
# by user albanita
#
VariablesBegin:

# No variables currently declared. #

VariablesEnd:

#GlyphBegin

# Glyph 'Display Image'
Glyph:ENVISION:putimage::3:105:185:: -use_cmap 1

# Glyph 'Shrink'
Glyph:DATAMANIP:kshrink::7:805:105:: -wmag '0.25' -hmag '0.25'

# Glyph 'Display Image'
Glyph:ENVISION:putimage::11:885:105:: -use_cmap 1

# Glyph 'Remove Segments'
```

```

Glyph:DATAMANIP:krmseg::15:125:105:: -map

# Glyph 'Convert Type'
Glyph:DATAMANIP:kconvert::19:225:105:: -type 5 -scale '1' -roff
'0' -ioff '0' -val

# Glyph 'User defined'
Glyph:DESIGN:user_def::23:25:105:: -o ./imagem2/imag104.xpm

# Glyph 'Square'
Glyph:DATAMANIP:ksqr::27:305:105::

'Procedure'
ProcedureBegin:Procedure:31:205:305:
runwks -o /usr/tmp/ioQBAA005Bz -i /usr/tmp/ioKAAa002Kw

VariablesBegin:

# No variables currently declared. #

VariablesEnd:

#GlyphBegin

# Glyph 'Extract'
Glyph:DATAMANIP:kextract::3:645:245: o!o: -wsize '256' -hsize '16'
-woff '3' -hoff '3' -subpos 0 -o /usr/tmp/ioQBAA005Bz

# Glyph 'Multiply'
Glyph:DATAMANIP:kmul::9:545:245::

# Glyph 'Impulse'
Glyph:DATAMANIP:kimpulse::13:445:325:: -wsize '258' -wnum '64'
-hsize '18' -hnum '4' -dsize '1' -dnum '1' -tsize '1' -tnum '1'
-esize '1' -enum '1' -wsp '4' -woff '3' -hsp '4' -hoff '3'
-dsp '1' -doff '0' -tsp '1' -toff '0' -esp '1' -eoff '0'
-fgreal '1.0' -bgreal '0.0' -fgimag '0.0' -bgimag '0.0' -type 10

# Glyph 'Extract'
Glyph:DATAMANIP:kextract::17:445:245:: -wsize '258' -hsize '18'
-woff '2' -hoff '2' -subpos 0

# Glyph 'Pad'
Glyph:DATAMANIP:kpad::21:263:163: !i!i: -wsize '260' -hsize '20'
-woff '2' -hoff '2' -doff '0' -toff '0' -eoff '0' -real '0'

```

```

-imag '0' -valid 1

# Glyph 'Linear Operator'
Glyph:DATAMANIP:klinearop::27:343:243:: -flip 0 -l -real '0'
-imag '0' -size 1 -w -h
.....

# Glyph 'Divide'
Glyph:DATAMANIP:kdiv::31:245:285:: -real '16'

# Glyph 'Impulse'
Glyph:DATAMANIP:kimpulse::35:165:285:: -wsize '4' -wnum '4'
-hsize '4' -hnum '4' -dsize '1' -dnum '1' -tsize '1' -tnum '1'
-eseize '1' -enum '1' -wsp '1' -woff '0' -hsp '1' -hoff '0'
-dsp '1' -doff '0' -tsp '1' -toff '0' -esp '1' -eoff '0'
-fgreal '1.0' -bgreal '0.0' -fgimag '0.0' -bgimag '0.0' -type 10

#GlyphEnd
#ConnectionBegin

# 'Extract'
NodeConnection:3:o:4:o:

# 'Multiply'
NodeConnection:9:o:3:i:

# 'Impulse'
NodeConnection:13:o:9:i2:

# 'Extract'
NodeConnection:17:o:9:i1:

# 'Pad'
NodeConnection:21:o:27:i1:

# 'Linear Operator'
NodeConnection:27:o:17:i:

# 'Divide'
NodeConnection:31:o:27:i2:

# 'Impulse'
NodeConnection:35:o:31:i1:

#ConnectionEnd

```

#ExternalConnectionEnd

AnnotationsBegin

AnnotationsEnd

ProcedureEnd:Procedure

# Glyph 'Extract'

Glyph:DATAMANIP:kextract::35:725:105:: -wsize '256' -hsize '16'  
' -woff '3' -hoff '3' -subpos 0

# Glyph 'Multiply'

Glyph:DATAMANIP:kmul::39:645:105::

# Glyph 'Impulse'

Glyph:DATAMANIP:kimpulse::43:545:185:: -wsize '258' -wnum '64'  
-hsize '18' -hnum '4' -dsize '1' -dnum '1' -tsize '1' -tnum '1'  
-esize '1' -enum '1' -wsp '4' -woff '3' -hsp '4' -hoff '3'  
-dsp '1' -doff '0' -tsp '1' -toff '0' -esp '1' -eoff '0'  
-fgreal '1.0' -bgreal '0.0' -fgimag '0.0' -bgimag '0.0' -type 10

# Glyph 'Extract'

Glyph:DATAMANIP:kextract::47:565:105:: -wsize '258' -hsize '18'  
-woff '2' -hoff '2' -subpos 0

# Glyph 'Pad'

Glyph:DATAMANIP:kpadd::51:385:105:: -wsize '260' -hsize '20'  
-woff '2' -hoff '2' -doff '0' -toff '0' -eoff '0' -real '0'  
-imag '0' -valid 1

# Glyph 'Linear Operator'

Glyph:DATAMANIP:klinearop::55:485:105:: -flip 0 -l -real '0'  
-imag '0' -size 1 -w -h

# Glyph 'Divide'

Glyph:DATAMANIP:kdiv::59:385:165:: -real '16'

# Glyph 'Impulse'

Glyph:DATAMANIP:kimpulse::63:305:225:: -wsize '4' -wnum '4'  
-hsize '4' -hnum '4' -dsize '1' -dnum '1' -tsize '1' -tnum '1'  
-esize '1' -enum '1' -wsp '1' -woff '0' -hsp '1' -hoff '0'  
-dsp '1' -doff '0' -tsp '1' -toff '0' -esp '1' -eoff '0'  
-fgreal '1.0' -bgreal '0.0' -fgimag '0.0' -bgimag '0.0' -type 10

# Glyph 'Shrink'

Glyph:DATAMANIP:kshrink::67:345:305:: -wmag '0.25' -hmag '0.25'

```

# Glyph 'Subtract'
Glyph:DATAMANIP:ksub::71:645:245::
-o /home/galois/albanita/imagen2/var104

# Glyph 'Square'
Glyph:DATAMANIP:ksqr::75:445:305::

# Glyph 'Display Image'
Glyph:ENVISION:putimage::79:765:225:: -use_cmap 1

# Glyph 'Supported Formats'
Glyph:DATAMANIP:kformats::83:785:325:: -ascii

# Glyph 'File Viewer'
Glyph:DESIGN:khelpp::87:885:325::

# Glyph 'Display Image'
Glyph:ENVISION:putimage::91:825:385:: -use_cmap 1

# Glyph 'User defined'
Glyph:DESIGN:user_def::95:545:405:: -o ./stack2

# Glyph 'Append'
Glyph:DATAMANIP:kappend::99:685:385::
-append 1 -mapping 0 -h -o /home/galois/albanita/stack2

# Glyph 'File Viewer'
Glyph:DESIGN:khelpp::103:885:485::

# Glyph 'Supported Formats'
Glyph:DATAMANIP:kformats::107:785:485:: -ascii

#GlyphEnd
#ConnectionBegin

# 'Shrink'
NodeConnection:7:o:11:i:
NodeConnection:7:o:71:i1:

# 'Remove Segments'
NodeConnection:15:o:19:i:

# 'Convert Type'
NodeConnection:19:o:27:i1:

```

NodeConnection:19:o:31:i:  
  
# 'User defined'  
NodeConnection:23:o:15:i:  
NodeConnection:23:o:3:i:  
  
# 'Square'  
NodeConnection:27:o:51:i:  
  
# 'Procedure'  
NodeConnection:31:o:67:i:  
  
# 'Extract'  
NodeConnection:35:o:7:i:  
  
# 'Multiply'  
NodeConnection:39:o:35:i:  
  
# 'Impulse'  
NodeConnection:43:o:39:i2:  
  
# 'Extract'  
NodeConnection:47:o:39:i1:  
  
# 'Pad'  
NodeConnection:51:o:55:i1:  
  
# 'Linear Operator'  
NodeConnection:55:o:47:i:  
  
# 'Divide'  
NodeConnection:59:o:55:i2:  
  
# 'Impulse'  
NodeConnection:63:o:59:i1:  
  
# 'Shrink'  
NodeConnection:67:o:75:i1:  
  
# 'Subtract'  
NodeConnection:71:o:79:i:  
NodeConnection:71:o:83:i:  
NodeConnection:71:o:99:i2:  
  
# 'Square'

NodeConnection:75:o:71:i2:

# 'Supported Formats'

NodeConnection:83:o:87:i:

# 'User defined'

NodeConnection:95:o:99:i1:

# 'Append'

NodeConnection:99:o:91:i:

NodeConnection:99:o:107:i:

# 'Supported Formats'

NodeConnection:107:o:103:i:

#ConnectionEnd

#ExternalConnectionEnd

AnnotationsBegin

AnnotationsEnd

Os programas para agregação nos modos semi-paralelo e semi-sequencial são listados a seguir.

```
% ***** Programa para a Agregacao no Modo Semi-Paralelo *****
%
%                               MATLAB 4.0
%
% ***** dados de entrada:
% matriz - arquivo ascii correspondente a matriz dos valores de
%          atividade de todas as imagens (obtida do programa
%          elaborado no KHOROS 2.0)
% N       - numero total de imagens
% alt    - altura de cada imagem (em pixels)
% larg   - largura de cada imagem (em pixels)
%
% ***** saida:
% y      - serie agregada
%
% *****
% m      - numero de linhas da matriz
% n      - numero de colunas da matriz
```

```

clear;
load -ascii matriz;
m = N*alt/bloc;
n = N*larg/bloc;

l =1;
for k=1:n,
    for j=0:((alt/bloc)-1),
        i=1;
        while i =< m,
            y(l)=matriz(j+i,k);
            i=i+(alt/bloc);
            l=l+1;
        end;
    end
end;

stem(y);

% ***** Programa para a Agregacao no Modo Semi-Sequencial *****
%
%                               MATLAB 4.0
%
% *****  dados de entrada:
% matriz - arquivo ascii correspondente a matriz dos valores de
%          atividade de todas as imagens (obtida do programa
%          elaborado no KHOROS 2.0)
% N       - numero total de imagens
% alt    - altura de cada imagem (em pixels)
% larg   - largura de cada imagem (em pixels)
%
% ***** saida:
% y      - serie agregada

% *****
% m     - numero de linhas da matriz
% n     - numero de colunas da matriz

clear;
load -ascii matriz;

```

```

m = N*alt/bloc;
n = N*larg/bloc;

clear;
load -ascii matriz;

l=1;

    for j=0:((alt/bloc)-1),
        i=1;

            while i =< m,
                for k=1:n,
                    y(l)=matriz(i+j,k);
                l=l+1;
                end;
                i=i+(alt/bloc);
            end;
        end;

stem(y);

```

### B.3. Programas para o Cálculo da Variância das Seqüências de Vídeo

Estes programas foram implementados no ambiente *Khoros 2.1*, através da linguagem *Cantata* e são bastante semelhantes. Como exemplo, a Figura B.3 mostra o programa para a seqüência *mobile* do *rpi.ipl*. Os quadros foram inicialmente “empilhados” de 10 em 10 e depois foi feita a extração da área de  $8 \times 8$  pixels. A seguir, essas áreas foram “empilhadas” e foi calculada a variância de cada uma, obtendo-se, assim, uma série de comprimento igual ao número de quadros da seqüência de vídeo.

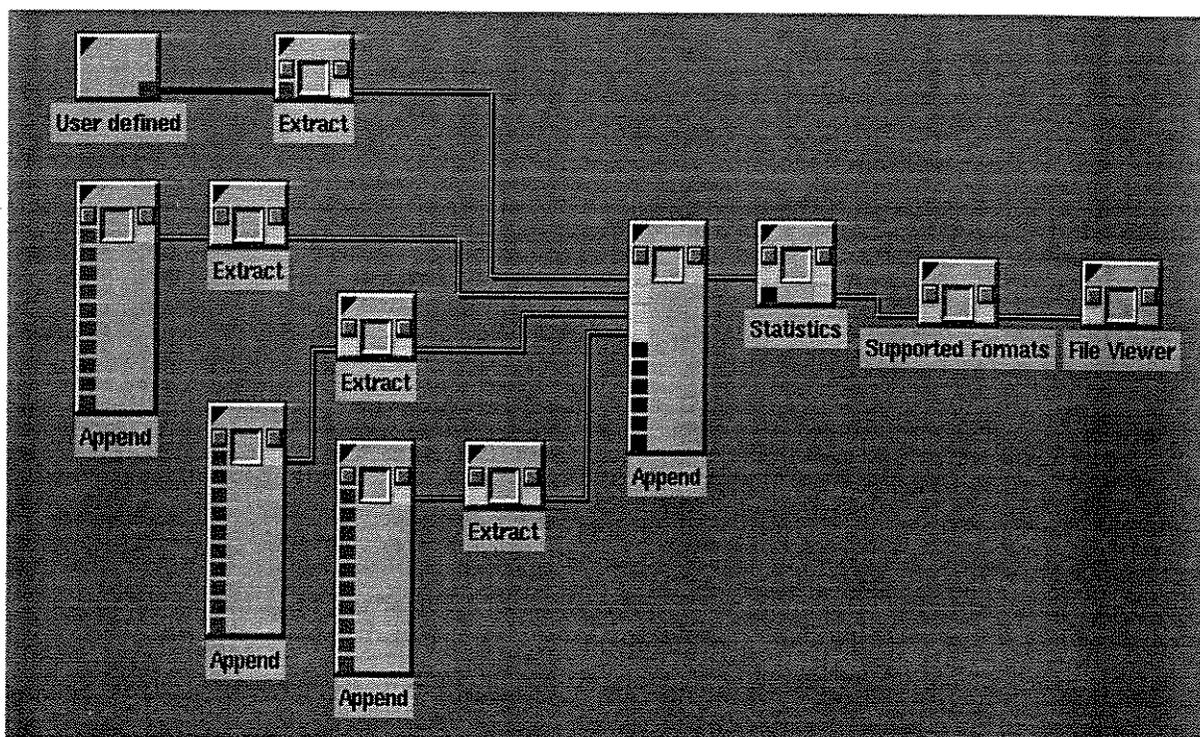


Figura B.3: Programa elaborado na linguagem *Cantata* do Ambiente *Khoros 2.0* para o cálculo da variância de um bloco de  $8 \times 8$  pixels da seqüência *mobile*.

A listagem correspondente à este programa é apresentada a seguir.

```
# Khoros Visual Programming Workspace
#
# cantata workspace file (/.../mob.wk) was created
# on Thu Mar 13 11:11:20 1997
# by user albanita
#

WorkspaceBegin: 2.1

VariablesBegin:

# No variables currently declared. #

VariablesEnd:

# Glyph 'Append'
Glyph:DATAMANIP:kappend::localhost:1:242:142::
-append 1 -mapping 0 -d -i1 /tmp/mobile010 -i2 /tmp/mobile011
-i3 /tmp/mobile012 -i4 /tmp/mobile013 -i5 /tmp/mobile014
```

```

-i6 /tmp/mobile015 -i7 /tmp/mobile016 -i8 /tmp/mobile017
-i9 /tmp/mobile018 -i10 /tmp/mobile019
-o /tmp/seq2

# Glyph 'Append'
Glyph:DATAMANIP:kappend::localhost:3:582:162:: -append 1
-mapping 0 -d -o /home/galois/albanita/mob8x8

# Glyph 'User defined'
Glyph:DESIGN:user_def::localhost:5:242:62:: -o /tmp/seq

# Glyph 'Extract'
Glyph:DATAMANIP:kextract::localhost:7:362:62:: -wsize '8'
-hsize '8' -woff '300' -hoff '500' -subpos 0
-o /home/galois/albanita/seqextr1

# Glyph 'Extract'
Glyph:DATAMANIP:kextract::localhost:9:322:142:: -wsize '8'
-hsize '8' -woff '300' -hoff '500' -subpos 0
-o /home/galois/albanita/seqextr2

# Glyph 'Extract'
Glyph:DATAMANIP:kextract::localhost:11:402:202:: -wsize '8'
-hsize '8' -woff '300' -hoff '500' -subpos 0
-o /home/galois/albanita/seqextr3

# Glyph 'Append'
Glyph:DATAMANIP:kappend::localhost:13:322:262:: -append 1
-mapping 0 -d -i1 /tmp/mobile020 -i2 /tmp/mobile021
-i3 /tmp/mobile022 -i4 /tmp/mobile023 -i5 /tmp/mobile024
-i6 /tmp/mobile025 -i7 /tmp/mobile026 -i8 /tmp/mobile027
-i9 /tmp/mobile028 -i10 /tmp/mobile029 -o /tmp/seq

# Glyph 'Statistics'
Glyph:DATAMANIP:kstats::localhost:15:662:162:: -w -h -var
-f /home/galois/albanita/varmob
-o /home/galois/albanita/varmobin

# Glyph 'Append'
Glyph:DATAMANIP:kappend::localhost:17:402:282::
-append 1 -mapping 0 -d -i1 /tmp/mobile030 -i2 /tmp/mobile031
-i3 /tmp/mobile032 -i4 /tmp/mobile033 -i5 /tmp/mobile034
-i6 /tmp/mobile035 -i7 /tmp/mobile036 -i8 /tmp/mobile037
-i9 /tmp/mobile038 -i10 /tmp/mobile039
-o /tmp/seq

```

```
# Glyph 'Extract'
Glyph:DATAMANIP:kextract::localhost:19:482:282::
-wsize '8' -hsize '8' -woff '300' -hoff '500'
-subpos 0 -o /home/galois/albanita/seqextr4
```

```
# Glyph 'Supported Formats'
Glyph:DATAMANIP:kformats::localhost:21:762:182::
-ascii -o /home/galois/albanita/varmobas
```

```
# Glyph 'File Viewer'
Glyph:DESIGN:khelph::localhost:23:862:182::
```

```
# Connections 'Append'
NodeConnection:data:1:o:9:i
```

```
# Connections 'Append'
NodeConnection:data:3:o:15:i
```

```
# Connections 'User defined'
NodeConnection:data:5:o:7:i
```

```
# Connections 'Extract'
NodeConnection:data:7:o:3:i1
```

```
# Connections 'Extract'
NodeConnection:data:9:o:3:i2
```

```
# Connections 'Extract'
NodeConnection:data:11:o:3:i3
```

```
# Connections 'Append'
NodeConnection:data:13:o:11:i
```

```
# Connections 'Statistics'
NodeConnection:data:15:o:21:i
```

```
# Connections 'Append'
NodeConnection:data:17:o:19:i
```

```
# Connections 'Extract'
NodeConnection:data:19:o:3:i4
```

```
# Connections 'Supported Formats'
NodeConnection:data:21:o:23:i
```

```
# Connections 'File Viewer'
```

```
AnnotationsBegin
```

```
AnnotationsEnd
```

```
WorkspaceEnd: 2.1
```

# Bibliografia

- [1] ACAMPORA, A. S. An Introduction to Broadband Networks. New York: Plenum Press, 1994.
- [2] McDYSAN, D. E. e SPOHN, D. L. ATM: Theory and Application. New York: McGraw-Hill Inc., 1995.
- [3] TANENBAUM, A. S. Computer Networks. Segunda Edição. New Jersey: Prentice Hall, 1989.
- [4] STALLINGS, W. Local and Metropolitan Area Networks. Quarta Edição. Macmillan Publishing, 1993.
- [5] PRYCKER, M. Asynchronous Transfer Mode; Solution for Broadband ISDN. New York: Prentice Hall, 1995.
- [6] MANDELBROT, B. B. The Fractal Geometry of Nature. New York: W. H. Freeman and Co., 1982.
- [7] BARNSLEY, M. F. Fractals Everywhere. USA: Academic Press, 1988.
- [8] FALCONER, K. Fractal Geometry. USA: John Wiley, 1990.
- [9] MANDELBROT, B. B.; VAN NESS J. W. Fractional Brownian Motions, Fractional Noises and Applications. *SIAM Review*, vol. 10, pp 422-437, 1968.
- [10] KOLMOGOROV, A. N. Wiener'sche Spiralen und einige interessante Kurven im Hilbert'schen Raum. *C. R. (Doklady) Acad. Sci. URSS (N.S.)*, 26, pp. 115-118, 1940.
- [11] BARNSLEY, M. F.; ANSON, L. The Fractal Transform. USA: AK Peters Ltd., 1993.
- [12] BARNSLEY, M. F.; HURD, L. P. Fractal Image Compression. USA: AK Peters Ltd., 1993.
- [13] FISHER, Y. (Ed.). Fractal Image Compression. New Jersey: Springer-Verlag, 1995.
- [14] SCHROEDER, M. Fractals, Chaos, Power Laws. New Jersey: W. H. Freeman and Co., 1991.

- [15] PEITGEN, H. O.; SAUPE, D. (Eds.). The Science of Fractal Images. USA: Springer-Verlag, 1988.
- [16] MALASSENET, F. J. Self-affine Signals and Weighted Multiresolution Processes. *Tese de Doutorado*, Georgia Institute of Technology, 1991.
- [17] GRANGER, C. W. J.; Joyeux, R. An introduction to Long-Memory Time Series Models and Fractional Differencing. *Journal of Time Series Analysis*, vol. 1, pp 15-29, 1980.
- [18] HOSKING, J. R. M. Fractional Differencing. *Biometrika*, vol. 68, pp.165-176, 1981.
- [19] BOX, G. E. P.; JENKINS, G. M. Time Series Analysis Forecasting and Control. Segunda Edição. San Francisco: Hodden-Day, 1976.
- [20] MANDELROT, B. B.; WALLIS J. R. Some Long-Run Properties of Geophysical Records. *Water Resources Research*, 5, pp. 321-340, 1969.
- [21] TAQQU, M. S.; LEVY, J. B. Using Renewal Process to Generate Long-Range Dependence and High Variability. In: Dependence in Probability and Statistics, *Progress in Prob. and Stat.*, E. Eberlein e M. S. Taqu (Eds.). Boston: Birkhauser, vol. 11, pp. 73-89, 1986.
- [22] FOX, R.; TAQQU, M. S. Large-Sample Properties of Parameter Estimates for Strongly Dependent Stationary Gaussian Time Series. *Annals of Statistics*, 14, pp. 517-532, 1986.
- [23] DAHLHAUS, R. Efficient Parameter Estimation for Self-Similar Process. *Annals of Statistics*, 17, pp.1749-1766, 1989.
- [24] MANDELROT, B. B. Long-Run Linearity, Locally Gaussian Process, H-Spectra and Infinite Variances. *International Economics Review*, vol.10, no. 1, pp. 82-113, 1969.
- [25] LELAND, W. E. et al. On the Self-Similar Nature of Ethernet Traffic. *Proceedings ACM Sigcomm'93*. San Francisco, set 1993.
- [26] LELAND, W. E. et al. On the Self-Similar Nature of Ethernet Traffic (Extended Version). *IEEE Trans. on Networking*, vol. 2, n. 1, fev 1994.
- [27] TAQQU, M. S.; TEVEROVSKY, V.; WILLINGER, W. Estimators for Long-Range Dependence: an empirical study. *Fractals*, vol. 3, n. 4, pp 785-788, 1995.
- [28] FOWLER, H. J.; LELAND, W. E. Local Area Network Traffic Characteristics, with Implications for Broadband Network Congestion Manager. *IEEE Journal on Selected Areas in Communications*, pp 1139-1149, 1991.
- [29] BERAN, J. et al. Variable-Bit-Rate Video Traffic and Long-Range Dependence. Aceito para publicação no *IEEE Trans. on Communications*, 1992.

- [30] TAQQU, M. S.; TEVEROVSKY, V.; WILLINGER, W. Is Network Traffic Self-Similar or Multifractal. A ser publicado no jornal *Fractals*.
- [31] KOKOSZKA, P. S.; TAQQU, M. S. Parameter Estimation for Infinite Variance Fractional ARIMA. A ser publicado no *The Annals fo Statistics*.
- [32] KOKOSZKA, P. S.; TAQQU, M. S. Fractional ARIMA with Stable Innovations. *Stochastic Processes and Their Applications*, R. J. Adler Ed., n. 60, pp 19-47, dez 1995.
- [33] WILLINGER, W.; TAQQU, M. S.; ERRAMILI, A. A bibliographical Guide to Self-Similar Traffic and Performance Modeling for Modern High-Speed Networks. In: *Sthocastic Networks: Theory and Applications*, KELLY, F. P., ZACHARY, S. e ZIEDINS, I. (Eds.). / Oxford: Claredon Press, pp 339-366, 1996.
- [34] PAXSON, V.; FLOYD, S. Wide Area Traffic: The Failure of Poisson Modeling. *IEEE Transactions on Networking*, vol. 3, n. 3, pp 226-244, jun. 1995.
- [35] CHAMBERS, J. M.; MALLOW, C. L.; STUCK, B. W. A Method for Simulating Stable Random Variables. *Journal of the American Statistical Association*, vol. 71, n. 354, pp 340-344, jun. 1976.
- [36] STALLINGS, W. Viewpoint: Self-similarity upsets data traffic assumptions. *IEEE Spectrum*, jan. 1997.
- [37] LIKHANOV, N.; TSYBAKOV, B.; GEORGANAS, N. D. Analysis of an ATM Buffer with Self-Similar (Fractal) Input Traffic. *Proceedings of IEEE INFOCOM'95*, pp. 985-992, 1995.
- [38] NORROS, I. On the Use of Fractional Brownian Motion in the Theory of Connectionless Networks. *IEEE Journal on Selected Areas in Communications*, vol. 13, n. 6, ago. 1995.
- [39] HURST, H. E. Methods of Using Long-Term Storage in Reservoirs. *Proc. of the Institution of Civil Engineers*, Part I, pp 519-577, 1955.
- [40] ERRAMILI, A.; SINGH, R. P. Application of Deterministic Chaotic Maps to Model Packet Traffic in Broadband Networks. *Proc. 7th ITC Specialists Seminar*. Morristown, NJ, 1990, pp 8.1.1-8.1.3.
- [41] TEWFIK, A. H.; KIM, M. Correlation Structure of the Discrete Wavelet Coefficients of Fractional Brownian Motion. *IEEE Trans. Inf. Theory*, vol. 38, n. 2, mar 1992.
- [42] FLANDRIN, P. Wavelet Analysis and Synthesis of Fractional Brownian Motion. *IEEE Trans. Inf. Theory*, vol. 38, n. 2, mar. 1992.
- [43] PRATT, William K. Digital Image Processing. New York: Wiley, 1991.

- [44] JANIL, Anil K. Fundamentals of Digital Image Processing. Englewood Cliffs: Prentice Hall, 1989.
- [45] LIM, Jae S. Two-dimensional Signal and Image Processing. Englewood Cliffs: Prentice Hall, 1990.
- [46] GONZALEZ, Rafael; WINTZ, Paul. Digital Image Processing. Reading: Addison-Wesley, 1987.
- [47] NUTRAVALI, A.; HASKELL, B. Digital Pictures ; Representation and Compression. New York: Plenum Press, 1988.
- [48] JAYANT, N.; NOLL, P. Digital Coding of Waveforms ; Principles and Applications to Speech and Video. Englewood Cliffs: Prentice Hall, 1984.
- [49] GERSHO, Allen; GRAY, Robert. Vector Quantization and Signal Compression. Norwell: Kluwer Academic Publishers, 1992.
- [50] RABBANI, Majid; JONES, Paul. Digital Image Compression Techniques. Bellingham: SPIE Optical Engineering Press, 1991.
- [51] COVER, Thomas; THOMAS, Joy. Elements of Information Theory. New York: Wiley, 1991.
- [52] BERGER, Toby. Rate Distortion Theory. Englewood Cliffs: Prentice Hall, 1971.
- [53] *Proceedings of the IEEE*. Special Issue on Advances in Image and Video Compression, fev. 1995.
- [54] *Multimedia Systems*. Special Issue on Video Compression, Assoc. for Computing Machinery. Springer International, dez. 1994.
- [55] VAUGHN, Val D.; WILKINSON, Timothy S. System Considerations for Multispectral Image Compression Designs. *IEEE Signal Processing Magazine*, pp. 32-43, jan. 1995.
- [56] SAGHRI, J.; TESCHER, A.; REAGAN, J. Practical Transform Coding of Multispectral Imagery. *IEEE Signal Processing Magazine*, pp. 19-31, jan. 1995.
- [57] LARGE, David. Creating a Network for Interactivity. *IEEE Spectrum*, pp. 58-63, abr. 1995.
- [58] HAMALAINEN, Jukka. Video recording goes digital. *IEEE Spectrum*, pp. 76-79, abr. 1995.
- [59] ANG, Peng; RUERTZ, Peter; AULD, David. Video compression makes big gains. *IEEE Spectrum*, pp. 16-19, out. 1991.
- [60] Le GALL, D. MPEG: A Video Compression Standard for Multimedia Applications. *Communications of the ACM*, vol. 34, n. 4, pp. 305-313, abr. 1991.

- [61] PANCHHA, P.; el ZARKI, M. Bandwidth-Allocation Schemes for Variable-Bit-Rate MPEG Sources in ATM Networks. *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 3, n. 3, jun. 1993.
- [62] DING, W.; LIU, B. Rate Control of MPEG Video Coding and Recording by Rate-Quantization Modeling. *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, n. 1, fev 1996.
- [63] KHADEMI, M.; PAOLONI, F. J. Performance and Statistical Analysis of Two-layer MPEG Coding Algorithms for ATM Networks. *IEEE Proceedings on Communications*, vol. 143, n. 2, jun. 1996.