

**URIEL DA VEIGA SEGALL**

**ANÁLISE E MODELAGEM DE TRÁFEGO DE  
VÍDEO HIERÁRQUICO**

Dissertação apresentada ao Programa de Pós-Graduação em Informática Aplicada, da Pontifícia Universidade Católica do Paraná, como requisito parcial para obtenção do título de Mestre em Informática Aplicada.

**CURITIBA**

**2005**



**URIEL DA VEIGA SEGALL**

**ANÁLISE E MODELAGEM DE TRÁFEGO DE  
VÍDEO HIERÁRQUICO**

Dissertação apresentada ao Programa de Pós-Graduação em Informática Aplicada da Pontifícia Universidade Católica do Paraná como requisito parcial para obtenção do título de Mestre em Informática Aplicada.

Área de Concentração: *Sistemas Distribuídos*

Orientador: Prof. Dr. Marcelo Eduardo Pellenz

**CURITIBA**

**2005**

SEGALL, Uriel da Veiga

Análise e Modelagem de Tráfego de Vídeo Hierárquico. Curitiba, 2005. 143p.

Dissertação – Pontifícia Universidade Católica do Paraná. Programa de Pós-Graduação em Informática Aplicada.

1. Modelagem de Tráfego de Vídeo; 2. Autocorrelação; 3. Auto-Similaridade; 4. Tráfego de Vídeo Hierárquico. I. Pontifícia Universidade Católica do Paraná. Centro de Ciências Exatas e de Tecnologia. Programa de Pós-Graduação em Informática Aplicada II-t





Dedico este trabalho à minha mãe.





## **Agradecimentos**

Agradeço a todas as pessoas que se envolveram no desenvolvimento desse trabalho, sempre contribuindo e incentivando para a verificação de melhores resultados. Agradeço em especial à minha mãe que acredita e possibilita o meu desenvolvimento como pessoa e profissional. Agradeço ao Prof. Dr. Marcelo Pellenz pelo apoio e suporte durante todo o desenvolvimento desse trabalho.



# Sumário

<b>Agradecimentos</b>	<b>vii</b>
<b>Sumário</b>	<b>ix</b>
<b>Lista de Figuras</b>	<b>xiii</b>
<b>Lista de Tabelas</b>	<b>xvii</b>
<b>Lista de Símbolos</b>	<b>xviii</b>
<b>Lista de Abreviaturas</b>	<b>xix</b>
<b>Resumo</b>	<b>xxi</b>
<b>Abstract</b>	<b>xxiii</b>
<b>Capítulo 1</b>	
<b>Introdução</b>	<b>1</b>
1.1. Motivação	1
1.2. Proposta	4
1.3. Organização	5
<b>Capítulo 2</b>	
<b>Fundamentos Matemáticos</b>	<b>7</b>
2.1. Conceitos Básicos	7
2.2. Variáveis Aleatórias	9
2.2.1. Variáveis Aleatórias Discretas	10
2.2.2. Variáveis Aleatórias Contínuas	11
2.2.3. Médias	12
2.2.4. Momentos	14
2.2.5. Variância	14
2.3. Processos Aleatórios	15
2.4. Autocorrelação	17
2.5. Autocovariância	18

2.6. Processos de Markov	19
2.6.1. Cadeias de Markov	20
2.7. Processos Auto-Similares	23
2.7.1. Definição	24
2.7.2. Propriedades	26
2.7.3. Estimativa do Coeficiente de Auto-Similaridade	27
2.8. Conclusões	30
<b>Capítulo 3</b>	
<b>Métodos de Compressão de Vídeo</b>	<b>31</b>
3.1. Análise do padrão de Compressão MPEG	31
3.2. Estrutura de Fluxo MPEG	34
3.3. Estrutura e Codificação do GOP	35
3.4. Conclusões	37
<b>Capítulo 4</b>	
<b>Análise Estatística de Tráfego de Vídeo</b>	<b>39</b>
4.1. Arquivos de Tráfego de Vídeo	39
4.2. Análise Estatística dos Arquivos de Vídeo	43
4.3. Análise da Auto-Similaridade	65
4.4. Conclusões	73
<b>Capítulo 5</b>	
<b>Modelagem de Tráfego de Vídeo</b>	<b>75</b>
5.1. Métodos de Modelagem	75
5.2. Modelagem de Processos Auto-Similares	77
5.2.1. F-ARIMA – Algoritmo da Modelagem de Tráfego	77
5.2.2. F-ARIMA – Adaptação Combinatória	80
5.2.3. F-ARIMA – Adaptação Hosking	82
5.2.4. F-ARIMA – Diferença Fracional	84
5.3. Agrupamento de Quadros	85
5.4. Conclusões	89

<b>Capítulo 6</b>	
<b>Simulações de Tráfego de Vídeo Hierárquico</b>	<b>91</b>
6.1. Avaliação dos Algoritmos de Simulação	92
6.2. Geração de Simulação	97
6.3. Modelagem Hierárquica	107
6.4. Conclusões	111
<b>Capítulo 7</b>	
<b>Conclusão</b>	<b>113</b>
<b>Referências Bibliográficas</b>	<b>115</b>



## Lista de Figuras

Figura 3.1	Reordenação dos Coeficientes DCT.....	33
Figura 3.2	Descrição da Ordem das Informações em um Fluxo de Vídeo MPEG...	34
Figura 3.3	Descrição da Ordem das Informações em um Fluxo de Áudio MPEG...	34
Figura 3.4	Relacionamento de Quadros IPB.....	36
Figura 4.1	Reordenação dos Coeficientes DCT.....	43
Figura 4.2	Tamanhos dos quadros em média resolução.....	44
Figura 4.3	Tamanhos dos quadros em baixa resolução.....	44
Figura 4.4	Distribuição de frequência para um vídeo em baixa resolução.....	45
Figura 4.5	Distribuição de frequência para um vídeo em média resolução.....	47
Figura 4.6	Distribuição de frequência para um vídeo em alta resolução.....	49
Figura 4.7	Função de densidade de probabilidade para vídeo em alta resolução.....	51
Figura 4.8	Função de distribuição cumulativa para vídeo em alta resolução.....	52
Figura 4.9	Função de densidade de probabilidade para vídeo em média resolução.	52
Figura 4.10	Função de distribuição cumulativa para vídeo, em média resolução.....	53
Figura 4.11	Função de densidade de probabilidade para vídeo em baixa resolução..	53
Figura 4.12	Função de distribuição cumulativa para vídeo em baixa resolução.....	54
Figura 4.13	Autocorrelação para um vídeo de baixa resolução.....	56
Figura 4.14	Autocorrelação para um vídeo de média resolução.....	57
Figura 4.15	Autocorrelação para um vídeo de alta resolução.....	58
Figura 4.16	Autocorrelação do vídeo <i>Jurassic Park</i> , em baixa resolução, por tipo de quadro.....	59
Figura 4.17	Autocorrelação do vídeo <i>Alladin</i> , em média resolução, por tipo de quadro.....	60
Figura 4.18	Autocorrelação do vídeo <i>Star Wars IV</i> , em alta resolução, por tipo de quadro.....	61
Figura 4.19	Autocorrelação de quadros do tipo I nos arquivos de alta, média e	

	baixa resolução.....	62
Figura 4.20	Autocorrelação de quadros do tipo P nos arquivos de alta, média e baixa resolução.....	63
Figura 4.21	Autocorrelação de quadros do tipo B nos arquivos de alta, média e baixa resolução.....	64
Figura 4.22	Estimativa de Hurst em função da $\log(R/S)$ e $\log(k)$ , de um vídeo em alta resolução.....	65
Figura 4.23	Estimativa de Hurst em função da $\log(R/S)$ e $\log(k)$ para os quadros I de um vídeo em alta resolução.....	66
Figura 4.24	Estimativa de Hurst em função da $\log(R/S)$ e $\log(k)$ para os quadros P de um vídeo em alta resolução.....	67
Figura 4.25	Estimativa de Hurst em função da $\log(R/S)$ e $\log(k)$ para os quadros B de um vídeo em alta resolução.....	67
Figura 4.26	Valores de Hurst para 500 GOPs.....	69
Figura 4.27	Valores de Hurst para quadros do tipo I em alta resolução.....	70
Figura 4.28	Valores de Hurst para quadros do tipo P em alta resolução.....	71
Figura 4.29	Valores de Hurst para quadros do tipo B em alta resolução.....	71
Figura 5.1	Autocorrelação real em comparação com os valores das funções de autocorrelação teóricas.....	76
Figura 5.2	Complexidade do algoritmo F-ARIMA – Recursividade.....	79
Figura 5.3	Complexidade do algoritmo F-ARIMA – Adaptação Combinatória.....	82
Figura 5.4	Complexidade do algoritmo F-ARIMA – Adaptação Hosking.....	83
Figura 5.5	Complexidade do algoritmo F-ARIMA – Diferença Fracional.....	85
Figura 6.1	Autocorrelação de Simulação em comparação à autocorrelação teórica.....	93
Figura 6.2	Autocorrelação de Simulação Hosking e Combinatória em comparação à autocorrelação teórica.....	94
Figura 6.3	Autocorrelação de Simulação de Hosking em comparação com a teórica.....	95
Figura 6.4	Comparação da Autocorrelação Teórica e de Simulação por diferença fracional.....	96
Figura 6.5	Autocorrelação da simulação de quadros I em alta resolução.....	98



Figura 6.6	Autocorrelação da simulação de quadros P em alta resolução.....	99
Figura 6.7	Autocorrelação da simulação de quadros B em alta resolução.....	100
Figura 6.8	Autocorrelação da simulação de quadros I em média resolução.....	101
Figura 6.9	Autocorrelação da simulação de quadros P em média resolução.....	102
Figura 6.10	Autocorrelação da simulação de quadros B em média resolução.....	103
Figura 6.11	Autocorrelação da simulação de quadros I em baixa resolução.....	104
Figura 6.12	Autocorrelação da simulação de quadros P em baixa resolução.....	105
Figura 6.13	Autocorrelação da simulação de quadros B em baixa resolução.....	106
Figura 6.14	Modelo para geração de tráfego de vídeo hierárquico.....	108
Figura 6.15	Autocorrelação geral da simulação para alta resolução.....	109
Figura 6.16	Autocorrelação geral da simulação para média resolução.....	110
Figura 6.17	Autocorrelação geral da simulação para baixa resolução.....	110



## Lista de Tabelas

Tabela 3.1	Características de vídeos em diferentes resoluções.....	33
Tabela 4.1	Exemplo de trecho de arquivo da estrutura de dados dos quadros.....	41
Tabela 4.2	Trecho do arquivo de dados, ordenado pela ordem de apresentação do quadro.....	42
Tabela 4.3	Percentagem de agrupamento por classe de frequência em baixa resolução.....	46
Tabela 4.4	Percentagem de agrupamento por classe de frequência em média resolução. ....	48
Tabela 4.5	Percentagem de agrupamento por classe de frequência em alta resolução. ....	50
Tabela 4.6	Estatísticas dos quadros obtidas para os vídeos em baixa, média e alta resolução. ....	55
Tabela 4.7	Estatísticas por tipo de quadro obtido para os arquivos em baixa, média e alta resolução.....	55
Tabela 4.8	Média dos valores de Estimativa de Hurst.....	68
Tabela 4.9	Estimativa de Hurst, para todos os quadros em diferentes resoluções, para k intervalos.....	72
Tabela 5.1	Relação de quadros para o modelo hierárquico.....	86
Tabela 5.2	Características das complexidades dos algoritmos.....	89
Tabela 6.1	Parâmetros definidos para o modelo de simulação de tráfego de vídeo....	108

## Lista de Símbolos

$p()$	Probabilidade
$F()$	Função de distribuição cumulativa
$f()$	Função de densidade de probabilidade
$t$	Instante de tempo
$k$	Intervalo
$T$	Tempo total de observação
$r()$	Função de autocorrelação
$v()$	Função de autocovariância
$X()$	Processo aleatório
$i$	Índice de iteração
$N$	Número total de amostras
$X$	Variável aleatória
$P$	Distribuição de probabilidade
$\sigma^2$	Variância
$\sigma$	Desvio Padrão
$\mu$	Média

## Lista de Abreviaturas

ACF	<i>Autocorrelation Function</i>
FDC	Função de distribuição cumulativa
FDP	Função de densidade de probabilidade
FIR	<i>Finite Impulse Response</i>
F-ARIMA	<i>Fractional Autoregressive Integrated Moving Average</i>
SRA	<i>Successive Random Addition</i>
GOP	<i>Group of Pictures</i>
RMD	<i>Random Midpoint Displacement</i>
LRD	<i>Long-range Dependence</i>
SRD	<i>Short-range Dependence</i>
DCT	<i>Discrete Cosine Transform</i>
MPEG	<i>Moving Picture Experts Group</i>
VBR	<i>Video Bit Rate</i>
R/S	<i>Rescaled Adjusted Range Statistic</i>
MTU	<i>Maximum Transmission Unit</i>



## Resumo

Este trabalho tem como finalidade apresentar uma modelagem de tráfego de vídeo hierárquico, em cujas simulações foram avaliadas as características de processos aleatórios e de processos auto-similares. Foram identificadas informações para construção do modelo proposto através de análises estatísticas verificadas em arquivos de vídeo real. As análises estatísticas apresentam informações relacionadas às definições e características de processos aleatórios e de processos auto-similares. Propostas do algoritmo F-ARIMA foram avaliadas e utilizadas para construção do modelo de simulação de tráfego de vídeo. Propõe-se um método de agrupamento da estrutura MPEG para geração do modelo hierárquico. O modelo de simulação proposto aproxima os resultados estatísticos de autocorrelação aos resultados obtidos através de análise de vídeo real. O modelo de tráfego de vídeo hierárquico possibilita gerar tráfego de vídeo para simulações que podem ser utilizadas em análise em redes de computadores e de telecomunicações.

**Palavras-Chave:** 1. Modelagem de Tráfego de Vídeo; 2. Autocorrelação; 3. Auto-Similaridade; 4. Tráfego de Vídeo Hierárquico.





# Abstract

This document has the main goal to introduce a hierarchical video traffic modelling. The modelling results had been analyzed in comparison to the random and self-similar processes. Statistical information data had been collected from video trace files analyses. The statistical analyses introduce information related to random process and self-similar processes characteristics and definitions. Implementations from F-ARIMA algorithm had been discussed and used to implement the simulation model for hierarchical traffic video. There is a discussion for hierarchical simulation grouping for MPEG structure. The simulation model approximates the autocorrelation results collected from the real video traffic files. The hierarchical video model can be used to generate traffic simulation that is for computer or telecommunications networks evaluations.

**Keywords:** 1. Video Traffic Modelling; 2. Simulation; 3. Self-Similarity Process; 4. Hierarchical Video Traffic.



# Capítulo 1

## Introdução

Aplicações multimídia em tempo real estão sendo intensamente desenvolvidas em diversas plataformas e topologias. Atualmente são encontradas aplicações multimídia em computadores pessoais e em pequenos dispositivos de comunicação. Esse mercado está em constante expansão. As estruturas para Internet e Intranet estão utilizando cada vez mais características de serviços de rede integrados, onde aplicações de características diferenciadas são executadas usando um mesmo modelo de serviços e estrutura de rede.

No entanto, a estrutura de comunicação comum oferece riscos à qualidade do conteúdo multimídia em função das políticas definidas para transmissão e descarte de pacotes, e da complexidade na formatação dos dados. A infra-estrutura de redes necessita constantemente avaliar mecanismos para aperfeiçoar e garantir o desempenho dessas aplicações. A limitação física em caso de sobrecarga nos roteadores para direcionamento e retransmissão de pacotes de comunicação gera perda de informações para as aplicações de tempo real, prejudicando a qualidade do conteúdo apresentado.

A estruturação do conteúdo multimídia apresenta quadros de diferentes tipos e prioridades para o processo de decodificação. Essa situação é particularmente crítica quando a informação é transmitida sem oferecer nenhum controle de transporte e entrega, com o objetivo de minimizar o atraso de transporte na rede. Essa é uma característica comum no que se refere às aplicações multimídia.

Tipicamente, todos os pacotes possuem a mesma probabilidade de descarte, característica crítica em situações de sobrecarga na rede, especificamente para as aplicações multimídia. Sendo assim, informações essenciais podem ser perdidas, degradando o procedimento de decodificação dos dados e conseqüentemente na apresentação do vídeo

transmitido. Essa situação ocasiona a quebra da seqüência das informações multimídia e de sincronismo temporal entre os quadros, degradando a qualidade do vídeo apresentado.

Arquivos de conteúdo multimídia em geral possuem um volume elevado de informações para transporte. Algoritmos de compressão são utilizados para reduzir o tamanho dos dados, facilitando o processo de codificação dos quadros multimídia e minimizando o custo de transmissão. Em contrapartida, diminui-se a resolução dos quadros de vídeo devido à perda de informações após o procedimento de compressão.

A priorização das informações, inserida na política de descarte de pacotes, reduz os impactos nos roteadores e conseqüentemente na qualidade do vídeo transmitido em casos de sobrecarga de tráfego. Os quadros de conteúdo menos significativo ao vídeo podem ser descartados, ao invés de quadros de maior relevância e com mais informações. No entanto deve-se considerar que a perda de pacotes em redes congestionadas, não corresponde à perda de quadros na mesma proporção, e isso é observado através de uma acentuada degradação na qualidade do vídeo recebido.

### **1.1. Motivação**

As redes de comutação de pacotes, assim como os seus componentes, são amplamente afetadas quando existem processos transmitindo uma grande quantidade de informações continuamente. Essa característica é comum em aplicações de tempo real, onde os dados são transmitidos a todo tempo, para vários pontos na rede consumidores de tal informação. As aplicações multimídia são aplicações de tempo real e por isso apresentam uma grande geração de informações e alto consumo do canal de comunicação.

Outra característica das aplicações multimídia é definida pela seqüencialidade dos quadros para a devida apresentação do vídeo ou somente do som. As imagens devem ser devidamente organizadas, em função do intervalo de tempo a definir a seqüência de apresentação. Devido a essa característica, caso um quadro seja entregue atrasado, em relação ao tempo de apresentação da imagem, esse quadro já não possui qualquer utilidade de tratamento pelo processo receptor.

Padrões definidos para a codificação de vídeo compactam as imagens, atribuindo uma menor compressão nas regiões onde ocorrem menores variações nos quadros, em comparação com a variação da imagem nos quadros vizinhos. Mesmo compactadas, as

imagens apresentam grande volume de informações, especialmente quando se trabalha com imagens em alta resolução.

Em especial, as aplicações multimídia consomem uma grande quantidade de banda de rede, sobrecarregando os roteadores e congestionando a rede em função da transmissão constante do fluxo informações. A saturação na rede causada pelo tráfego de vídeo pode causar impactos na qualidade de serviço de outros fluxos da rede, na qualidade da imagem entregue e também para as demais aplicações que compartilham o mesmo canal de comunicação.

O grande volume de informação transmitido afeta em primeira instância aos roteadores, devido à necessidade de definir e aplicar políticas de descarte de pacotes, em caso de congestionamento nas filas de armazenamento temporário de informação. Quando a política de descarte de pacotes é definida pela ordem de chegada na fila, então algumas informações relevantes à decodificação do vídeo são perdidas.

Essa política é essencialmente crítica, quando se considera que tráfego de vídeo é transmitido sob o protocolo UDP, o qual não oferece controle de transmissão na rede. Em contra partida, o protocolo UDP oferece melhor desempenho ao se considerar a transmissão em função da velocidade e do grande volume de dados. Essas características descartam a possibilidade da utilização de controle de transmissão na camada de rede oferecido pelo TCP, de acordo com as definições do modelo OSI. Sendo assim, a rede não oferece qualquer controle na transmissão de informação de vídeo.

De forma a contornar essas questões, soluções em tempo de codificação e soluções na camada de comunicação são objetos de estudo com o objetivo de reduzir esses problemas. As soluções para transmissão de vídeo propõem mecanismos de proteção na rede, como por exemplo: avaliação no gerenciamento de filas nos roteadores e diferenciação de prioridade de pacotes [MAR03], transmissão unificada [LEE99] e retransmissão confiável de mídia contínua [ANS02]. Soluções para tráfego de vídeo propõem mecanismos de proteção em tempo de codificação, como por exemplo: redundância de dados [BOL96], interpolação temporal e espacial [SEE04], e codificação hierárquica [SCH01].

A codificação hierárquica trabalha com a separação do conteúdo a ser apresentado em camadas com diferentes níveis de resolução e compressão. Essas camadas são divididas por tipo de quadros de acordo com o padrão MPEG. O fluxo de vídeo é transmitido para todos os receptores. No entanto, para aqueles que possuem melhor banda de rede disponível, são

considerados os quadros com melhor resolução e com maior detalhamento da imagem. Quadros com melhores e diferentes níveis de resolução e compressão são distribuídos para os receptores com banda de rede disponível para transmissão.

## 1.2. Proposta

O modelo de tráfego de vídeo hierárquico proposto aproxima os valores estatísticos de vídeo real, para avaliar as características das redes através da simulação da transmissão de vídeo. Por meio das simulações é possível identificar os pontos de aperfeiçoamento na configuração e na estrutura da rede, através dos impactos causados por uma transmissão de vídeo. As simulações têm como finalidade reduzir a complexidade para preparação da estrutura física de geração de vídeo e análise do tráfego de informações.

Os componentes da rede que se destacam na avaliação da transmissão do vídeo são: as considerações de uma determinada topologia, da banda de rede, a taxa de transmissão, de mecanismos de controle e políticas de descarte de pacotes. Esses componentes, quando afetados, degradam a qualidade de serviço no canal de comunicação.

Os modelos de simulação tratam por considerar a taxa de tamanho dos quadros no vídeo, de forma a independer do padrão de codificação utilizado. No entanto, nesse trabalho foram verificadas as considerações do padrão de codificação MPEG. O tamanho dos quadros define um processo aleatório com características estatísticas e características de auto-similaridade. A partir desses valores define-se um coeficiente de auto-similaridade utilizado como parâmetro na geração da simulação.

As informações estatísticas identificadas na análise de vídeo real apresentam informações referentes ao tipo de resolução do vídeo e também aos tipos de quadros definidos no padrão MPEG. Os valores da média e variância dos vídeos analisados parâmetros de simulação, assim como o coeficiente de auto-similaridade. O modelo de tráfego de vídeo hierárquico basicamente requer as seguintes características para definição os valores de simulação:

- Tamanho do agrupamento de quadros do padrão MPEG;
- Coeficiente de auto-similaridade por tipo de quadro;
- Valores estatísticos do vídeo real analisado;

Essas características são parâmetros de entrada para o modelo de simulação, onde são gerados quadros com as devidas definições estatísticas encontradas no vídeo real. O coeficiente de auto-similaridade define o grau de aproximação dos quadros gerados no modelo de simulação em relação aos valores estatísticos de quadros reais. Uma vez analisados os arquivos de vídeo real e com os valores gerados através do modelo, podem se realizar comparações desses resultados, em função dos gráficos de autocorrelação gerados pela análise de auto-similaridade.

As análises de simulação de tráfego de vídeo nas redes podem utilizar o modelo de simulação de vídeo hierárquico. O modelo de simulação provê informações relacionadas a vídeos reais, e que em simulações de rede, identificam-se as fragilidades de acordo com as definições da estrutura de rede avaliada. Dessa forma, as propostas de controle de tráfego e de transmissão de aplicações multimídia são medidas e comparadas, com a finalidade de definir estratégias de estruturação e de qualidade de serviço na configuração da rede.

### **1.3. Organização**

Esse trabalho está organizado nas seções de fundamentação estatística e matemática, métodos de codificação de vídeo, análise estatística de arquivos de vídeo, método de simulação, resultados das simulações e por último a conclusão do trabalho.

Na Seção 2 serão apresentados os conceitos estatísticos requeridos para avaliação de processos aleatórios auto-similares. Também será tratado um método de estimativa do coeficiente de Hurst, a partir de um vídeo real e utilizado na análise do resultado do modelo de simulação.

Na Seção 3 serão apresentados os conceitos e definições referentes ao padrão de codificação MPEG. São apresentadas a estrutura e as características do processo de codificação MPEG.

Na Seção 4 serão apresentados os resultados da análise estatística, realizada em um arquivo de tráfego de vídeo real. Será apresentado o procedimento utilizado para obtenção dos arquivos de tamanhos dos quadros, a análise estatística dos vídeos e a análise de auto-similaridade dos vídeos analisados.

Na Seção 5 será apresentado o método de simulação para modelagem de tráfego de vídeo. Nessa seção serão detalhadas as informações referentes à escolha da função teórica,

referente ao processo de simulação, propostas dos algoritmos de simulação e as características para agrupamento de tipos de quadros MPEG.

Na Seção 6 serão apresentados os resultados das simulações por resolução e por tipo de quadro MPEG, utilizando o método proposto na Seção 5. Serão apresentados também os resultados da modelagem hierárquica em comparação com os tipos de resolução de vídeo. Por último será apresentada a conclusão do trabalho e os resultados obtidos no modelo de simulação de tráfego de vídeo hierárquico.



## Capítulo 2

### Fundamentos Matemáticos

As modelagens de tráfego de vídeo estão fundamentadas em conhecimentos de Estatística e Matemática. Essas definições serão necessárias para se compreender o comportamento dos vídeos e modelos de simulação avaliados. Sendo assim, esse capítulo apresenta uma revisão nos conceitos estatísticos utilizados nos próximos capítulos.

#### 2.1. Conceitos Básicos

De acordo com as definições apresentadas em [SHA74] serão expostos alguns conceitos básicos, frequentemente utilizados na estatística. A **população** é o conjunto total dos valores possíveis e existentes para se determinar uma coleção total de objetos ou das medidas em consideração. A população pode ser finita ou infinita. A **amostra** é um subconjunto de valores obtidos da população. Comumente estimam-se as características da população, determinando-se as propriedades estatísticas da amostra.

O conjunto de dados analisado isoladamente não representa qualquer informação. Uma das primeiras análises estatísticas é a da **distribuição de freqüências** dos valores trabalhados. A distribuição de freqüências separa os dados em classes e apresenta o número de ocorrências de cada classe, ou freqüência de classe [SHA74]. Os limites de classe referem-se aos valores máximos e mínimos da classe. O intervalo da classe diz respeito à diferença entre os limites da classe mínima e máxima para uma classe particular.

Pela distribuição de freqüência é possível apresentar o **histograma**, gráfico que se baseia no número de ocorrências de valores para uma classe de freqüência. Nesse processo ocorre um arredondamento grosseiro dos valores, para formação da classe correspondente. Por esse gráfico se visualiza o conjunto de valores mais representativo da amostra, e

conseqüentemente, é possível avaliar a classe de maior percentual de repetição de um conjunto de variáveis aleatórias.

Um meio de descrever uma massa de dados está em expressar as medidas de **tendência central**. As medidas de tendência central mais comumente utilizadas são a média, a mediana e a moda. A **média** (2.1), ou valor esperado ou expectativa [SHA74], representa o valor da somatória de todos os valores de uma amostra dividido pela quantidade de amostras.

$$\mu = \frac{\sum_{i=1}^N x_i}{N} \quad (2.1)$$

Quando o conjunto de dados é ordenado em ordem crescente e seqüencialmente, em razão do seu valor, a próxima medida de tendência central é a **mediana**. O valor da mediana se encontra exatamente no meio desse novo conjunto de dados. Assume-se essa condição caso o número de amostras seja ímpar, onde a mediana seria o dado contido na posição  $i = N/2$ . Caso o número de amostras seja par, então não há um valor exatamente definido no meio desse conjunto. Portanto, a mediana é a média dos dois valores do meio.

A última medida de tendência central a ser apresentada é a **moda**. A moda é o valor que ocorre com maior freqüência em um conjunto de dados. A moda não é necessariamente única e nem sempre existe. Um conjunto de dados pode ter mais de uma moda ou não ter nenhuma.

Em geral a média, a mediana e a moda não possuem o mesmo valor. Se a distribuição de freqüência for simétrica e decrescer continuamente, a partir do centro, como por exemplo, a distribuição normal, então as medidas de tendência central serão mantidas.

A **distribuição normal**, ou também conhecida como distribuição de Gauss, é contínua e simétrica em torno da média e estende-se de menos infinito ao mais infinito. A função de densidade de probabilidade da distribuição normal é definida pela equação (2.2).

$$f(x) = \frac{1}{\sqrt{2\pi\sigma}} \cdot e^{-(x-\mu)^2/2\sigma^2}, \quad -\infty < x < \infty \quad (2.2)$$

Uma outra maneira de se descrever certa massa de dados consiste em expressar as medidas de **dispersão**, que são: a amplitude, o desvio-padrão e a variância. A **amplitude** é a simples diferença entre o maior e o menor valor do conjunto de dados. Uma desvantagem da amplitude é que não se leva em consideração a aglomeração dos dados.

A medida de dispersão mais significativa é a **variância**  $\sigma^2$ , ou a sua raiz quadrada, o **desvio-padrão**  $\sigma$ . A variância (2.3) mede basicamente os desvios da média, elevados ao quadrado. Em outras palavras, a variância é a média dos quadrados das diferenças dos valores, em relação à média.

$$\sigma^2 = \sum_{i=1}^N (x_i - \mu)^2 \quad (2.3)$$

## 2.2. Variáveis Aleatórias

**Variáveis aleatórias**, também conhecidas como variáveis estocásticas ou randômicas [SHA74] são valores numéricos que assumem valores inesperados e diferentes, gerados como resultados de um determinado experimento aleatório. Quando uma variável possui resultados que tendem a variar de uma observação para outra, em razão de fatores relacionados ao acaso, define-se variável aleatória.

As variáveis aleatórias podem resultar de valores gerados a partir de funções, às quais se aplicam regras de cálculo previamente definidas. Qualquer função de uma variável aleatória é também uma variável aleatória. Essas funções também podem somente aplicar as regras de cálculo, em que se atribui um valor definido, baseado no resultado final de um experimento.

A distribuição de probabilidade  $P$  é a representação de uma lista de probabilidades (2.4), associadas a cada variável aleatória encontrada em um experimento.

$$p(x) = P\{X = x\} \quad (2.4)$$

A soma dessas probabilidades (2.5) sempre assume um valor unitário.

$$\sum_{i=1}^{\infty} p(x_i) = 1 \quad (2.5)$$

Os valores possíveis para atribuição, em uma variável aleatória, estão compreendidos em um conjunto de todos os números reais possíveis. A **função de distribuição cumulativa**  $F(x)$ , ou também função de distribuição [ROS97], de uma determinada variável aleatória  $X$  pertencente ao conjunto dos números reais  $x$ , é definido pela fórmula (2.6):

$$F(x) = P\{X \leq x\}, \quad -\infty < x < \infty \quad (2.6)$$

### 2.2.1. Variáveis Aleatórias Discretas

Se os valores são elementos separados e distintos, então se classificam os dados como **discretos**. O processo de contagem caracteriza um exemplo de dados discretos. Por exemplo, um número de homens necessários para se executar uma determinada tarefa pode ser diferente a cada experimento.

Uma variável aleatória é dita discreta, quando é definida em um conjunto de valores finitos ou conjunto de valores contáveis. Os valores discretos são variáveis aleatórias com probabilidades determinadas e estão distribuídas dentro de um intervalo definido. Uma aplicação de variáveis discretas é um valor gerado em um determinado instante de tempo, por um processo de geração de valores aleatórios.

A **função de densidade de probabilidade**  $f(x)$  é uma função pela qual é possível verificar a probabilidade de uma variável aleatória discreta em um determinado intervalo. Para uma variável aleatória discreta  $X$  a definição de  $f(x)$  (2.7) é dada pela função de distribuição cumulativa  $F(x)$ , como segue:

$$f(x) = \frac{d}{dx} F(x) \quad (2.7)$$

A somatória dos valores da função de densidade de probabilidade  $f(x_i)$  para os valores  $x_i$  é igual a 1 (2.8), onde  $i$  varia de 1 a  $\infty$ .

$$\sum_{i=1}^{\infty} f(x_i) = 1 \quad (2.8)$$

Outra condição (2.9) é que a probabilidade nunca será negativa para qualquer valor de  $x$  definido no conjunto dos números reais.

$$f(x) > 0, \quad x \in \mathfrak{R} \quad (2.9)$$

### 2.2.2. Variáveis Aleatórias Contínuas

Os dados **contínuos** podem teoricamente assumir qualquer valor entre dois valores limites, definindo assim um **intervalo** de valores possíveis. A maioria dos dados obtidos em medições é contínua. Os dados contínuos podem parecer discretos, devido ao término do processo de medição. No entanto, esses valores perpetuar-se-iam infinitamente caso o processo de medição não fosse interrompido.

São variáveis aleatórias contínuas quando existe uma função não-negativa, definida para todos os números reais. Variáveis aleatórias contínuas podem assumir qualquer valor, durante a execução do processo de geração de valores. Uma aplicação de variáveis contínuas é um conjunto de valores gerados em um determinado intervalo de tempo, por um processo de geração de valores aleatórios.

A função densidade de probabilidade para variáveis aleatórias contínuas  $f(x)$  também é uma função que determina a probabilidade de uma variável aleatória em um determinado intervalo. Para uma variável aleatória contínua  $X$  a definição de  $f(x)$  (2.10) é dado pela função de distribuição cumulativa  $F(x)$ , como segue:

$$\int_a^b f(x) dx = F(a) - F(b) = P(a < X < b) \quad (2.10)$$

As condições definidas em (2.8) e (2.9) também devem ser satisfeitas para as variáveis aleatórias contínuas e a respectiva função densidade de probabilidade.

### 2.2.3. Médias

Para as estatísticas das variáveis aleatórias, considera-se a definição de um conjunto de variáveis aleatórias definido por  $X \in \{x_1, x_2, \dots, x_n\}$ , onde  $x_n$  são valores que  $X$  pode assumir. O experimento é realizado  $N$  vezes quando  $N \rightarrow \infty$ . O conjunto  $\{m_1, m_2, \dots, m_n\}$  é o conjunto de resultados favoráveis a  $\{x_1, x_2, \dots, x_n\}$ , onde  $m_n$  é o número de ocorrências de  $x_n$ . Sendo assim, a média  $\mu_X$  das variáveis aleatórias é definida pela equação (2.11).

$$\mu_X = \frac{(m_1 x_1 + m_2 x_2 + \dots + m_n x_n)}{N} \quad (2.11)$$

Dessa forma, o resultado do número de ocorrências  $m_n$  dividido pelo número total de resultados encontrados  $N$  é a própria função de densidade de probabilidade da variável aleatória  $x_n$ . Essa definição está matematicamente apresentada na equação (2.12).

$$f_X(x_n) = \frac{m_n}{N} \quad (2.12)$$

A média  $\mu_X$  de uma variável aleatória [SHA74] é a soma dos valores esperados individuais e expressos matematicamente como apresentado na equação (2.13). Essa equação resulta do teorema fundamental de variáveis aleatórias, onde se define que a média da soma é a soma das médias [WAG72].

$$E(x_1 + x_2 + \dots + x_n) = \mu_X = \sum_{i=1}^n \mu_i \quad (2.13)$$

Outra maneira de expressar a média  $\mu_X$  de uma variável aleatória são os valores que a variável aleatória assume, multiplicado pela probabilidade de ocorrência desse valor. Para um conjunto de variáveis aleatórias discretas, considera-se a equação (2.14), onde o cálculo da média ocorre em função da somatória.

$$\mu_X = \sum_{i=1}^n x_i \cdot p_X(x_i) \quad (2.14)$$

Da mesma maneira a equação (2.15) é utilizada para variáveis aleatórias contínuas, onde o cálculo da média ocorre em função integral.

$$\mu_X = \int_{-\infty}^{+\infty} x \cdot f_X(x) dx \quad (2.15)$$

A média  $\mu_Y$  de uma função de variável aleatória, definida por  $Y = g(X)$ , é obtida através da probabilidade em função de  $X$ . Para um conjunto de funções de variáveis aleatórias discretas, a equação (2.16) define o cálculo da média ocorre em função da somatória.

$$\mu_Y = \sum_{i=1}^n g(x_i) \cdot p_X(x_i) \quad (2.16)$$

A equação (2.17) calcula a média em função integral para as funções de variáveis aleatórias contínuas.

$$\mu_Y = \int_{-\infty}^{+\infty} y \cdot f_Y(y) dy = \int_{-\infty}^{+\infty} g(x) \cdot f_X(x) dx \quad (2.17)$$

As propriedades de operações aritméticas, para as operações de médias de duas variáveis aleatórias, são aplicáveis também às operações das médias. As médias da soma (2.18), do produto (2.19), e da média quadrática da soma (2.20), em função de  $X$  e  $Y$ , também devem ser consideradas para efeitos de cálculo entre duas variáveis aleatórias. No entanto, para a média do produto e a média quadrática das somas, somente são consideradas as variáveis aleatórias independentes.

$$\overline{X + Y} = \overline{X} + \overline{Y} \quad (2.18)$$

$$\overline{X \cdot Y} = \overline{X} \cdot \overline{Y} \quad (2.19)$$

$$\overline{(X + Y)^2} = \overline{X^2} + \overline{Y^2} + 2 \cdot \overline{X} \cdot \overline{Y} \quad (2.20)$$

#### 2.2.4. Momentos

O momento de uma variável aleatória informa a variância dos valores em torno da média. O momento de uma variável aleatória é definido pela função densidade de probabilidade. O momento de ordem  $n$  de uma variável aleatória contínua é definido pela equação (2.21).

$$E[X^n] = \int_{-\infty}^{+\infty} x^n \cdot f_x(x) dx \quad (2.21)$$

O momento de ordem central, ou de ordem geral, de uma variável aleatória contínua é definido pela equação (2.22).

$$E[(x - m)^n] = \int_{-\infty}^{+\infty} (x - m)^n \cdot f_x(x) dx \quad (2.22)$$

#### 2.2.5. Variância

A variância representa a dispersão da variável aleatória em torno da sua média. A variância de uma soma de variáveis aleatórias  $\sigma_T^2$  é definida na equação (2.23).

$$\sigma_T^2 = \sum_{i=1}^n \sigma_i^2 + 2 \sum_{i=1}^{n-1} \sum_{j=i+1}^n \sigma_{ij} \quad (2.23)$$

No entanto,  $\sigma_i^2$  é a variância da variável aleatória na posição  $i$  e  $\sigma_{ij}$  são termos de covariância. Os termos de covariância podem ser expressos como na equação (2.24).

$$\text{cov}(x_i, x_j) = \sigma_{ij} = E(x_i x_j) - E(x_i)E(x_j) \quad (2.24)$$



### 2.3. Processos Aleatórios

Define-se **processo aleatório**  $\{X(t), t \in T\}$  como um conjunto de variáveis aleatórias que variam em função do tempo. Na literatura há referências de processos aleatórios denominados como processos estocásticos. O espaço amostral para um processo estocástico é definido como o conjunto de todos os valores possíveis que uma variável aleatória  $X(t)$  pode assumir [ROS00]. Para cada  $t \in T$  existe uma variável aleatória  $X(t)$ , onde o índice  $t$  é interpretado como o tempo e  $X(t)$  como o estado ou amostra do processo no instante  $t$ .

Uma variável determinística  $X(A_j, t) = X_j(t)$ , obtida em um experimento, é resultado das funções amostrais do evento  $A$  e do tempo  $t$ . Sendo assim, é possível determinar  $N$  funções amostrais para um conjunto de funções  $\{X(t)\}$  e para um evento específico  $A_j, X(A_j, t) = X_j(t)$ . As funções amostrais possuem um comportamento imprevisível e de resultados independentes.

Quando  $T$  é um valor contável, o processo estocástico é dito como um processo de tempo discreto. Um exemplo de um processo estocástico de tempo discreto é quando o processo está em função dos números inteiros não-negativos, definidos pelo conjunto  $\{X_n, n = 0, 1, \dots\}$ .

Se  $T$  é intervalo de tempo compreendido no conjunto dos números reais, o processo estocástico é dito como um processo de tempo contínuo. Um exemplo de um processo estocástico de tempo contínuo é quando o processo está em função dos números reais não-negativos, definidos pelo conjunto  $\{X(t), t \geq 0\}$ .

Para um processo aleatório, a **média** ou valor esperado [ROS97] (do inglês *expected value*) é definido em função das probabilidades para variáveis aleatórias discretas  $P(X)$  e da função de densidade de probabilidades  $p(X)$ . A média  $\mu$  de um processo aleatório para variáveis aleatórias discretas é definida pela função (2.25).

$$\mu = \overline{X(x)} = E[X] = \sum_i x_i \cdot P\{X = x_i\} \quad (2.25)$$

Para variáveis aleatórias contínuas a média é definida pela função (2.26), em analogia à função (2.25) apresentada em [ROS97]. Seguem as declarações das fórmulas:

$$E[X] = \int_{-\infty}^{+\infty} xp(x)dx \quad (2.26)$$

O cálculo da **variância** em um processo aleatório consiste em considerar o quadrado da diferença entre o valor médio e a média. Sendo assim, de acordo com a definição apresentada em [ROS97], onde dada uma variável aleatória  $X$  com média  $\mu$ , então a variância de  $X$  representada por  $\sigma^2(x)$  é definida pela função (2.27).

$$\sigma^2(x) = E[(X - \mu)^2] \quad (2.27)$$

A função (2.28) é derivada da função (2.27) apresentada também em [ROS97].

$$\sigma^2(x) = E[X^2] - (E[X])^2 \quad (2.28)$$

Processos aleatórios podem requerer a definição de métricas adicionais para mensurar algum determinado comportamento, através das estatísticas das variáveis aleatórias. Conforme proposto por [FIT01], relacionam-se as medidas de dispersão com as medidas de tendência central para identificar um coeficiente de variação.

O coeficiente de variação será definido por *cov* (do inglês *Coefficient of Variation*). Esse coeficiente vem por identificar uma métrica na variação dos tamanhos dos quadros. Quanto maior os coeficientes da variação, mais variáveis são os tamanhos dos quadros. A equação (2.29) define a relação entre o desvio padrão e a média para cálculo do coeficiente de variação.

$$\text{cov} = \frac{\sigma}{\mu} \quad (2.29)$$

Da mesma forma, uma outra métrica é definida para verificar a diferença entre o valor médio e o valor máximo obtido no experimento. Essa métrica irá definir o pico da média e será definida em (2.30) por  $p_\mu$  (do inglês *Peak-to-Mean*). Por essa métrica será possível identificar a variação do valor máximo em relação a média, somente a considerar as medidas de tendência central. Esse valor mensura o grau de dispersão em relação à medida central.

$$p_\mu = \frac{X_{\max}}{\mu} \quad (2.30)$$

Processos estocásticos estacionários são aqueles em que, mesmo num processo com propriedades aleatórias, as características estatísticas não se alteram em função do tempo. Processos não-estacionários são aqueles onde as características estatísticas se alteram em função do tempo.

Processos estacionários de primeira ordem são processos onde o valor da média (2.25) é sempre constante, independentemente do intervalo de tempo considerado. Processos estacionários de segunda ordem são processos onde o valor da função de distribuição de probabilidade (2.7) é inalterado, com exceção do intervalo de tempo definido no processo.

Outra característica é a equivalência da função de autocorrelação entre os processos estacionários em diferentes intervalos no tempo. Os processos estacionários de segunda ordem são equivalentes em valores da autocorrelação, onde o parâmetro de dependência é a diferença no instante de tempo. A definição de função de autocorrelação será apresentada nas seções 2.4 e 2.5 e a equivalência da autocorrelação em processos auto-similares será apresentada na Seção 2.7.

## 2.4. Autocorrelação

A correlação de uma variável é o produto de duas variáveis aleatórias. Ao tratar de duas amostras de processos aleatórios, considera-se que cada variável é obtida em um instante diferente do tempo. Quando as duas variáveis estão em um mesmo instante de tempo, essas variáveis possuem alta correlação. No entanto ao considerar instantes diferentes de tempo, essa diferença é parâmetro para a função de autocorrelação.

Define-se autocorrelação ou correlação serial, como a correlação de uma variável aleatória em relação a ela mesma sucessivamente, determinada em um intervalo de tempo. Se

as variáveis aleatórias  $X(t_1)$  e  $X(t_2)$  são denotadas por  $X_1$  e  $X_2$  respectivamente, em um determinado deslocamento do tempo, então em um processo estocástico real, a função de autocorrelação  $r_X(t_1, t_2)$  é formalmente definida em (2.31), a partir da função de densidade como:

$$r_X(t_1, t_2) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x_1 x_2 f_{X_1} f_{X_2}(x_1, x_2) \cdot dx_1 dx_2 \quad (2.31)$$

Outra forma de apresentação da função de autocorrelação  $r(k)$  é quando as variáveis aleatórias estão definidas somente em função do intervalo de tempo  $k$ . Essa diferença de tempo entre  $t_2$  e  $t_1$  é definida por  $k = t_2 - t_1$  e a função de autocorrelação é generalizada (2.32) como segue:

$$r(k) = E[X_n X_{n+k}] \quad (2.32)$$

## 2.5. Autocovariância

Define-se autocovariância como a covariância entre dois instantes de um mesmo processo aleatório, dado o valor no tempo  $t_2$  e o valor no tempo  $t_2 - t_1$ . Considerando-se os valores para média  $\mu$  e variância  $\sigma^2$ , formalmente a função de autocovariância  $v$  (2.33) define-se como:

$$v(k) = \frac{E[(X_n - \mu)(X_{n+k} - \mu)]}{\sigma^2} \quad (2.33)$$

Um processo estocástico, cujas características estatísticas não variam com o tempo, é classificado como um processo estocástico estacionário. A variável aleatória possui a mesma densidade em todos os tempos de amostras. Outra característica é que a autocorrelação só depende da diferença do tempo entre  $t_2 - t_1$ .

A diferença entre a autocorrelação e a autocovariância está em função do resultado da divisão entre a média  $\mu$  e a variância  $\sigma^2$ . Dessa forma, uma nova função (2.34) é apresentada, a relacionar a autocorrelação com a autocovariância.

$$v(k) = r(k) - \frac{\mu}{\sigma^2} \quad (2.34)$$

Em se tratando de modelagem de tráfego de vídeo, a equação de autocovariância (2.34) apresenta a relação entre a autocorrelação e a autocovariância. A normalização da autocorrelação, geralmente é referenciada na literatura como ambas as funções de autocorrelação e de autocovariância [ANS02]. A diferença entre as duas funções está na normalização, em função da média e a variância, como apresentado na equação (2.35).

$$\mu/\sigma^2 \quad (2.35)$$

## 2.6. Processos de Markov

Um processo aleatório  $\{X(t), t \geq 0\}$  é um **Processo de Markov** se o futuro, dado o presente, é independente do passado. Sendo assim determina-se o próximo estado de um processo, onde são conhecidos o estado atual e as probabilidades dos próximos estados em que o processo pode assumir. Nessas condições, os dados coletados anteriormente não interessam para o modelo. O que se considera é do presente em diante.

Formalmente a equação (2.36) define o processo de Markov quando se assume valor discreto para  $X(t)$ .

$$P[X(t_{n+1}) = x_{n+1} | X(t_n) = x_n, X(t_{n-1}) = x_{n-1}, \dots, X(t_1) = x_1] = P[X(t_{n+1}) = x_{n+1} | X(t_n) = x_n] \quad (2.36)$$

Quando  $X(t)$  assume valores contínuos, então o processo de Markov é definido pela equação (2.37).

$$P[a < X(t_{n+1}) \leq b | X(t_n) = x_n, X(t_{n-1}) = x_{n-1}, \dots, X(t_1) = x_1] = P[a < X(t_{n+1}) \leq b | X(t_n) = x_n] \quad (2.37)$$

Os processos de Markov podem ser descritos por uma matriz de probabilidades de transição de estados, e também por um diagrama de estados. A matriz de transições de estados provê as probabilidades dos próximos estados dado o estado atual do processo. A definição das dimensões dessa matriz depende do número de estados possíveis definido por  $N$ , a resultar em uma matriz de duas dimensões  $N \times N$ .

### 2.6.1. Cadeias de Markov

Um Processo de Markov que assume somente valores discretos (inteiros) é chamado de **Cadeia de Markov**. As cadeias de Markov possuem as propriedades relacionadas abaixo [SHA74]:

- O conjunto de sucessos possíveis é finito.
- A probabilidade do próximo sucesso dependerá apenas do sucesso atual.
- Essas probabilidades são constantes durante todo o tempo.

Cada sucesso individual é chamado de estado. O número de estados possíveis é o número de sucessos possíveis. Toda vez que ocorre um novo sucesso ou resultado, diz-se que o processo percorreu um passo ou aumento de um passo. Esse procedimento pode ocorrer indefinidamente. O passo representa um período de tempo ou qualquer outra condição que resulte em outro sucesso possível.

Por questão de notação, um estado é definido por  $S_i$ , onde  $i$  representa o estado de um total de  $m$  estados possíveis, onde  $1 < m < \infty$ . O número de passos ou incrementos do processo será representado por  $n$ .

Uma cadeia de Markov  $X_n$  tem probabilidades de transição homogêneas, se as probabilidades de transição para um passo são fixas e não variam com o tempo. Essas probabilidades podem ser representadas em uma matriz de probabilidades de transição definida pela equação (2.38).

$$P = \begin{bmatrix} p_{00} & p_{01} & \cdots & p_{0j} \\ p_{10} & p_{11} & \cdots & p_{1j} \\ \vdots & \vdots & \ddots & \vdots \\ p_{i0} & p_{i1} & \cdots & p_{ij} \end{bmatrix} \quad (2.38)$$

Essa matriz é uma matriz quadrada de tamanho  $m$ , onde cada elemento da tabela representa uma probabilidade de iniciar em um estado de origem  $i$  e chegar a um estado de destino  $j$ , no próximo passo. O valor dessa probabilidade deve sempre estar entre 0 ou 1 (2.39), refletindo que é impossível uma probabilidade negativa ou superior a 1.

$$0 \leq p_{ij} \leq 1 \quad (2.39)$$

A somatória dos valores dessas probabilidades deve assumir sempre um valor unitário. A equação (2.40) descreve matematicamente essa condição.

$$\sum_{j=1}^n p_{ij} = \sum_{j=1}^n P[X_{n+1} = j | X_n = i] = 1, \quad i = 1, 2, \dots, n \quad (2.40)$$

A **probabilidade de transição** (2.41) para  $n$  passos é calculada de acordo com a matriz de transição de estados descrita na equação (2.38).

$$P(n) = \{p_{ij}(n)\} \quad (2.41)$$

Os elementos da matriz devem satisfazer às condições listadas a seguir [SHA74]:

- A probabilidade de cada elemento deve possuir valor entre 0 e 1.
- A somatória das probabilidades de cada linha é exatamente igual a 1.

A equação (2.42) apresenta o modelo probabilístico para a probabilidade da variável aleatória  $X_n$  assumir o estado  $j$ , dada a variável aleatória atual estar no estado  $i$ .

$$p_{ij}(n) = P[X_{n+k} = j | X_k = i] \quad \forall n \geq 0 \quad \forall k \geq 0 \quad (2.42)$$

A probabilidade de transição para 2 estados, considera que a probabilidade de sair do estado  $i$  em  $t = 0$ , passando pelo estado  $k$  em  $t = 1$  e terminando no estado  $j$  em  $t = 2$ . A equação (2.43) apresenta o resultado dessa operação.

$$P[X_2 = j, X_1 = k | X_0 = i] = p_{ik}(1) \cdot p_{kj}(1) \quad (2.43)$$

A equação (2.42) quando igualada com o resultado obtido em (2.43), define a probabilidade para 2 estados, definida na equação (2.44).

$$p_{ij}(2) = \sum_k p_{ik}(1) \cdot p_{kj}(1) \quad \forall i, j \quad (2.44)$$

De acordo com a equação (2.41), define-se um valor para as probabilidades para 2 estados, definida na equação (2.45).

$$P(2) = P(1) \cdot P(1) = P^2 \quad (2.45)$$

A partir da probabilidade de transição definida para 2 estados apresentada em (2.45), a probabilidade de transição para  $n$  estados é generalizada através da equação (2.46). As probabilidades de ocupação dos estados irão tender a um valor de probabilidade comum.

$$P(n) = P^n \quad (2.46)$$

Seja o vetor linha  $\bar{p}(n)$  definido por  $\bar{p}(n) = p_j(n)$ , onde cada probabilidade  $p_j(n)$  dos estados  $n$  está atribuída a um índice  $j$ . A probabilidade  $p_j(n)$  relaciona-se a  $\bar{p}(n-1)$ , através da equação (2.47).



$$p_j(n) = \sum_i P[X_n = j | X_{n-1} = i] \cdot P[X_{n-1} = i] = \sum_i p_{ij} \cdot p_i(n-1) \quad (2.47)$$

A equação (2.47) apresenta que o vetor linha  $\bar{p}(n)$  é obtido pela multiplicação do vetor linha  $\bar{p}(n-1)$  pela matriz de transição de probabilidades  $P$ . A equação (2.48) apresenta essa condição.

$$\bar{p}(n) = \bar{p}(n-1) \cdot P \quad (2.48)$$

As cadeias de Markov em tempo contínuo seguem a função de probabilidades, como definido na equação (2.49).

$$P[X(s+t) = j | X(s) = i] = P[X(t) = j | X(0) = i] = p_{ij}(t), \quad \forall s \quad t \geq 0 \quad (2.49)$$

Seja  $P(t) = \{p_{ij}(t)\}$  a matriz de transição de probabilidades em um intervalo de comprimento  $t$ . Desde que  $p_{ii}(0) = 1$  e  $p_{ij}(0) = 0$  para  $i \neq j$ , tem-se  $P(0) = I$ , onde  $I$  é a matriz identidade.

## 2.7. Processos Auto-Similares

Essa seção apresenta a definição de processos auto-similares e as suas propriedades matemáticas e estatísticas. O principal parâmetro de verificação é a autocorrelação de um processo aleatório e comparação com o valor de um outro processo denominado auto-similar. Outra característica é a metrificação do grau de similaridade entre os processos auto-similares, através da estimativa do coeficiente de auto-similaridade. Esse coeficiente informa o grau de similaridade dos processos aleatórios.

A estimativa do coeficiente de auto-similaridade depende de grandes intervalos para indicar a presença de auto-similaridade. Essa é a característica da dependência de longa duração dos processos auto-similares. Os valores estimados para o coeficiente de auto-

similaridade, para processos observados em um pequeno intervalo de tempo, devem atingir valores muito pequenos. Essa situação indica uma baixa similaridade entre os processos aleatórios.

### 2.7.1. Definição

Os processos auto-similares, ou também encontrados na literatura [PRA96] como processos fractais, são processos aleatórios  $X(t)$  com variáveis aleatórias  $X_t$ . Os processos possuem a mesma autocorrelação em relação a outro processo, definido pelo agrupamento de amostras de outro processo aleatório. A manifestação da auto-similaridade acontece matematicamente na ocorrência consecutiva de um fenômeno, durante longos períodos de observação.

$$X(t) = (X_t : t = 1, 2, \dots) \quad (2.50)$$

A média do processo aleatório estacionário  $X(t)$ , definido em (2.50), possui uma média constante obtida pela fórmula (2.51).

$$\mu = E[X_t] \quad (2.51)$$

Ao considerar a definição (2.50) de um processo aleatório estacionário  $X(t)$ , a variância finita para um processo auto-similar é obtida pela fórmula (2.52).

$$\sigma^2 = E[(X_t - \mu)^2] \quad (2.52)$$

Dado um processo aleatório estacionário  $X(t)$  definido em (2.50), com média constante definida em (2.51) e variância finita (2.52), define-se a função de autocorrelação em (2.53), a qual depende somente do valor de  $k$ .

$$r(k) = \frac{E[(X_t - \mu) \cdot (X_{t+k} - \mu)]}{E[(X_t - \mu)^2]}, \quad k \geq 0 \quad (2.53)$$

Como apresentado em [ANS02], as funções de autocovariância e de autocorrelação em geral são utilizadas para representar a mesma informação. A diferença é a divisão pela covariância e a subtração da média a partir do valor da autocorrelação. Essa situação está devidamente definida na fórmula (2.34).

Ao aproximar o valor da função de autocorrelação (2.54) como proposto em [PRA96], um processo aleatório pode ser definido a partir da respectiva função de autocorrelação.

$$r(k) \approx \alpha \cdot k^{-\beta}, \quad k \rightarrow \infty, \quad 0 < \beta < 1 \quad (2.54)$$

Um novo processo aleatório estacionário  $X^{(m)}$  definido em (2.55) é gerado a partir dos valores das médias da seqüência do processo original  $X$ . Os valores da média são calculados a partir do agrupamento de blocos definido por  $m$  elementos.

$$X^{(m)} = (X_k^{(m)} : k > 0), \quad m > 0 \quad (2.55)$$

Para cada agrupamento  $m$ , o processo  $X^{(m)}$  é calculado a partir da equação (2.56).

$$X_k^{(m)} = \frac{(X_{km-m+1} + \dots + X_{km})}{m}, \quad k > 0 \quad (2.56)$$

Em [PRA96] define-se o processo  $X_k^{(m)}$  como processo auto-similar de segunda ordem (do inglês *second-order self-similar process*), dado o parâmetro de similaridade do coeficiente de auto-similaridade definido em (2.57), ou coeficiente Hurst, como apresentado em [PRA96], e pela equivalência das propriedades estatísticas definidas em (2.58) e (2.59) para o processo auto-similar, em função da variância e da autocorrelação respectivamente.

$$H = 1 - (\beta/2) \quad \therefore \quad \beta = 2 - (2 \cdot H) \quad (2.57)$$

$$\text{var}(X^{(m)}) = \sigma^2 \cdot m^{-\beta}, \quad m > 0 \quad (2.58)$$

$$r^{(m)}(k) = r(k), \quad k \geq 0 \quad (2.59)$$

A equivalência dos valores da autocorrelação entre o processo  $X(t)$  e o processo agregado  $X_k^{(m)}$  define a similaridade entre os processos. O processo  $X(t)$  com  $r(k) \approx k^{-\beta}, k \rightarrow \infty$  e  $0 < \beta < 1$  é chamado de processo exatamente auto-similar, se o processo agregado  $X_k^{(m)}$  possuir identicamente a mesma estrutura da autocorrelação como definido em (2.59). Em outras palavras, o processo  $X(t)$  e o processo agregado das médias  $X_k^{(m)}, m \geq 1$  possuem idêntica estrutura de autocorrelação.

Um processo estacionário  $X(t)$  é denominado de processo assintoticamente auto-similar considerando (2.59), se  $r^{(m)}(k) \rightarrow r(k)$  enquanto  $m \rightarrow \infty$  sendo  $r(k)$  definida em (2.54). Como o valor das funções de autocorrelação é similar, o gráfico da autocorrelação somente se diferencia em função das variáveis aleatórias não serem distribuídas independentemente. Essa característica é apresentada na equação (2.60).

$$r^{(m)}(k) \rightarrow 0, \quad m \rightarrow \infty \quad (2.60)$$

### 2.7.2. Propriedades

As propriedades para esse tipo de processo são:

- Dependência de longa duração (LRD do inglês *long-range dependence*) e de longa memória (do inglês *long memory*).
- Declínio lento da variância.
- Efeito do coeficiente de Hurst.

Formalmente, processos de longa duração são caracterizados por uma função de autocorrelação que decresce hiperbolicamente no intervalo  $k$ . Essa propriedade implica em que o resultado da soma dos resultados obtidos da função de autocorrelação tenda ao infinito (2.61). Por outro lado, processos de dependência de baixa duração (SRD do inglês *short-range dependence*) possuem crescimento exponencial em relação à função de autocorrelação.

$$\sum_k r(k) = \infty \quad (2.61)$$

Os valores das variâncias (2.62) possuem declínio lento em função do tempo, em comparação ao tamanho relativo da amostra. Considerar em (2.62)  $0 < \beta < 1$ .

$$\sigma^2(X_k^{(m)}) \rightarrow m^{-\beta}, \quad m \rightarrow \infty \quad (2.62)$$

A definição do coeficiente de Hurst  $H$  mede o grau de similaridade de valores em função do tempo. Obtém-se  $H$  por cálculos estimados de um processo auto-similar. Quando o coeficiente de Hurst assume um valor em torno de  $0,5$ , então o processo é SRD e não possui memória. Essas características indicam que o processo não apresenta similaridade [FIT03]. Por outro lado, quando esse valor está compreendido entre  $0,5$  e  $1$ , então o processo é caracterizado como um processo auto-similar de longa duração e de memória infinita (2.61). Valores acima de  $1$  indicam um processo altamente auto-similar.

O coeficiente de Hurst, ou parâmetro de auto-similaridade, destaca uma característica expressiva dos processos auto-similares. Esse coeficiente é utilizado para quantificar as taxas de pico durante obtenção de valores em grande escala. Esses valores altamente variáveis estão presentes em processos de características de dependência de longa duração [FIT01], quando considerado um longo tempo de observação.

Em uma observação de tráfego de pacotes, por exemplo, os valores de Hurst quantificam a variação das rajadas, razão entre a taxa de pico e a taxa média do tráfego. O coeficiente de Hurst por sua vez [FIT03] é uma medida da persistência de um fenômeno estatístico e mede o tamanho da dependência de longa duração do processo aleatório.

### 2.7.3. Estimativa do Coeficiente de Auto-Similaridade

O coeficiente de Hurst, ou coeficiente de auto-similaridade, é estimado através da interpolação do logaritmo da equação R/S (2.68) (do inglês *Rescaled Adjusted Range Statistic*), em função do logaritmo do intervalo  $k$  [FIT01]. O conjunto de valores  $k$  é um intervalo definido entre o primeiro quadro e o quadro indicado no índice  $k$ . O gráfico R/S provê a interpretação gráfica da estimativa de Hurst definida pelo logaritmo do intervalo  $k$ , em função do logaritmo obtido por R/S.

Os valores para R/S são calculados de acordo com a somatória parcial definida por  $Y(k)$ , em cada intervalo  $k$ . A equação (2.63) apresenta a somatória definida para o intervalo  $k$ , a partir de um processo aleatório  $X(k)$ .

$$Y(k) = \sum_{i=1}^k X_i \quad (2.63)$$

A partir do valor da equação (2.63), chega-se à equação (2.64) para obtenção da média de  $Y(k)$ , a ser definida por  $\bar{Y}(k)$ .

$$\bar{Y}(k) = \frac{Y(k)}{k} \quad (2.64)$$

A próxima etapa requer o cálculo da variância do processo  $X(k)$ , no intervalo definido por  $k$ . A equação (2.65) apresenta a equação para obtenção da variância definida no intervalo.

$$\sigma_X^2(k) = \frac{1}{k} \cdot \sum_{i=1}^k \left[ X_i^2 - \left( \frac{1}{k} \right)^2 \cdot Y^2(k) \right] \quad (2.65)$$

As somas parciais dos valores definidos no intervalo  $k$  devem ser consideradas. Para cada índice  $j$  definido nesse subintervalo, onde o valor deve estar compreendido entre  $0 \leq j \leq k$ . Esses valores devem gerar uma nova lista de valores definida por  $W(j)$ , de tamanho  $k$ , com os resultados da somatória acumulada para cada item  $j$ . O resultado dessa somatória é igual à subtração do valor da variável aleatória  $Y(j)$  do índice do subintervalo  $j$ , multiplicado pela média acumulada  $\bar{Y}(k)$ , no intervalo. Esse cálculo está formalmente definido na equação (2.66).

$$W(j) = \sum_{j=0}^k (Y(j) - j \cdot \bar{Y}(k)) \quad (2.66)$$

A partir dos valores mínimos e máximos definidos em  $W(j)$ , calcula-se o coeficiente R/S, o qual será utilizado para estimar o coeficiente de Hurst. A variância nesse cálculo é convertida para o desvio-padrão, aplicando-se a raiz quadrada da variância. A equação (2.67) apresenta a formalização da obtenção do coeficiente R/S.

$$\frac{R}{S}(k) = \frac{[\max(W(k)) - \min(W(k))]}{S_x(k)} \quad (2.67)$$

Ao gerar o gráfico de  $\log(k)$  em função de  $\log(R/S(k))$ , o resultado dessa divisão resulta no coeficiente de Hurst, definido para o experimento. A equação (2.68) define o processo de obtenção do coeficiente de Hurst.

$$\frac{R}{S}(k) = k^H \rightarrow H = \frac{\log(R/S(k))}{\log(k)} \quad (2.68)$$

Uma característica para esse tipo de gráfico é que quanto menor o tempo de medição, o coeficiente de Hurst se aproxima de 0,5. Com o decorrer do tempo, então o coeficiente de Hurst tende a chegar a 1. No entanto, cada processo aleatório possui uma característica única, que exige uma análise isolada.

Outros métodos de estimativa de Hurst foram encontrados na literatura. Como por exemplo, o método de estimativa de Hurst por variância é apresentado em [PRA96], [GAR94] e em [QIN94]. No entanto, os resultados obtidos em relação aos processos aleatórios foram similares aos valores obtidos no método R/S.

## **2.8. Conclusões**

Os conceitos estatísticos apresentados visam formular um embasamento teórico, em relação aos métodos de modelagem e simulação de processos aleatórios. Esses conceitos apresentam fórmulas de análise, quanto ao comportamento de um processo e também como evidenciar matematicamente o comportamento de um processo simulado em relação a um processo de geração de variáveis aleatórias de um processo real.

As características presentes em processos auto-similares devem ser verificadas tanto em um conjunto de valores aleatórios, baseado em observações no ambiente, quanto nos modelos de simulação de processos auto-similares. Através do coeficiente de Hurst é possível mensurar e calibrar a similaridade em um modelo de simulação.



## Capítulo 3

### Métodos de Compressão de Vídeo

O método para compressão de vídeo MPEG (do inglês *Moving Picture Experts Group*) será usado como objeto de estudo sobre mecanismos de proteção de vídeo em tempo de codificação, em especial a técnica de codificação hierárquica. As características de agrupamento dos tipos de quadros definem a estrutura hierárquica de proteção e conseqüentemente definem o procedimento para avaliação do modelo de simulação.

#### 3.1. Análise do padrão de Compressão MPEG

Os métodos de codificação têm a finalidade de organizar as informações para armazenamento. A codificação por compressão [SEE04] tem a finalidade de eliminar a redundância das informações em um conjunto de dados. Esse método gera perda de informações, por ocorrer arredondamento e compensação das informações, devido à redundância. Essa característica reduz a qualidade visual do vídeo apresentado. A compressão tem como objetivo reduzir os custos de armazenamento e de comunicação. As técnicas de codificação podem ser categorizadas como segue:

- Entropia: Os dados são processados como uma seqüência de dígitos, existindo compressão sem perda (método de Huffman) [GAR94];
- Fonte: A semântica dos dados é considerada, ao separar dados relevantes dos irrelevantes. A codificação pode ser com perda (DCT, Compensação de Movimento) ou sem perda (Quantização);
- Codificação Híbrida: aplicação de ambas as técnicas.

O padrão de compressão MPEG é baseado no método de codificação de quadros, assim como o padrão JPEG. Esses padrões consideram que o olho humano não é capaz de perceber todos os detalhes de uma figura. As imperfeições da imagem não são percebidas em regiões onde a visão humana não explora, geralmente em regiões periféricas. Dada essa situação, o procedimento de codificação de um fluxo multimídia pode utilizar pontos da imagem comum a quadros recebidos anteriormente e na previsão de quadros que estão a chegar no fluxo de vídeo.

As técnicas de codificação de vídeo consideram duas propriedades fundamentais na construção das seqüências de imagens. A primeira é a redundância espacial, ou *intra-frame*, que considera as informações contidas internamente em uma mesma imagem. A redundância espacial avalia a informação contida nos quadros vizinhos, do quadro em questão, em uma mesma imagem.

A segunda é a redundância temporal, ou *inter-frame*, que considera as informações entre os quadros sucessivos de imagens diferentes. As informações dos quadros são avaliadas em quadros consecutivos em função do tempo, por estimativa e compensação de movimento entre os quadros sucessivos.

Como descrito em [ROD96], a aplicação do algoritmo DCT (do inglês *Discrete Cosine Transform*) é o primeiro no processo de compressão de dados. O cálculo dos valores DCT explora a propriedade de redundância espacial entre os blocos, onde os valores de maior intensidade dos pontos da imagem (*pixels*) são transformados em valores de freqüência, os quais são denominados de Coeficientes DCT. Esses coeficientes representam as informações em diferentes freqüências espaciais e de forma compactada.

Cada quadro de vídeo é dividido em blocos 8 X 8 amostras de quadros. Cada bloco é transformado utilizando-se a DCT em um bloco 8 X 8 de coeficientes, os quais representam a freqüência espacial dos componentes no bloco original.

O processo de quantização ocorre num segundo momento, onde os dados são relacionados em entradas de uma tabela de correspondência, ou tabela de quantização. Como apresentado em [DEV01], os valores de freqüência mais baixos são arredondados e desconsiderados. O processo de arredondamento e descarte de valores baixos, resulta nas perdas de resolução da imagem. Sendo assim, quanto maior a quantização, menor a qualidade de resolução do vídeo.

Os valores de frequência são ordenados em uma tabela para facilitar a compressão, de cima para baixo e da esquerda para a direita, tal qual uma ordenação em ziguezague. A tabela de correspondência é apresentada na Figura 3.1. Os valores relacionados na tabela de correspondência são chamados de coeficientes DCT. Os valores mais altos se encontram na extremidade superior à esquerda da tabela de correspondência.

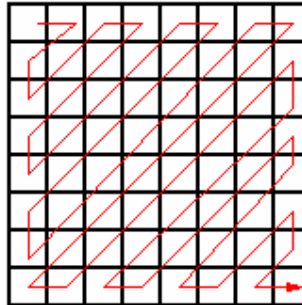


Figura 3.1: Reordenação dos Coeficientes DCT.

A compressão das informações tem impacto direto nas características visuais das imagens. Com o objetivo de reduzir o tamanho final da imagem, a compressão impacta diretamente na sua qualidade. Quanto maior a compressão, menor o tamanho e consequentemente menor a qualidade visual. A tabela apresenta as características da resolução em função da compressão, da qualidade e do tamanho final da imagem.

Tabela 3.1: Características de vídeos em diferentes resoluções.

<b>Resolução</b>	<b>Compressão</b>	<b>Qualidade</b>	<b>Tamanho</b>
<b>Alta</b>	*	***	***
<b>Média</b>	**	**	**
<b>Baixa</b>	***	*	*

### 3.2. Estrutura de Fluxo MPEG

O padrão de compressão MPEG, define a representação sincronizada de um *bit* digital do fluxo de áudio e vídeo, comprimido para se ajustar ao tamanho em uma banda de rede de 1,5 Mbits/s. São três as partes do arquivo MPEG: áudio, vídeo e sistema. O sistema MPEG é responsável por separar em fatias de tempo (*multiplex*) as mídias, e sincroniza-las em áudio e vídeo.

A principal aplicação do MPEG está em armazenar informações de áudio e vídeo em mídias de armazenamento digital, como por exemplo, discos compactos (CD) e fitas de áudio digital (DAT). Outra aplicação do MPEG consiste em definir a sintaxe e a semântica do fluxo de vídeo ao decodificar as informações no processo receptor. O método de compressão MPEG é adotado como padrão genérico de codificação para muitos modelos de codificação de vídeo.

Como ilustrado na Figura 3.2, um fluxo de vídeo MPEG inicia-se com uma seqüência de cabeçalho, seqüências alternadas de cabeçalho dos grupos de figuras (GOP, do inglês: *Group Of Pictures*) e o próprio GOP relativo à seqüência, e por final o código de finalização da seqüência MPEG.

Cabeçalho de Seqüência	Cabeçalho GOP	GOP	Cabeçalho GOP	GOP	..	Código de Finalização MPEG
------------------------	---------------	-----	---------------	-----	----	----------------------------

Figura 3.2: Descrição da Ordem das Informações em um Fluxo de Vídeo MPEG.

Um fluxo de áudio MPEG é uma seqüência de quadros, conforme ilustrado na Figura 3.3. Cada quadro possui um cabeçalho e um conjunto de dados. O conjunto de dados é codificado em valores numéricos, que representam o áudio codificado do quadro atual. A codificação de cada quadro de áudio depende dos quadros anteriores.

Cabeçalho de Áudio MPEG	Dados de Áudio MPEG	Cabeçalho de Áudio MPEG	Dados de Áudio MPEG	Cabeçalho de Áudio MPEG	Dados de Áudio MPEG
-------------------------	---------------------	-------------------------	---------------------	-------------------------	---------------------

Figura 3.3: Descrição da Ordem das Informações em um Fluxo de Áudio MPEG.

Alguns quadros próximos ao quadro atual, também podem estar codificados no conjunto de dados, se houver espaço para armazenamento de transmissão. Uma unidade máxima de transmissão na rede, equivalente a um MTU (do inglês *Maximum Transmission Unit*), é um valor padronizado com o tamanho de *1.500 bytes* para transmissão de pacotes.

### 3.3. Estrutura e Codificação do GOP

Um grupo de figuras, também referenciado por GOP, é uma série de quadros, onde cada um consiste em um cabeçalho da figura e os seus atuais dados. A seqüência de um grupo de figuras é uma seqüência de quadros entre quadros consecutivos do tipo 'I' de tamanho  $N$ .  $M$  representa a distância entre o quadro do tipo 'I' e o primeiro quadro do tipo 'P' na seqüência de quadros. A referência [DEV01] apresenta a estrutura de um GOP, onde cada quadro é dividido em  $16 \times 16$  *macroblocos*, que por sua vez são divididos por quatro blocos  $8 \times 8$  de luminosidade e dois blocos  $8 \times 8$  de crominância (do inglês *chrominance*).

Como apresentado em [DEV01], existem três tipos de quadros no padrão MPEG. Os tipos de quadros são intra-codificados ('I'), inter-codificados ('P') ou bidirecionalmente codificados ('B'). Esses tipos de quadros são organizados em grupos de figuras.

O primeiro tipo de quadro de um fluxo é o quadro intra-codificado (do tipo 'I', do inglês *intracoded*) que contém uma imagem descrevendo a cena e não possui nenhuma referência em relação aos quadros anteriores ou posteriores. Os quadros do tipo I são codificados explorando a redundância espacial dentro da figura. Cada bloco é independente e codificado pela redundância espacial, em domínios de freqüências, utilizados no cálculo DCT.

Além do quadro de informação da imagem, existe o tipo de quadro baseado na redundância temporal, ou inter-codificado [SEE04], ou ainda previsão antecipada (do tipo 'P', do inglês *forward prediction*). Esses quadros são codificados, ao considerar o quadro precedente de referência, que pode ser um quadro do tipo 'I' ou 'P'. O quadro do tipo 'P' explora a referência do vetor de movimento que se baseia nos quadros anteriores, referenciando o bloco do quadro anterior na mesma posição do quadro atual.

Quadros do tipo 'B' (do inglês *backward prediction*) também são codificados, baseados na redundância temporal. No entanto, a codificação ocorre nas referências de quadros anteriores e também dos quadros posteriores do fluxo de vídeo, que podem ser do tipo 'I' ou 'P'. O quadro do tipo 'B' explora a referência do vetor de movimento, em ambas as

direções, no futuro ou no passado, referenciando o bloco do próximo quadro na mesma posição do quadro atual.

Um GOP é a seqüência de quadros, a partir de um determinado quadro do tipo I, até o próximo quadro do tipo I. Cada GOP deve conter ao menos um quadro do tipo 'I'. O padrão existente na estrutura do GOP para os quadros do tipo I, P e B é chamado de padrão GOP ou estrutura GOP [SEE04]. Entre esses quadros existem os quadros do tipo B e P, onde para cada quadro do tipo I, existem 3 quadros do tipo P e 8 do tipo B. A relação entre os quadros ocorre como ilustrado na Figura 3.4.

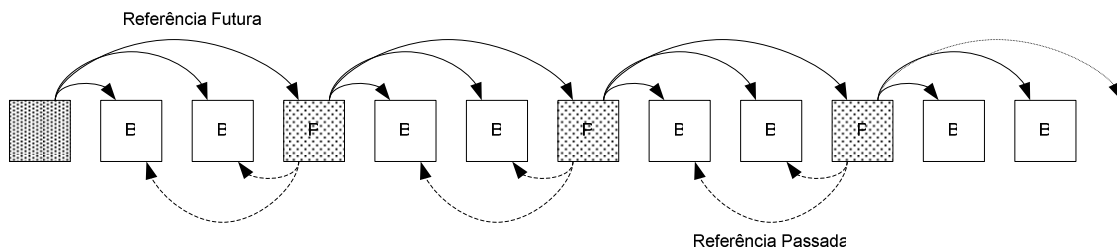


Figura 3.4: Relacionamento de Quadros IPB.

Os quadros do tipo 'B' podem ser codificados utilizando o método de compensação de movimento, baseado na previsão dos dois próximos quadros codificados, podendo ser quadros do tipo 'I' ou 'P'. A organização dos tipos de codificação dos quadros dentro da seqüência de vídeo é flexível para atender às necessidades de diversas aplicações e possuem diferentes tipos e significados durante a fase de decodificação.

O processo de compensação e estimativa de movimento explora a codificação por redundância temporal entre os quadros, ou inter-codificado [SEE04]. Esse processo ocorre no contexto dos *macroblocos* de um fluxo de vídeo. Utiliza-se a referência de um quadro situado no passado para prever as áreas do quadro atual. A interpolação de redundância temporal é o processo pelo qual se codifica os fluxos multimídia, utilizando-se a média ou a codificação bidirecional, a partir do apontamento de um quadro de referência ao passado e um quadro de referência ao futuro.

### 3.4. Conclusões

O padrão de codificação de vídeos MPEG segue a estrutura de um GOP. Um GOP geralmente é composto por 12 quadros de estrutura IBBPBBPBBPBB, os quais possuem a estrutura definida para os seguintes tipos de quadros:

- I: método de codificação dentro do quadro, que explora a redundância espacial dentro da imagem;
- P: método de codificação entre os quadros, que explora a redundância temporal dentro da seqüência do vídeo;
- B: método de codificação entre os quadros, que explora a codificação bidirecional;

Cada tipo de quadro dessa estrutura possui uma característica diferenciada quanto ao armazenamento das informações das imagens. A codificação hierárquica é definida pela redundância das informações distribuídas nos diferentes tipos de quadro, onde se define o nível de compressão e conseqüentemente a qualidade da imagem aplicada nas camadas de melhoria.

Os quadros do tipo I possuem uma média superior quanto ao valor do seu tamanho, pois armazenam mais informações requeridas na camada básica de apresentação da imagem. Os quadros do tipo B e P possuem menores tamanhos médios na mesma proporção, quanto à importância das suas informações no processo de decodificação do quadro MPEG. Essas características ficaram evidenciadas na análise realizada na próxima seção.





## Capítulo 4

### Análise Estatística de Tráfego de Vídeo

Os fundamentos matemáticos e os métodos de compressão de vídeo apresentados anteriormente, são conhecimentos requeridos para se iniciar a análise estatística do comportamento do tráfego de vídeo como um processo aleatório. Dados os resultados dessa análise, os valores obtidos serão utilizados para a construção de um modelo de simulação de tráfego de vídeo. Essas mesmas características estatísticas devem estar presentes no modelo desenvolvido.

Esse capítulo apresenta as análises estatísticas realizadas no desenvolvimento do trabalho e a metodologia para obtenção dos resultados. Serão apresentadas as formas de observação de um processo aleatório, as propriedades estatísticas relacionadas a esse processo e a relação entre essas propriedades, para o tráfego de vídeo.

#### 4.1. Arquivos de Tráfego de Vídeo

A modelagem de processos aleatórios requer a observação de fenômenos reais encontrados na natureza. Pode-se avaliar estatisticamente a relação entre os valores de característica aleatória, coletados e registrados durante a observação. Com os valores resultantes dessa avaliação, é possível modelar processos de simulação desse mesmo fenômeno. O modelo de simulação deve gerar valores aleatórios e de considerável relação com os valores reais observados.

As modelagens de tráfego de vídeo consideram pelo menos 3 maneiras [SEE04] para produzir valores aleatórios baseados na observação de transmissão de vídeo real. São elas:

- Fluxos de informações de vídeo (do inglês, *video bit streams*);
- Arquivos de vídeos, coletados em tráfego de vídeo (do inglês, *video traffic traces*);
- Modelos de tráfego de vídeo (do inglês, *video traffic models*);

O fluxo de informação de vídeo requer a utilização de vídeos reais e com isso possibilita-se trabalhar com a informação completa e detalhada em relação ao fluxo de vídeo. No entanto, deve-se considerar que perdas eventuais na transmissão podem refletir nos resultados dos dados observados em não apresentar a informação real de ocorrência da transmissão do vídeo.

Esse processo pode ter um custo elevado pelo fato de gerar informações excessivas para o objetivo em questão, além de requerer um grande tempo de processamento necessário para o registro desses valores. Com isso, é necessário desenvolver um procedimento de avaliação das informações, para se filtrar o resultado desejado para o modelo, de acordo com as características dos valores requeridos para a modelagem. Outro problema a considerar é a questão de direito autoral em relação à publicação do vídeo analisado.

Arquivos de vídeos são uma alternativa mais prática e viável para a observação do processo aleatório. Os arquivos de vídeos trabalhados nesse estudo possuem os tamanhos dos vídeos previamente coletados e devidamente organizados em uma estrutura de dados definida. Outra questão é a de não possuírem restrições quanto a direito autoral.

Modelos de tráfego de vídeo são gerados a partir de arquivos que contém informações organizadas em estruturas de dados. Com o uso de alguns parâmetros, modela-se um processo aleatório com propriedades estatísticas específicas. Os modelos de dados contém características matematicamente preferenciais, computacionalmente eficientes e de frequência precisa. Os modelos possibilitam gerar tráfego de vídeo sintetizado, nomenclatura essa também utilizada para referenciar à simulação de tráfego de vídeo.

Os arquivos gerados na observação de dados empíricos coletados em processamento de tráfego de vídeo real definiram as características da modelagem de simulação de tráfego de vídeo proposta nesse trabalho. Arquivos apresentados em [FIT01] contém as informações requeridas para atender as características do modelo de simulação.

Existem dois tipos de arquivos apresentados em [FIT01], que se diferenciam basicamente pela estrutura de organização dos dados coletados. O primeiro possui maior

detalhamento referente ao GOP, onde são apresentadas as informações como o número de ordenação do quadro, o tipo do quadro, a estampa de tempo e o tamanho do quadro. A Tabela 4.1 apresenta um trecho do arquivo de dados do vídeo *Star Wars IV* [FIT01], em baixa resolução.

Tabela 4.1: Exemplo de trecho de arquivo da estrutura de dados dos quadros.

<b>Quadro</b>	<b>Tipo de Quadro</b>	<b>Tempo (ms)</b>	<b>Tamanho (bytes)</b>
<b>1</b>	I	0	505
<b>2</b>	P	120	359
<b>3</b>	B	40	40
<b>4</b>	B	80	75
<b>5</b>	P	240	153
<b>6</b>	B	160	36
<b>7</b>	B	200	75
<b>8</b>	P	360	113
<b>9</b>	B	280	35
<b>10</b>	B	320	36
<b>11</b>	I	480	944
<b>12</b>	B	400	45
<b>13</b>	B	440	44
<b>14</b>	P	600	68
<b>15</b>	B	520	31
<b>16</b>	B	560	32
<b>17</b>	P	720	101
<b>18</b>	B	640	32
<b>19</b>	B	680	41
<b>20</b>	P	840	79
<b>21</b>	B	760	32
<b>22</b>	B	800	40
<b>23</b>	I	960	942
<b>24</b>	B	880	42

A ordenação do quadro segue a ordem de transmissão do quadro. Essa característica ocorre em função da classificação do grau de importância do quadro para a reconstrução do GOP, durante a fase de decodificação.

O outro arquivo disponível considera somente os valores dos tamanhos dos quadros, ordenados pela seqüência de apresentação dos mesmos. Assim, utilizam-se os dados da coluna tamanho da Tabela 4.2 para traçar as análises estatísticas. A Tabela 4.2 apresenta os dados apresentados na Tabela 4.1, ordenados pela seqüência de apresentação dos quadros no vídeo.

Tabela 4.2: Trecho do arquivo de dados, ordenado pela ordem de apresentação do quadro.

<b>Quadro</b>	<b>Tipo de Quadro</b>	<b>Tempo (ms)</b>	<b>Tamanho (bytes)</b>
<b>1</b>	I	0	505
<b>3</b>	B	40	40
<b>4</b>	B	80	75
<b>2</b>	P	120	359
<b>6</b>	B	160	36
<b>7</b>	B	200	75
<b>5</b>	P	240	153
<b>9</b>	B	280	35
<b>10</b>	B	320	36
<b>8</b>	P	360	113
<b>12</b>	B	400	45
<b>13</b>	B	440	44
<b>11</b>	I	480	944
<b>15</b>	B	520	31
<b>16</b>	B	560	32
<b>14</b>	P	600	68
<b>18</b>	B	640	32
<b>19</b>	B	680	41
<b>17</b>	P	720	101
<b>21</b>	B	760	32
<b>22</b>	B	800	40
<b>20</b>	P	840	79
<b>24</b>	B	880	42

A estrutura IBBPBBPBBPBB do GOP de tamanho de 12 quadros está ordenada em função do tempo. A estrutura do GOP segue o padrão definido pela ordem de apresentação dos quadros. Esse formato de arquivo, com a estruturação ordenada em função do tempo, foi utilizado na modelagem proposta.

Vários arquivos de vídeo foram capturados em [FIT01], inclusive com avaliação de diferentes resoluções para um mesmo vídeo. As diferentes resoluções representam diferentes qualidades de apresentação do vídeo. Os arquivos de vídeo possuem o tamanho e a seqüência dos quadros. A informação dos tamanhos dos quadros é o parâmetro requerido para análise estatística da variação dos valores, em função do tempo. Esses valores já estão organizados de acordo com a seqüência de apresentação definida para os quadros. Essa descrição considera a definição de processos aleatórios para valores aleatórios distribuídos em função do tempo.

## 4.2. Análise Estatística dos Arquivos de Vídeo

Uma vez considerados esses arquivos, a avaliação estatística realizada no conjunto de variáveis aleatórias definidas pelos tamanhos dos quadros em função do tempo, deve atender às definições e propriedades estatísticas aplicadas aos processos aleatórios. As simulações devem seguir as mesmas características apresentadas para as análises estatísticas de vídeos reais.

De acordo com as características de resolução dos vídeos, como apresentado na Tabela 3.1, as análises estatísticas devem considerar as diferentes resoluções de vídeo apresentadas em [FIT01]. Os valores a seguir foram obtidos através da análise estatística para os arquivos de vídeos do vídeo *Star Wars IV* em alta resolução, *Alladin* em média resolução e *Jurassic Park* em baixa resolução. Esses arquivos serão utilizados na análise estatística de tráfego de vídeo em dados reais e estão disponíveis para consulta em [FIT01].

A Figura 4.1, Figura 4.2 e Figura 4.3 apresentam os tamanhos em *bytes* dos quadros para os arquivos em alta, média e baixa resolução, respectivamente. Esses gráficos apresentam o tamanho dos quadros em *bytes*, definidos por  $X_n$ , em função do número do quadro  $n$ .

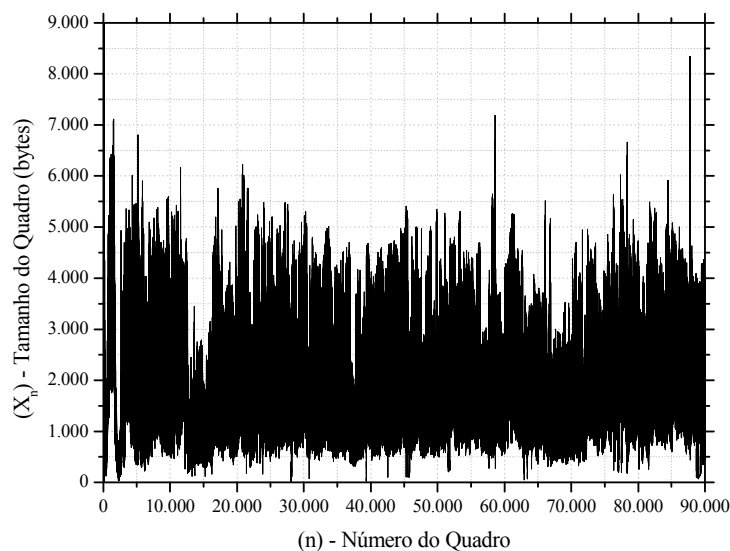


Figura 4.1: Tamanhos dos quadros em alta resolução.

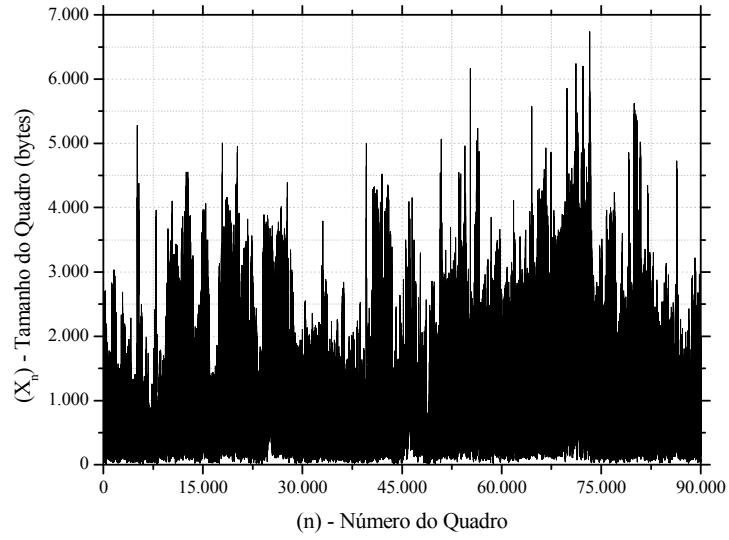


Figura 4.2: Tamanhos dos quadros em média resolução.

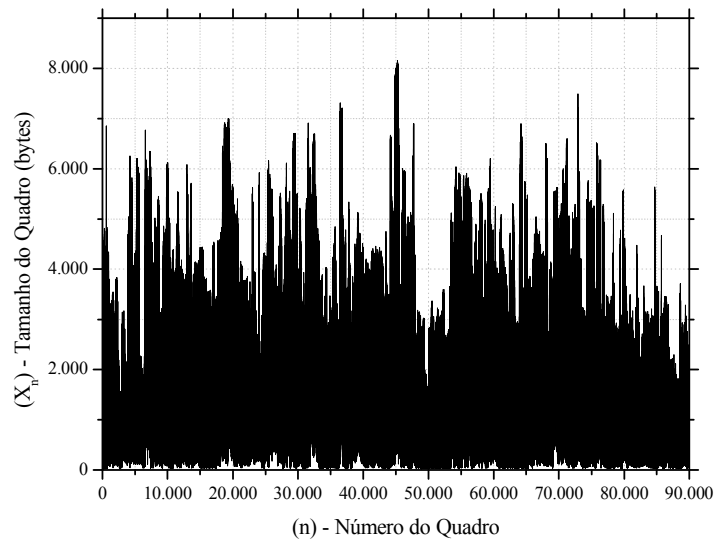


Figura 4.3: Tamanhos dos quadros em baixa resolução.

Na Figura 4.1, a codificação dos quadros em alta resolução segue uma distribuição relativamente suave dos tamanhos dos quadros, em função do número do quadro. Por outro lado, a Figura 4.3 apresenta a codificação dos quadros em baixa resolução, com grandes diferenças nos tamanhos dos quadros. Essas características representam a variações de cenas distintas nos filmes analisados [FIT03].

As figuras abaixo apresentam a distribuição de freqüências encontradas nos vídeos de baixa, média e alta resolução. Quanto menor a resolução, maior a concentração de número de quadros nas primeiras classes de freqüência, as quais agrupam quadros com menores tamanhos. Outra característica é que quanto menor for a resolução de vídeo, maior será a variação dos tamanhos dos quadros.

A Figura 4.4 apresenta um gráfico para um vídeo em baixa resolução. O intervalo de classe utilizado foi de 250 *bytes*.

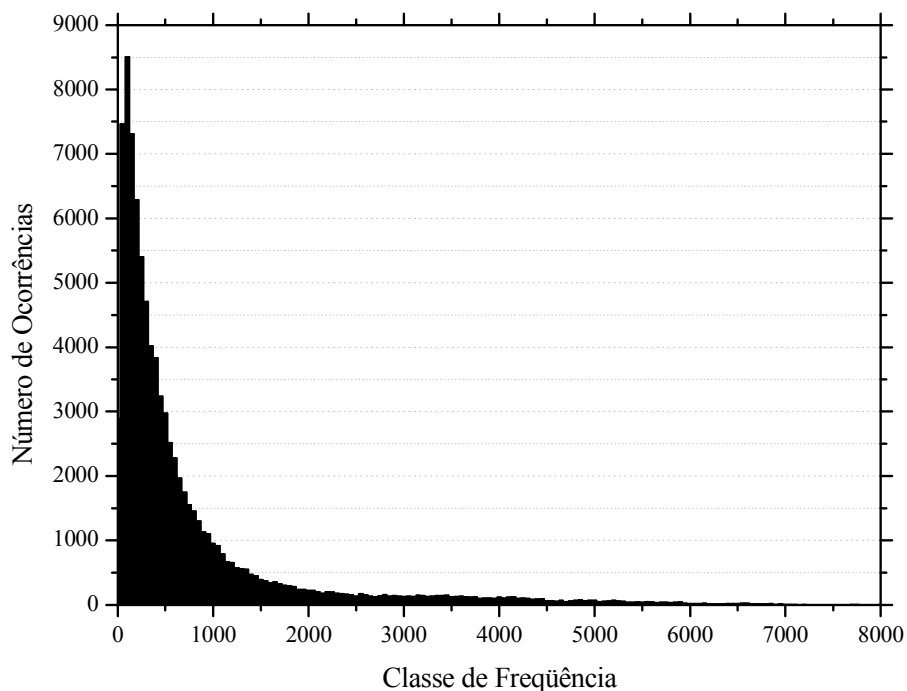


Figura 4.4: Distribuição de freqüência para um vídeo em baixa resolução.

Os dados da Tabela 4.3 apresentam que 84% dos tamanhos dos quadros chegam a até 1000 *bytes*, para o arquivo em baixa resolução. No entanto, deve-se observar que o tamanho dos quadros pode chegar a até 8500 *bytes*. Os intervalos representam o tamanho dos quadros, sucessivamente a cada 250 *bytes*.

Tabela 4.3: Percentagem de agrupamento por classe de frequência em baixa resolução.

<b>Intervalo</b>	<b>Ocorrências</b>	<b>(%)</b>
1	32451	36,057
2	21193	23,548
3	11483	12,759
4	6533	7,259
5	3974	4,416
6	2596	2,885
7	1782	1,980
8	1349	1,499
9	1020	1,133
10	862	0,958
11	705	0,783
12	696	0,773
13	670	0,745
14	683	0,759
15	613	0,681
16	529	0,588
17	555	0,617
18	470	0,522
19	293	0,325
20	337	0,375
21	291	0,323
22	226	0,251
23	196	0,218
24	164	0,182
25	81	0,090
26	66	0,073
27	80	0,089
28	48	0,053
29	20	0,022
30	2	0,002
31	6	0,007
32	11	0,012
Totais	89998	100,000%



A Figura 4.5 apresenta um gráfico para um vídeo em média resolução. O intervalo de classe utilizado foi de 250 *bytes*.

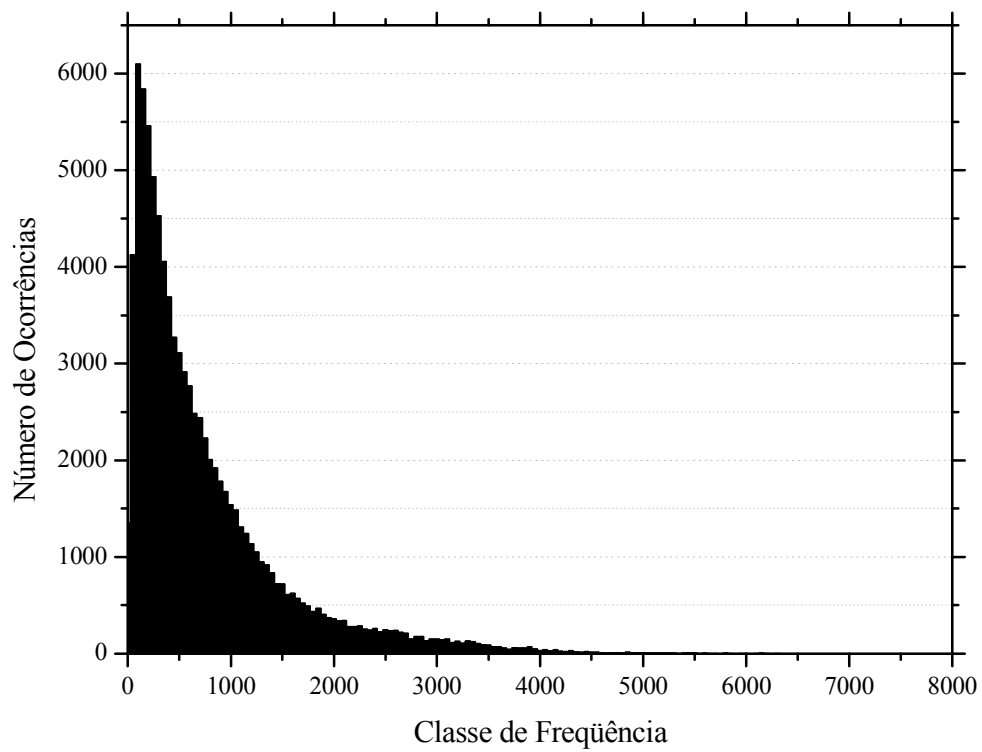


Figura 4.5: Distribuição de frequência para um vídeo em média resolução.

Na Tabela 4.4, apresenta-se que 81% dos tamanhos dos quadros chegam a até 1000 *bytes*, para o arquivo em média resolução.

Tabela 4.4: Percentagem de agrupamento por classe de frequência em média resolução.

<b>Intervalo</b>	<b>Ocorrências</b>	<b>(%)</b>
1	22855	25,395
2	20468	22,743
3	13706	15,230
4	9607	10,675
5	6705	7,450
6	4469	4,966
7	3040	3,378
8	2163	2,403
9	1591	1,768
10	1259	1,399
11	1145	1,272
12	773	0,859
13	675	0,750
14	551	0,612
15	318	0,353
16	289	0,321
17	154	0,171
18	89	0,099
19	43	0,048
20	36	0,040
21	20	0,022
22	22	0,024
23	9	0,010
24	4	0,004
25	5	0,006
26	1	0,001
Totais	89986	100,000%

A Figura 4.6 apresenta um gráfico para um vídeo em alta resolução. O intervalo de classe utilizado foi de 250 *bytes*.

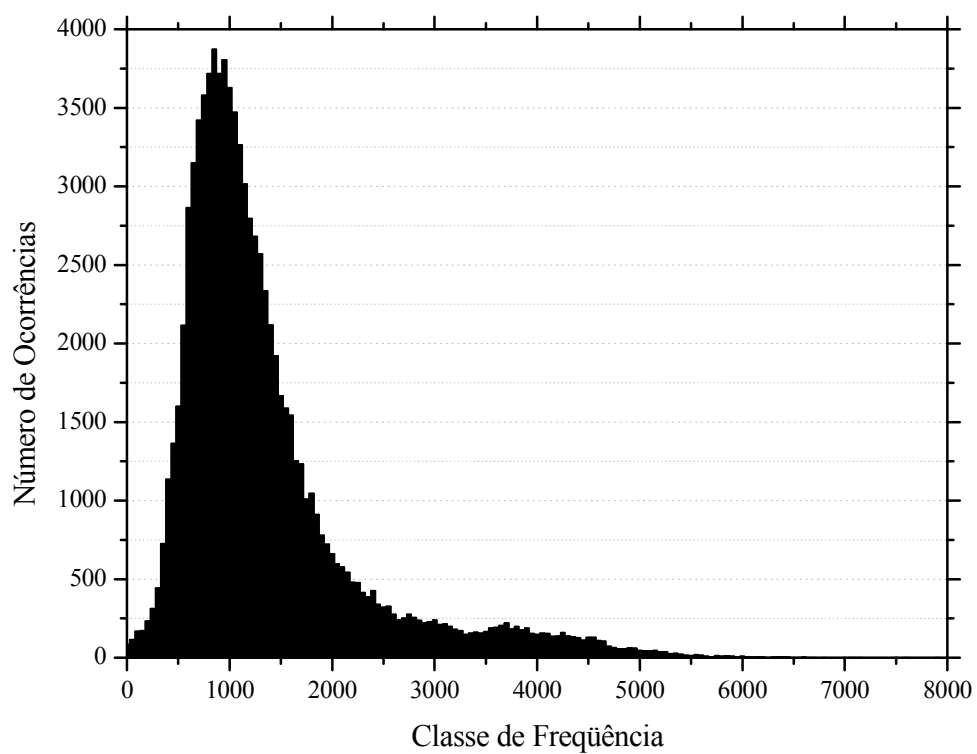


Figura 4.6: Distribuição de frequência para um vídeo em alta resolução.

Na Tabela 4.5 apresenta-se que 58% dos tamanhos dos quadros chegam a 1000 *bytes* para o arquivo em alta resolução. Verifica-se que 29% dos quadros restantes estão distribuídos na classe de frequência entre 1000 *bytes* e 2000 *bytes*. Isso naturalmente ocorre devido às características de codificação com baixa compressão dos quadros em alta resolução.

Tabela 4.5: Percentagem de agrupamento por classe de frequência em alta resolução.

<b>Intervalo</b>	<b>Ocorrências</b>	<b>(%)</b>
1	766	0,851
2	3973	4,414
3	13145	14,606
4	18686	20,763
5	16164	17,960
6	11624	12,916
7	7282	8,091
8	4462	4,958
9	2852	3,169
10	2041	2,268
11	1408	1,564
12	1215	1,350
13	1035	1,150
14	789	0,877
15	968	1,076
16	894	0,993
17	726	0,807
18	665	0,739
19	537	0,597
20	291	0,323
21	209	0,232
22	124	0,138
23	53	0,059
24	38	0,042
25	21	0,023
26	14	0,016
27	5	0,006
28	1	0,001
29	3	0,003
30	0	0
31	1	0,001
32	1	0,001
33	1	0,001
34	3	0,003
Totais	89998	100,000%

A distribuição de densidade de probabilidade para o vídeo de alta resolução está apresentada na Figura 4.7.

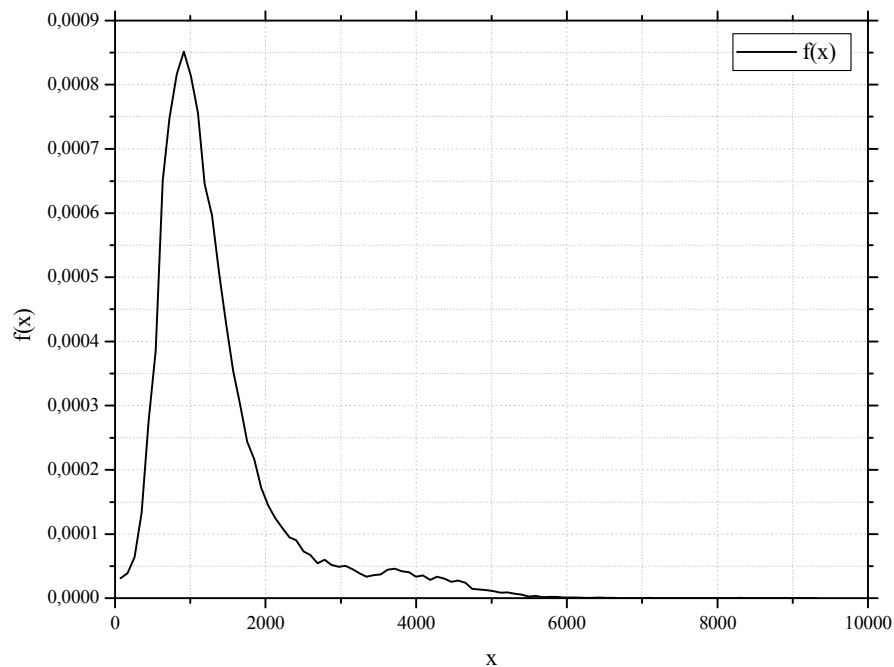


Figura 4.7: Função de densidade de probabilidade para vídeo em alta resolução.

A função distribuição cumulativa é obtida através da função densidade de probabilidade, como definido na equação (2.7). A função de distribuição cumulativa para o vídeo de alta resolução está apresentada na Figura 4.8. A distribuição de densidade de probabilidade para o vídeo de média resolução está apresentada na Figura 4.9.

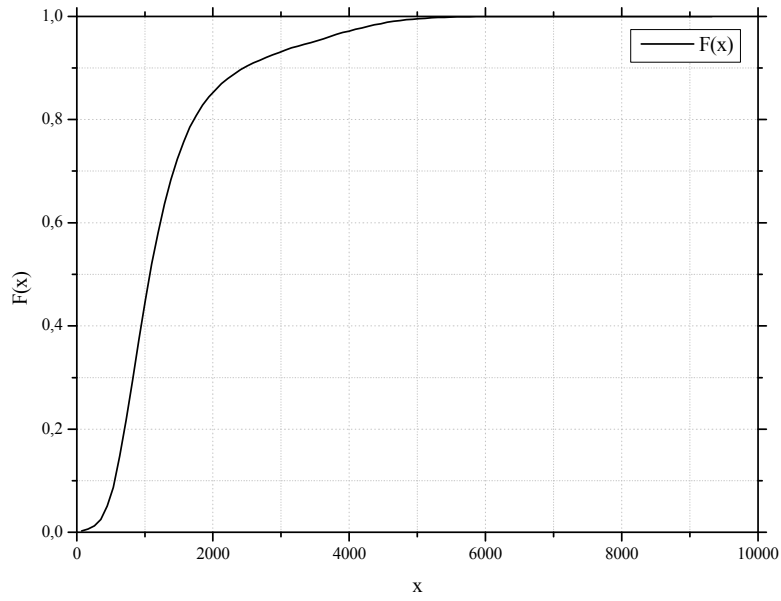


Figura 4.8: Função de distribuição cumulativa para vídeo em alta resolução.

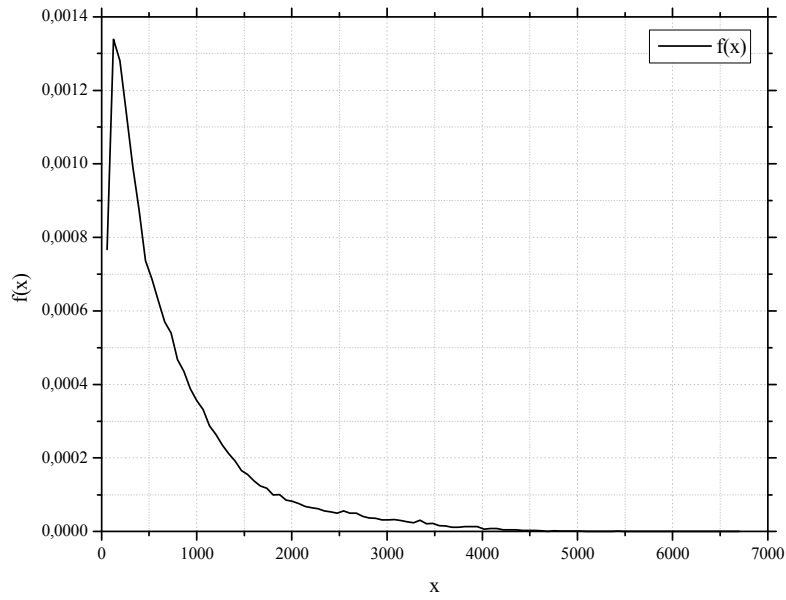


Figura 4.9: Função de densidade de probabilidade para vídeo em média resolução.

A função de distribuição cumulativa, para o vídeo de média resolução, está apresentada na Figura 4.10. A distribuição de densidade de probabilidade, para o vídeo de baixa resolução, está apresentada na Figura 4.11.

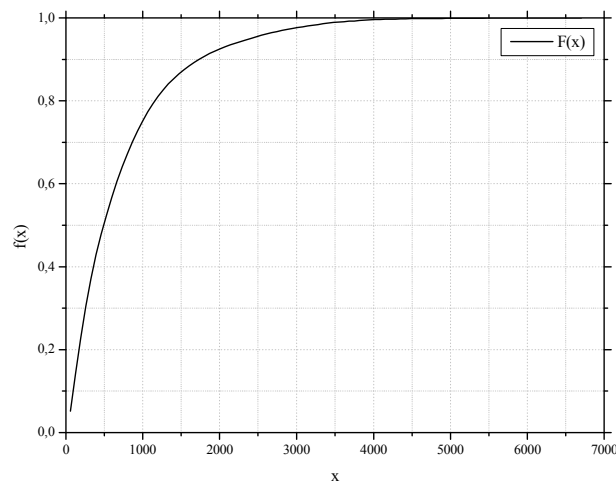


Figura 4.10: Função de distribuição cumulativa para vídeo, em média resolução.

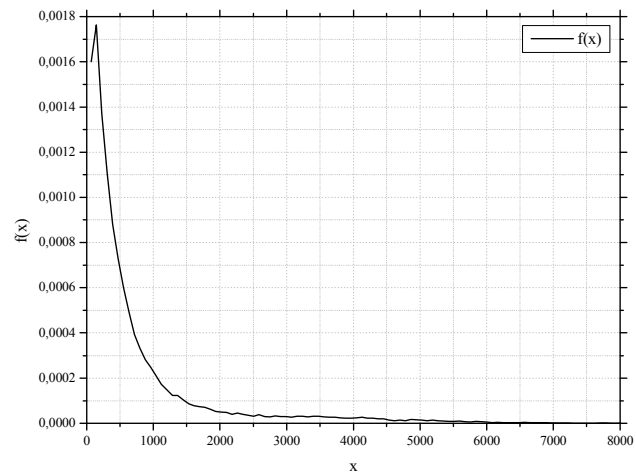


Figura 4.11: Função de densidade de probabilidade para vídeo em baixa resolução.

A função de distribuição cumulativa para o vídeo de baixa resolução está apresentada na Figura 4.12.

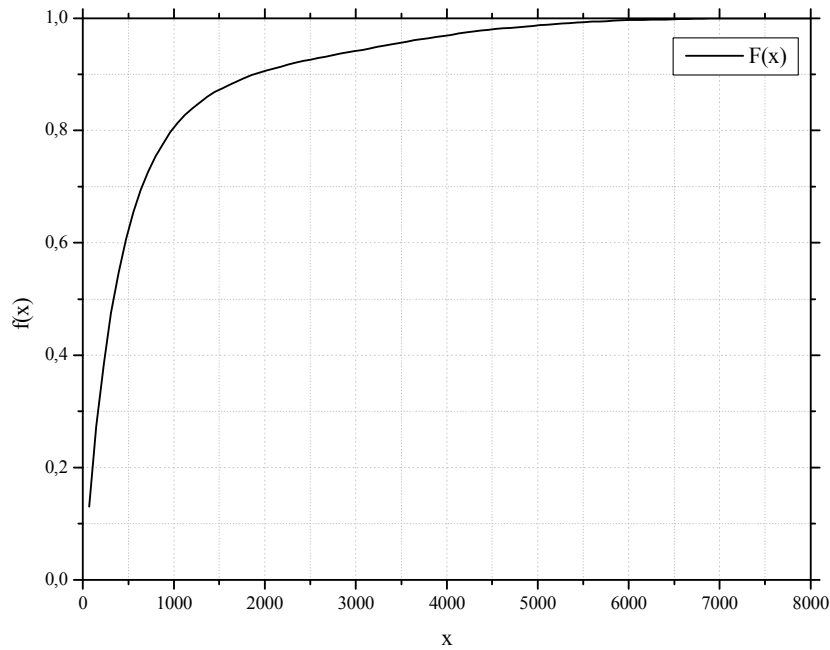


Figura 4.12: Função de distribuição cumulativa para vídeo em baixa resolução.

Outra característica a considerar, para cada tipo de arquivo, o valor máximo e médio de tamanho dos quadros aumenta proporcionalmente, à medida que se aumenta a resolução do vídeo. Essa característica está presente nos valores calculados das médias das amostras avaliadas.

A variância mede o grau de dispersão dos quadros em relação às outras resoluções. As resoluções dos vídeos apresentam coeficientes de variância diferentes em função das suas propriedades estatísticas. A Tabela 4.6 mostra a variação em números da média e da variância, para os diferentes tipos de resolução.



Tabela 4.6: Estatísticas dos quadros obtidas para os vídeos em baixa, média e alta resolução.

<b>Resolução</b>	<b>Média</b> ( $\mu \times 10^3$ ) bytes	<b>Variância</b> ( $\sigma^2 \times 10^6$ ) bytes	<b>cov</b> ( $\sigma/\mu$ ) bytes	<b>Pico/Média</b> ( $X_{\max}/\mu$ ) bytes
<b>Baixa</b>	0,76861	1,13836	1,38813	10,60869
<b>Média</b>	0,77481	0,57674	0,98017	8,69252
<b>Alta</b>	1,37641	0,81628	0,65641	6,80755

A mesma análise estatística foi realizada isoladamente no vídeo *Star Wars IV* em alta resolução, produzindo as estatísticas apresentadas na Tabela 4.7. No entanto, foram isolados os valores das médias e das variâncias, para todos os quadros, e para cada tipo de quadro no GOP.

Tabela 4.7: Estatísticas por tipo de quadro obtido para os arquivos em baixa, média e alta resolução.

<b>Resolução</b>	<b>Quadro</b>	<b>Média</b> ( $\mu \times 10^3$ ) bytes	<b>Variância</b> ( $\sigma^2 \times 10^6$ ) bytes	<b>cov</b> ( $\sigma/\mu$ ) bytes	<b>Pico / Média</b> ( $X_{\max}/\mu$ ) bytes
<b>Alta</b>	<b>I</b>	3,42473	1,01868	0,29471	2,31054
<b>Alta</b>	<b>P</b>	1,55134	0,62416	0,50929	6,03991
<b>Alta</b>	<b>B</b>	1,05488	0,22377	0,44843	6,45000
<b>Média</b>	<b>I</b>	2,21342	0,89688	0,42781	2,81962
<b>Média</b>	<b>P</b>	0,91768	0,47912	0,75427	7,33912
<b>Média</b>	<b>B</b>	0,54143	0,25263	0,92833	11,68206
<b>Baixa</b>	<b>I</b>	3,64596	1,72063	0,35977	2,23644
<b>Baixa</b>	<b>P</b>	0,96686	0,44366	0,68890	6,49524
<b>Baixa</b>	<b>B</b>	0,33466	0,08824	0,88764	8,98116

Cada tipo de quadro possui um conjunto de informações de diferente complexidade, para representar o fluxo de vídeo. Quanto maior a complexidade, menor a compressão obtida. Essa mesma característica ocorre em todas as outras resoluções dos demais vídeos avaliados.

A Figura 4.13 apresenta um gráfico da autocorrelação (2.32) calculada para o vídeo, independente de tipo de quadro para um vídeo de baixa resolução.

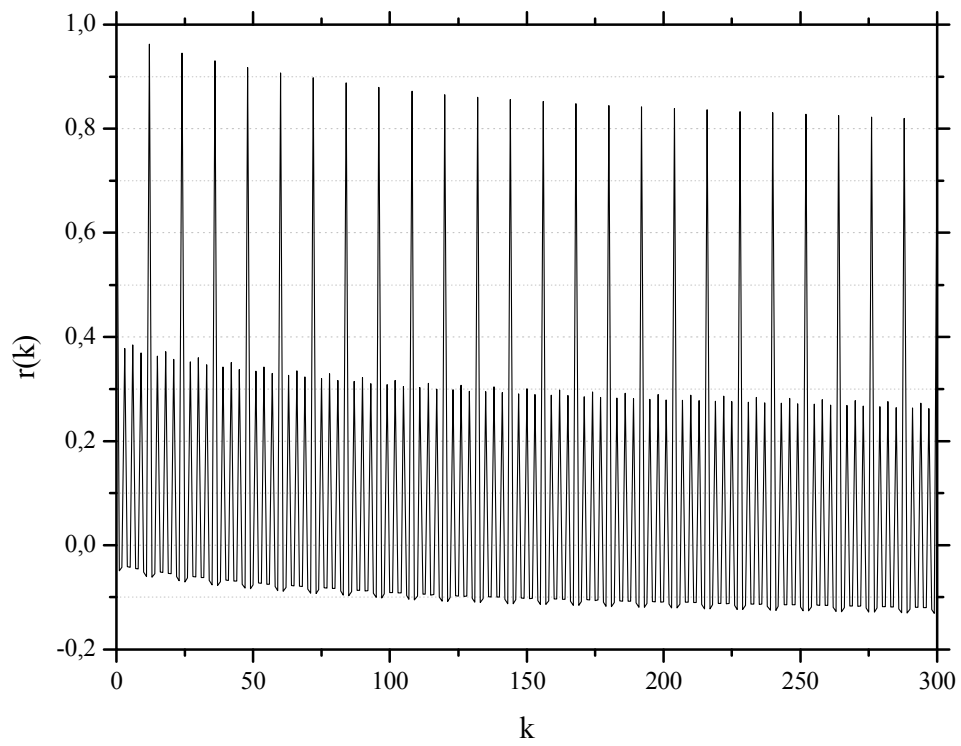


Figura 4.13: Autocorrelação para um vídeo de baixa resolução.

A Figura 4.14 apresenta um gráfico de autocorrelação, independente de tipo de quadro para um vídeo de média resolução.

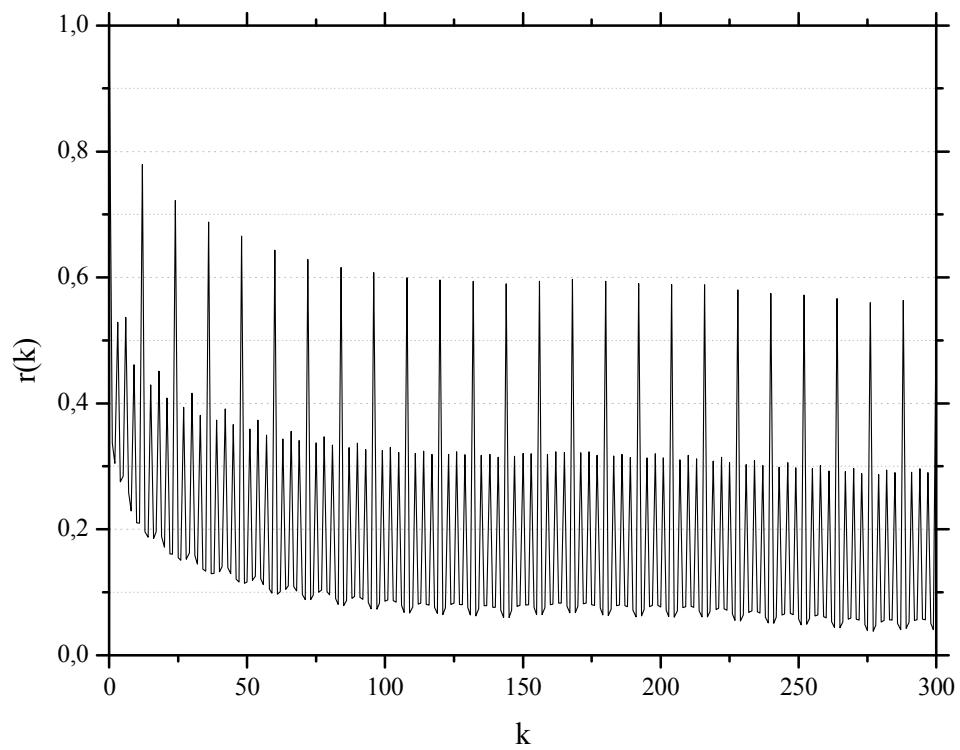


Figura 4.14: Autocorrelação para um vídeo de média resolução.

A Figura 4.15 apresenta um gráfico de autocorrelação, independente de tipo de quadro para um vídeo em alta resolução.

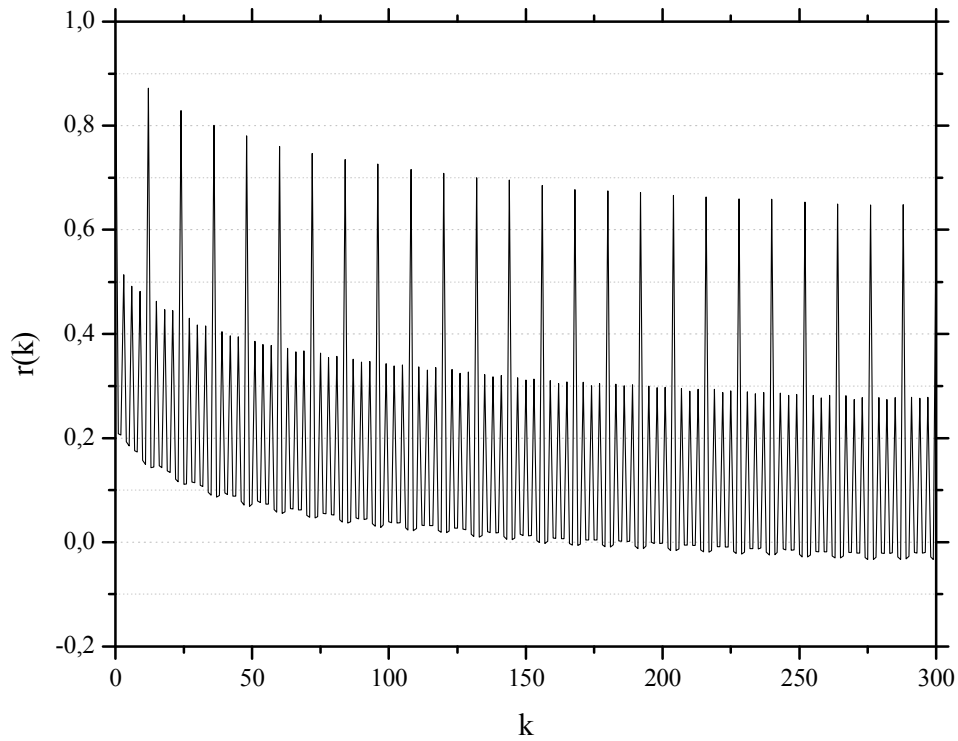


Figura 4.15: Autocorrelação para um vídeo de alta resolução.

Os valores máximos da autocorrelação representam os quadros do tipo I, os quais possuem os maiores valores de média acumulada e de variância, como apresentado na Tabela 4.7. Da mesma maneira, identifica-se que os valores intermediários representam os quadros do tipo P e os valores mínimos estão representando os quadros do tipo B. Essas figuras representam visivelmente a diferença dos tamanhos dos quadros em relação à estrutura do GOP.

O declínio da curva do gráfico de autocorrelação é uma característica da dependência de longa duração, como apresentado em [FIT01]. Ao se considerar independentemente os

valores dos quadros I, P e B em qualquer vídeo, os valores da autocorrelação também serão independentes, em virtude das diferenças das informações contidas em cada tipo de quadro.

A Figura 4.16 apresenta o gráfico do vídeo *Jurassic Park* em baixa resolução, onde foram calculados os valores da função de autocorrelação, para cada tipo de quadro isoladamente.

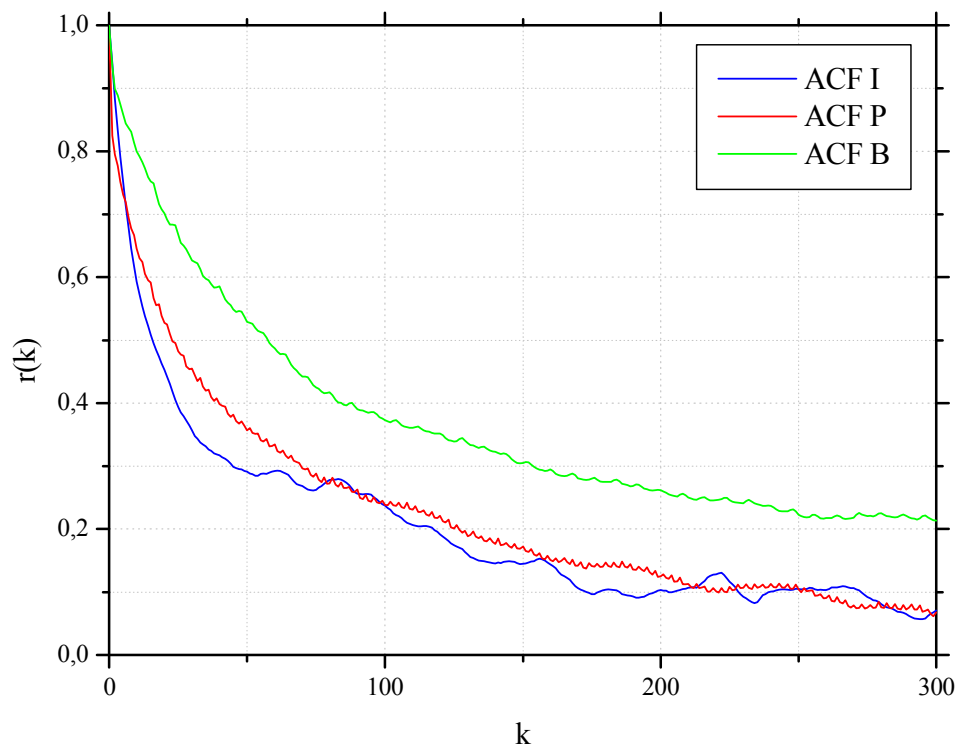


Figura 4.16: Autocorrelação do vídeo *Jurassic Park*, em baixa resolução, por tipo de quadro.

Na Figura 4.16, os quadros do tipo I e P possuem valores na curva de autocorrelação próximos. O declínio da curvatura do gráfico da autocorrelação indica a presença de auto-similaridade. A curva de autocorrelação dos quadros do tipo B, não se aproxima aos outros tipos de quadros. No entanto, a diferença na autocorrelação para os demais tipos de quadro, indica que existe um diferente grau de similaridade entre os quadros de diferentes tipos.

A Figura 4.17 apresenta o gráfico do vídeo *Alladin* em média resolução, onde foram calculados os valores da função de autocorrelação para cada tipo de quadro.

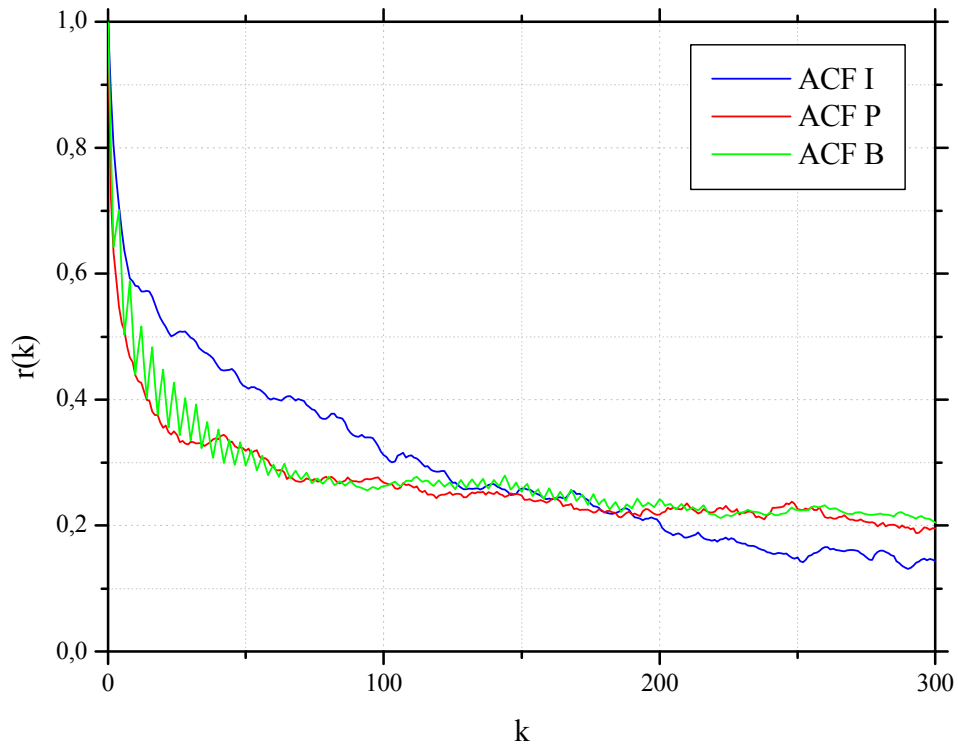


Figura 4.17: Autocorrelação do vídeo *Alladin*, em média resolução, por tipo de quadro.

Na Figura 4.17 os quadros do tipo P e B possuem valores de autocorrelação próximos. O mesmo não ocorre na Figura 4.16, na qual a curva com menor grau de similaridade é a do quadro do tipo B, e as curvas de autocorrelação com maior similaridade são as curvas dos quadros I e P.

A Figura 4.18 apresenta o gráfico do vídeo *Star Wars IV* em alta resolução, onde foram calculados os valores da função de autocorrelação para cada tipo de quadro.

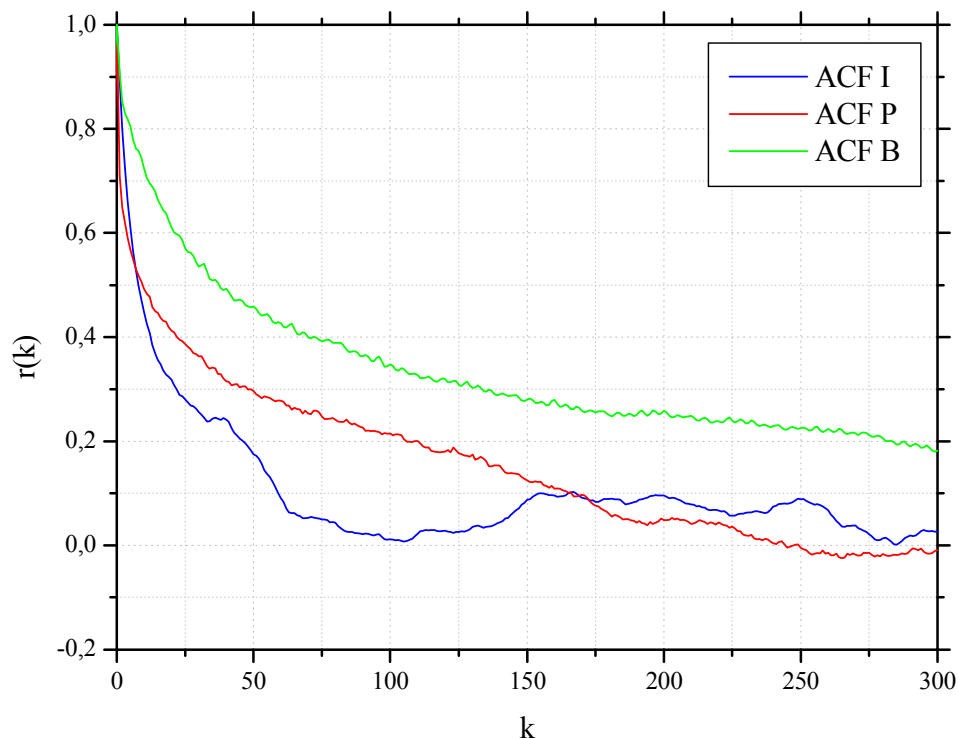


Figura 4.18: Autocorrelação do vídeo *Star Wars IV*, em alta resolução, por tipo de quadro.

Na Figura 4.18 todas as curvas de autocorrelação para quadros do tipo I, P e B não apresentam valores de autocorrelação próximos ou equivalentes. Apesar das curvas seguirem um padrão de autocorrelação de processos auto-similares, elas indicam que os quadros possuem independência entre si. Essa característica é considerada na modelagem de tráfego de vídeo, uma vez que a similaridade na autocorrelação para cada tipo de quadro é avaliada independentemente.

Os valores calculados da autocorrelação para cada tipo de quadro, e para diferentes resoluções de vídeo, também apresentaram diferenças. Essas diferenças são apresentadas nos gráficos da autocorrelação, ao considerar a análise estatística dos arquivos em baixa, média e

alta resolução. A Figura 4.19 apresenta os valores calculados da autocorrelação para os quadros do tipo I, presentes nos arquivos de vídeo em baixa, média e alta resolução.

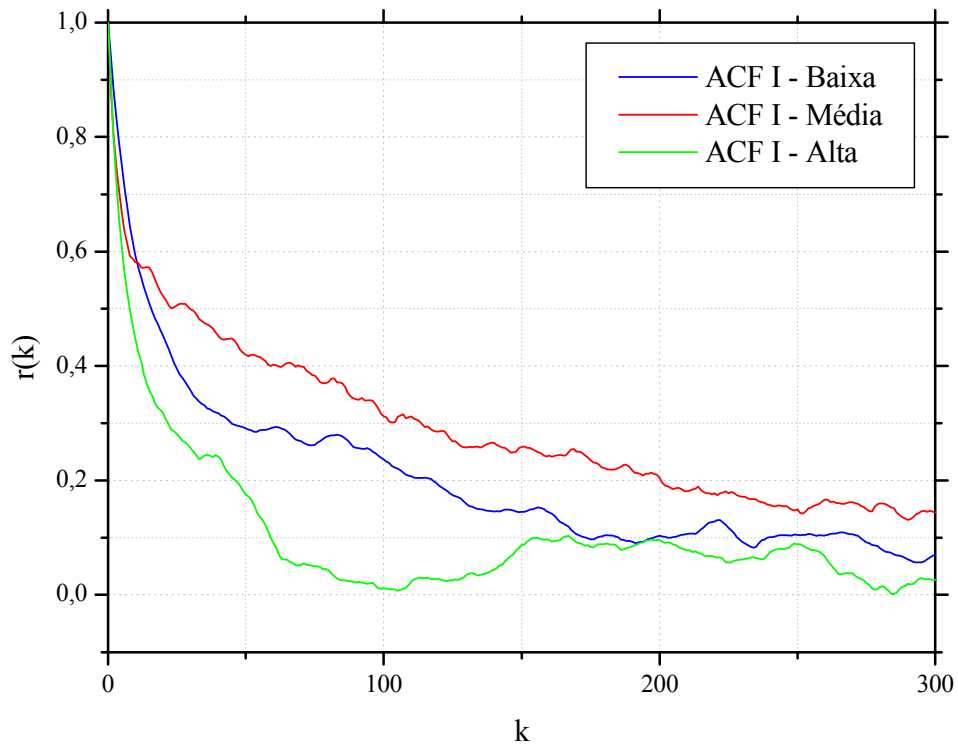


Figura 4.19: Autocorrelação de quadros do tipo I nos arquivos de alta, média e baixa resolução.

As curvas presentes na Figura 4.19 indicam a independência nos valores da autocorrelação, para o mesmo tipo de quadro em vídeos de diferentes resoluções. No entanto, os valores de autocorrelação das curvas são próximos em relação às demais resoluções.



A Figura 4.20 apresenta os valores calculados da autocorrelação para os quadros do tipo P presente nos arquivos de vídeo em baixa, média e alta resolução.

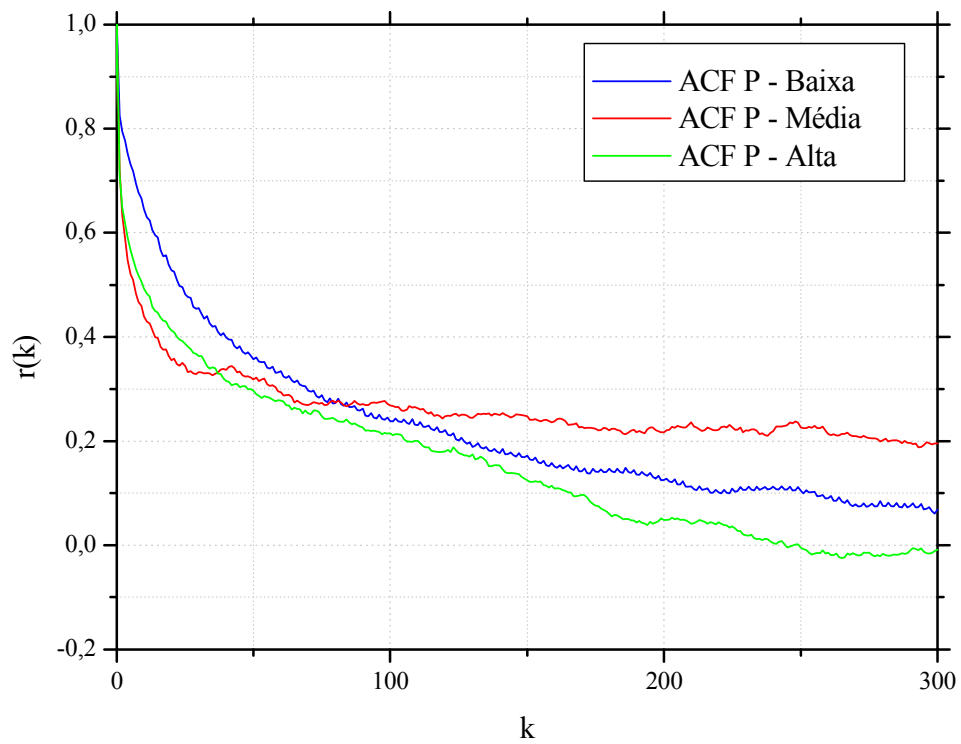


Figura 4.20: Autocorrelação de quadros do tipo P nos arquivos de alta, média e baixa resolução.

A Figura 4.21 apresenta os valores calculados da autocorrelação para os quadros do tipo B, presentes nos arquivos de vídeo em baixa, média e alta resolução.

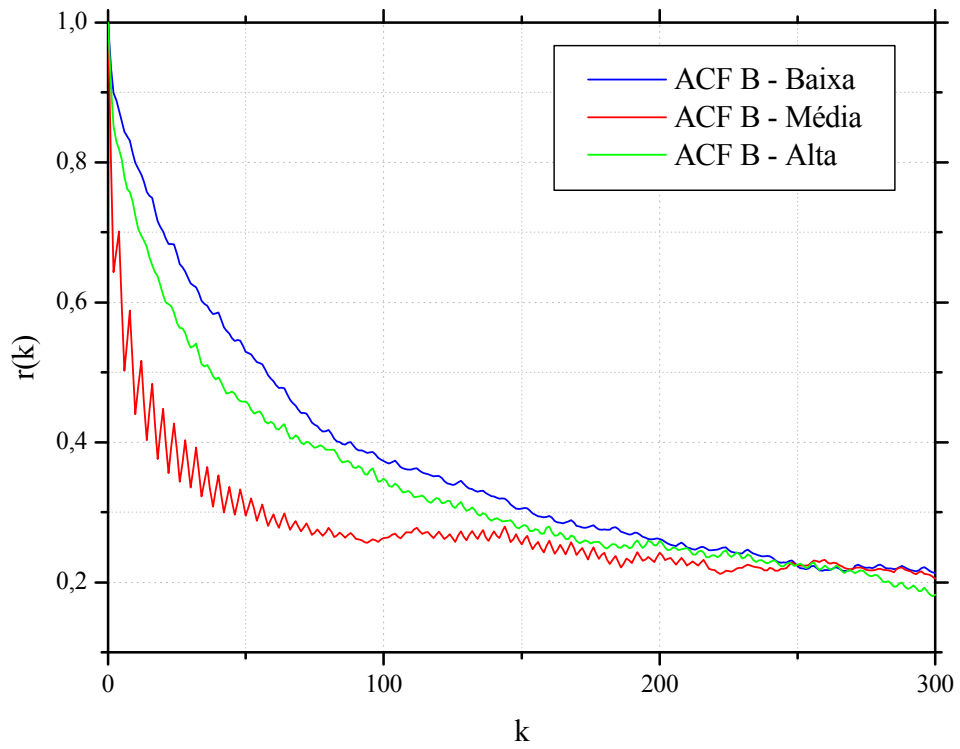


Figura 4.21: Autocorrelação de quadros do tipo B nos arquivos de alta, média e baixa resolução.

### 4.3. Análise da Auto-Similaridade

As próximas avaliações implicarão em verificar o coeficiente de auto-similaridade entre os tipos de quadros dos vídeos analisados em baixa, média e alta resolução. A auto-similaridade será mensurada através do método de estimativa de Hurst (2.68), apresentado na Seção 2.7.3.

O método de estimativa de Hurst identifica os coeficientes, através da análise logarítmica da estatística  $R/S$ , definida na equação (2.67). O gráfico do  $\log(k)$ , em função do  $\log(R/S)$ , aproxima a estimativa do parâmetro de similaridade Hurst.

A Figura 4.22 apresenta o valor estimado do coeficiente de Hurst, para o vídeo *Star Wars IV*, em alta resolução. Para todos os tipos de quadros, a Figura 4.22 apresenta os resultados de  $\log(k)$ , em função de  $\log(R/S)$ , com aproximação linear dos valores de estimativa.

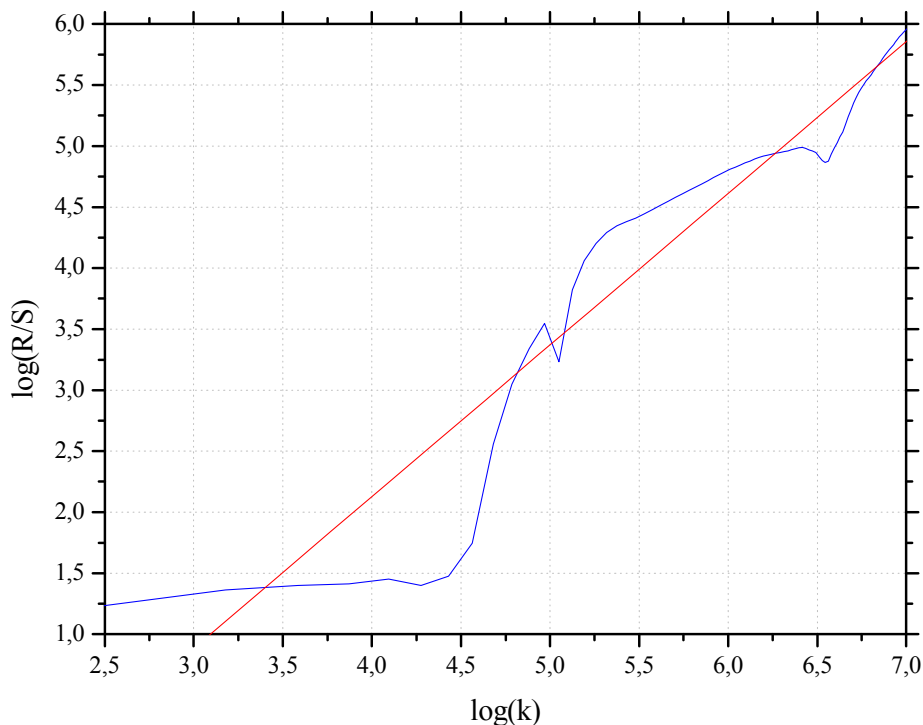


Figura 4.22: Estimativa de Hurst em função da  $\log(R/S)$  e  $\log(k)$ , de um vídeo em alta resolução.

A Figura 4.23, Figura 4.24 e Figura 4.25 apresentam o valor estimado do coeficiente de Hurst para cada quadro do GOP, no vídeo *Star Wars IV* em alta resolução. Os valores são calculados através da equação (2.68), com aproximação linear, para apresentar visualmente a presença da dependência de longa duração.

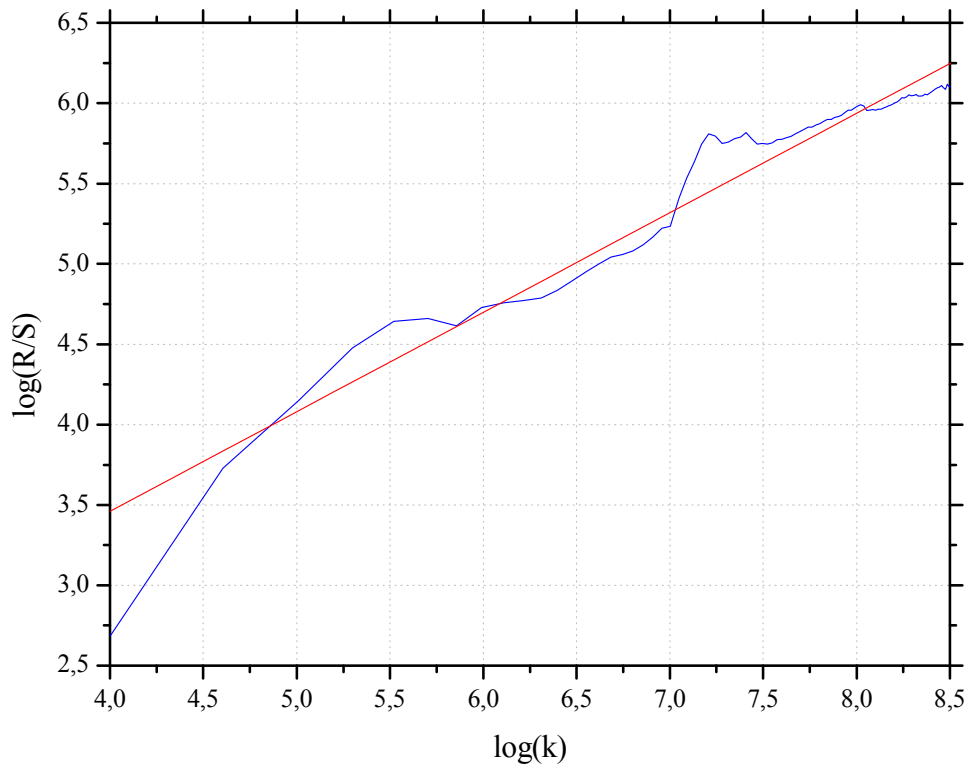


Figura 4.23: Estimativa de Hurst em função da  $\log(R/S)$  e  $\log(k)$  para os quadros I de um vídeo em alta resolução.

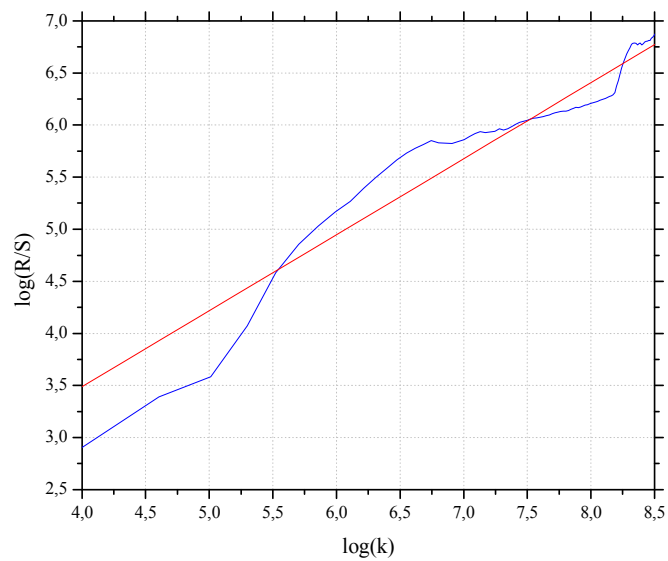


Figura 4.24: Estimativa de Hurst em função da  $\log(R/S)$  e  $\log(k)$  para os quadros P de um vídeo em alta resolução.

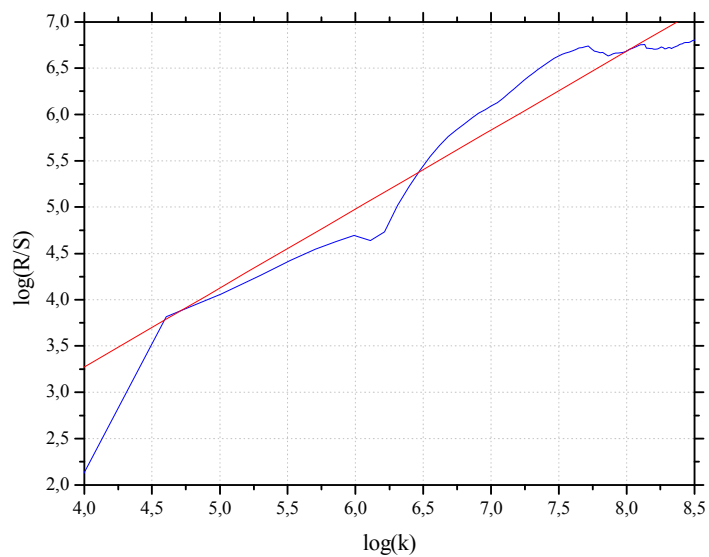


Figura 4.25: Estimativa de Hurst em função da  $\log(R/S)$  e  $\log(k)$  para os quadros B de um vídeo em alta resolução.

A Seção 2.7 apresentou as características e as propriedades dos processos auto-similares. Os valores de estimativa do coeficiente de Hurst variam durante todo o intervalo dos quadros analisados. Essa característica está presente em todas as análises realizadas para os vídeos em média e baixa resolução. Devido a essa característica, os modelos de simulação devem considerar o valor de Hurst relacionado com o tamanho do intervalo de quadros que se deseja simular.

Para se obter a estimativa de Hurst, deve ser calculada a média dos valores obtidos no gráfico. A média dos valores apresentados nos gráficos de  $\log(R/S)$  e  $\log(k)$  caracteriza um valor de tendência central que se pode utilizar nas simulações dos quadros. Dessa forma, os valores das médias calculadas para a estimativa de Hurst, se aproximam dos valores encontrados nos gráficos que serão apresentados a seguir. Os valores das estimativas tendem a estabilizar em um valor médio, encontrado para qualquer estimativa em longos intervalos.

A Tabela 4.8 apresenta as estimativas médias para o coeficiente de Hurst, por tipo de quadro e por tipo de resolução. Essa estimativa considerou todos os vídeos em alta, média e baixa resolução, utilizados nesse experimento.

Tabela 4.8: Média dos valores de Estimativa de Hurst.

<b>Resolução</b>	<b>Quadro</b>	$\mu_H$
<b>Alta</b>	<b>I</b>	0,74585
<b>Alta</b>	<b>P</b>	0,80376
<b>Alta</b>	<b>B</b>	0,82944
<b>Média</b>	<b>I</b>	0,81110
<b>Média</b>	<b>P</b>	0,77207
<b>Média</b>	<b>B</b>	0,73731
<b>Baixa</b>	<b>I</b>	0,77819
<b>Baixa</b>	<b>P</b>	0,77002
<b>Baixa</b>	<b>B</b>	0,77988

A Figura 4.26 apresenta a estimativa para o coeficiente de Hurst em todo o vídeo em alta resolução, sem considerar o tipo dos quadros. O intervalo de estimativa ocorreu para um agrupamento de 500 GOPs, onde cada GOP contém um conjunto de 12 quadros.

No entanto, para a estimativa do coeficiente de Hurst, deve-se tratar a variação da estimativa no início do gráfico. O cálculo da média dos coeficientes de Hurst, a partir dos valores utilizados para o cálculo, pode desconsiderar as amostras do início do gráfico, onde as variações dos coeficientes não estão estáveis em torno de um único valor médio.

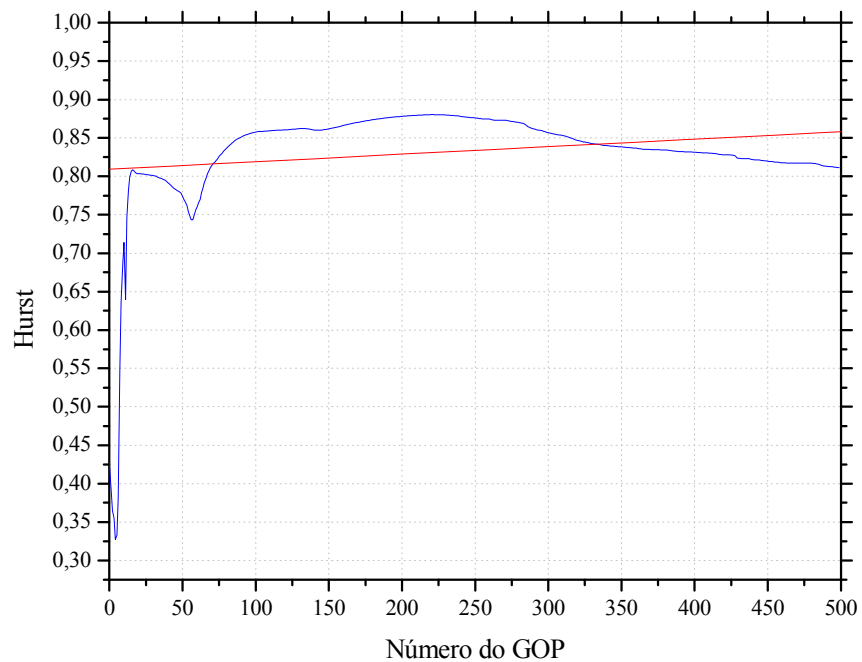


Figura 4.26: Valores de Hurst para 500 GOPs.

A Figura 4.27, a Figura 4.28 e a Figura 4.29, apresentam os gráficos de estimativa para o coeficiente de Hurst, para cada tipo de quadro. As estimativas foram realizadas a partir do cálculo com agrupamento definido em 50 quadros.

Visivelmente constata-se que quanto menor o intervalo do processo aleatório, mais próximo de  $0,5$  é o valor do coeficiente. Isso ocorre porque processos aleatórios com períodos curtos e insuficientes de observação não retratam as características de um processo auto-similar de dependência de longa duração.

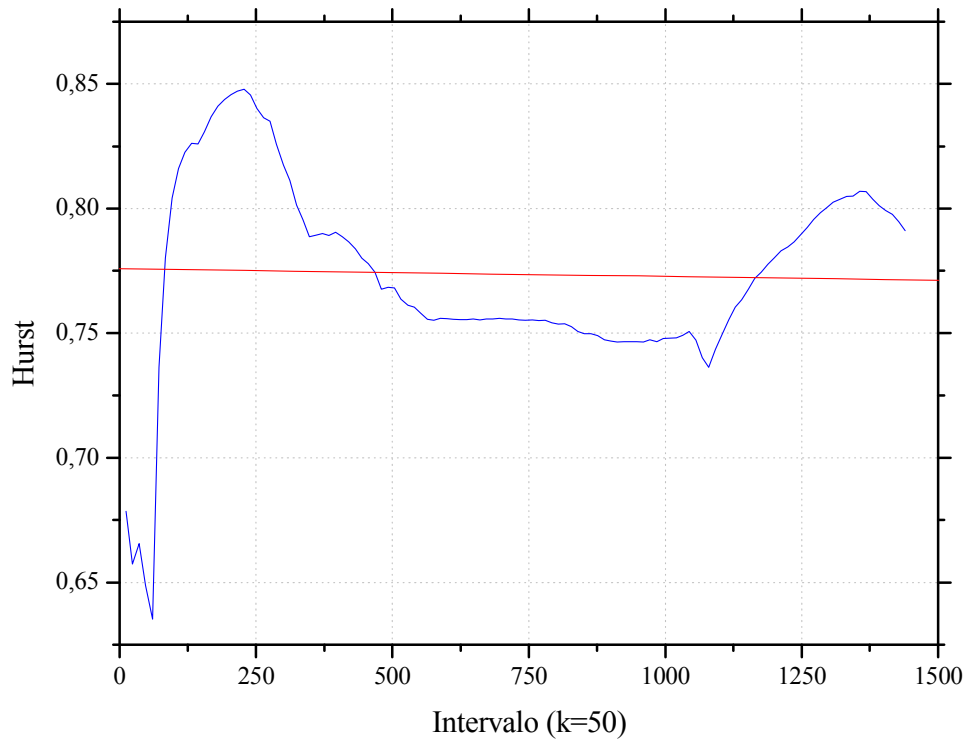


Figura 4.27: Valores de Hurst para quadros do tipo I em alta resolução.



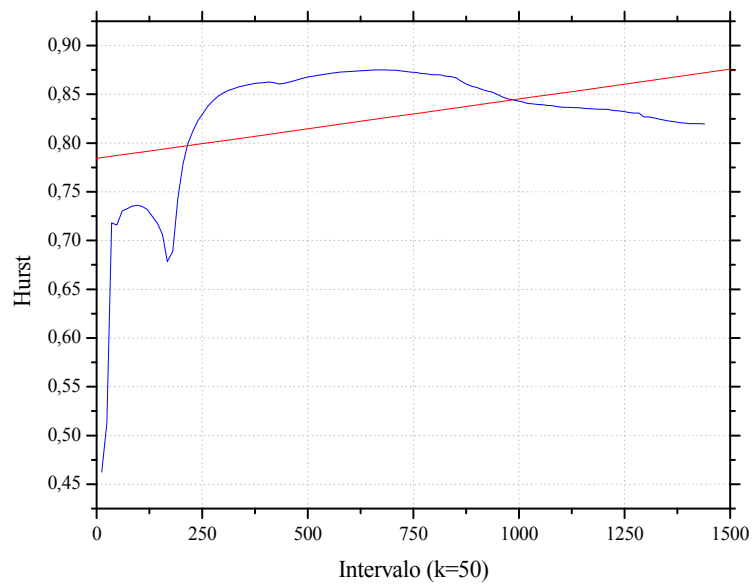


Figura 4.28: Valores de Hurst para quadros do tipo P em alta resolução.

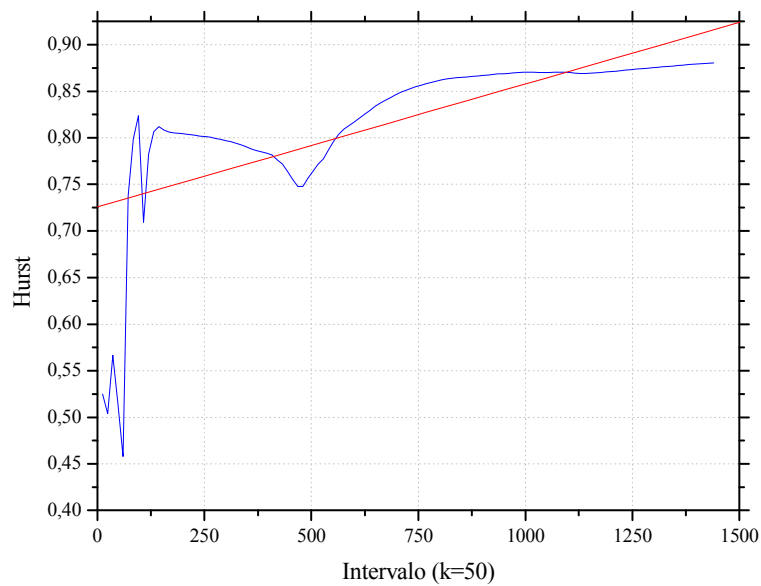


Figura 4.29: Valores de Hurst para quadros do tipo B em alta resolução.

As características de auto-similaridade, apresentadas na Seção 2.7, são evidenciadas quanto maior o tempo de observação do experimento. Essa situação indica a presença de dependência de longa duração, onde os valores de estimativa de Hurst tendem a se estabilizar, quanto maior o tempo de observação do experimento.

A Figura 4.28 e a Figura 4.29 apresentam valores da estimativa de Hurst acima de 0,5, somente a partir do 75° GOP, aproximadamente. A partir do 250° GOP, os valores tendem a se aproximar de um valor estável, devido à estabilização da variação das estimativas de Hurst. Isso já não ocorre para os quadros do tipo I, pois são os quadros do tipo I que apresentam maiores detalhes da imagem e conseqüentemente sofrem maiores variações no tamanho da imagem.

Para cada intervalo de tempo é estimado um valor diferente para o coeficiente de Hurst. Esse valor também é alterado em função da resolução do vídeo e do tipo de quadro. As constantes de Hurst, encontradas para o experimento, seguem os valores apresentados na Tabela 4.9. As estimativas estão organizadas para todas as resoluções de vídeo utilizadas nesse experimento, por tipo de quadro, de acordo com a estrutura do GOP, e por intervalo de quadros compreendidos entre 10 e 2000 quadros.

Tabela 4.9: Estimativa de Hurst, para todos os quadros em diferentes resoluções, para k intervalos.

Resolução	Quadro	Intervalo k							
		10	50	100	200	400	800	1600	2000
<b>Alta</b>	<b>I</b>	0,6143	0,6471	0,8095	0,8451	0,7893	0,7547	0,7846	0,7597
<b>Alta</b>	<b>P</b>	0,4482	0,7242	0,7366	0,7685	0,8623	0,8701	0,8115	0,7991
<b>Alta</b>	<b>B</b>	0,4767	0,4840	0,8291	0,8045	0,7831	0,8618	0,8835	0,8802
<b>Média</b>	<b>I</b>	0,5879	0,7109	0,6815	0,7298	0,7559	0,7723	0,8359	0,8435
<b>Média</b>	<b>P</b>	0,5751	0,6631	0,6171	0,7466	0,6899	0,7048	0,6951	0,7010
<b>Média</b>	<b>B</b>	0,4609	0,5584	0,6999	0,7314	0,8016	0,7999	0,7917	0,7368
<b>Baixa</b>	<b>I</b>	0,6037	0,7778	0,7917	0,7274	0,7786	0,8024	0,7882	0,8052
<b>Baixa</b>	<b>P</b>	0,5045	0,6815	0,8364	0,7989	0,7750	0,7459	0,7846	0,7674
<b>Baixa</b>	<b>B</b>	0,4874	0,5545	0,6424	0,8079	0,8291	0,7953	0,7847	0,7878

#### 4.4. Conclusões

Ao aplicar a fórmula de autocorrelação apresentada em (2.13) em um GOP composto por 12 quadros de estrutura IBBPBBPBBPBB, os valores da autocorrelação variam de acordo com o tamanho desses quadros, como consequência natural da diferença da própria complexidade da imagem codificada no GOP.

Para a modelagem de tráfego de vídeo, a autocorrelação representa a relação entre a seqüência dos tamanhos dos quadros, em intervalos de tempos distintos. Essa relação apresenta a similaridade entre os quadros, característica essa que deve estar presente na geração dos valores aleatórios, de acordo com as definições das propriedades estatísticas utilizadas no modelo. Essa propriedade é fundamental para a avaliação de processos auto-similares, em observação ao comportamento do modelo de simulação.

O coeficiente de Hurst possui característica importante nos cálculos dos valores da função de autocorrelação. O seu valor representa a variação das curvas dos gráficos apresentados. Esses valores podem assumir diferentes proporções ao se considerar o tipo de quadro e o tipo de resolução do vídeo.

Dado o valor para cálculo da autocorrelação a partir do coeficiente de Hurst, é possível modelar um tráfego de vídeo baseado nas características das imagens associadas ao vídeo em questão. Outra característica da utilização do coeficiente de Hurst é representar a melhor aproximação na simulação dos valores relacionados em função do tempo. Isso também ocorre na aproximação dos valores com maior relação ao tipo de quadro e resolução do vídeo simulado.



## Capítulo 5

### Modelagem de Tráfego de Vídeo

Uma modelagem de processos aleatórios requer a observação de fenômenos reais encontrados na natureza. Estatisticamente avalia-se a relação entre os valores de característica aleatória, baseando-se na coleta desses valores, durante a observação. Os resultados da análise estatística possibilitam construir modelos de simulação do fenômeno observado, a partir da geração de valores aleatórios e de considerável relação com os valores reais encontrados no experimento.

#### 5.1. Métodos de Modelagem

A modelagem de tráfego de vídeo foi desenvolvida para os experimentos descritos nesse documento, através de algoritmos de geração de dados aleatórios em comparação às funções de autocorrelação teórica. Os métodos de modelagem utilizam o processo aleatório auto-similar, como proposto em [ANS02], para aproximação dos dados empíricos de vídeo real.

Comparações da autocorrelação de vídeos reais e de vídeos simulados foram analisadas nesse experimento, com a finalidade de validar se a relação das amostras geradas segue o modelo estatístico definido para um vídeo real. Os valores modelados devem seguir a aproximação de autocorrelação teórica, como por exemplo, a fórmula matemática recursiva utilizada em [ANS02], para aproximar os resultados do cálculo da função de autocorrelação definida por  $r(k)$ .

Como proposto em [ANS02], as fórmulas teóricas para aproximação da autocorrelação de processos aleatórios auto-similares (5.1), Markov (5.2) e  $M/G/\infty$  (5.3), são utilizadas para apresentar gráficos teóricos da autocorrelação e para comparação da autocorrelação resultante das simulações geradas a partir do modelo de tráfego de vídeo.

$$r(k) = k^{-\beta} \quad (5.1)$$

$$r(k) = e^{-k\beta} \quad (5.2)$$

$$r(k) = e^{-\sqrt{k}\beta} \quad (5.3)$$

A equação (5.1) é uma especialização da equação (2.54), onde se considera  $\alpha = 1$  e é a fórmula definida para autocorrelação teórica para processos auto-similares. Considerando o valor definido de  $\beta$  pela fórmula (2.57) de transformação em função de Hurst, ou seja,  $\beta = 2 - (2 \cdot H)$ , a Figura 5.1 apresenta o gráfico da autocorrelação teórica das funções (5.1), (5.2) e (5.3) em comparação com a autocorrelação calculada para o arquivo de vídeo *Star Wars IV* em alta resolução, com constante de Hurst definida por 0,7480, estimada a partir de um intervalo de 1000 quadros:

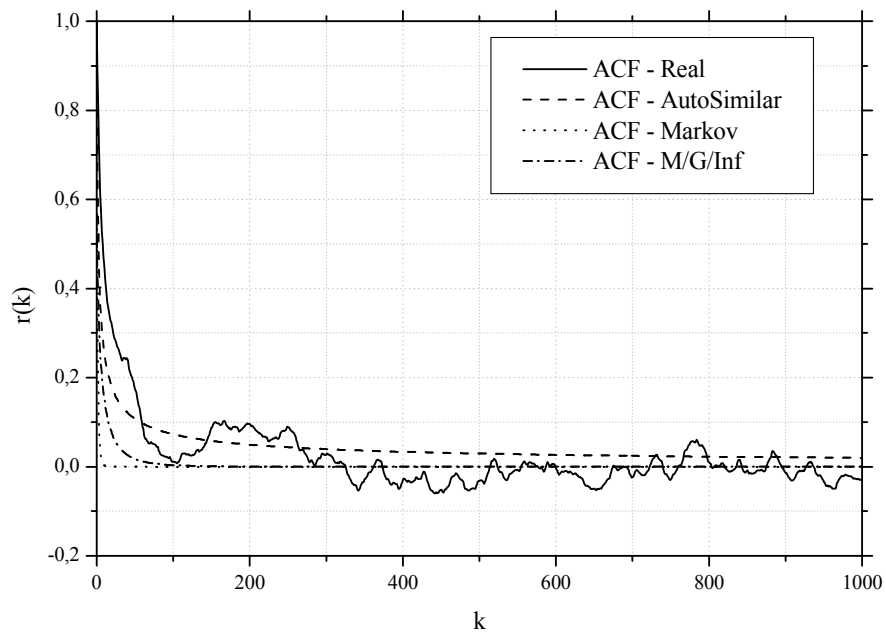


Figura 5.1: Autocorrelação real em comparação com os valores das funções de autocorrelação teóricas.

Como apresentado em [ANS02], a curva teórica da autocorrelação, para o processo auto-similar, é o que mais se aproxima da curva de autocorrelação, gerada a partir dos dados empíricos de um vídeo real. Isso deve ocorrer para qualquer uma das simulações dos quadros do tipo I, P ou B.

## 5.2. Modelagem de Processos Auto-Similares

Essa seção apresenta as propostas do método F-ARIMA para modelagem de processos auto-similares. As propostas de modelagens baseadas no algoritmo F-ARIMA [GAR94], [ANS02] e [QIN94] serão analisadas para o modelo de tráfego de vídeo hierárquico.

A implementação da função F-ARIMA (do inglês, *Fractional Autoregressive Integrated Moving Average*) representa o comportamento de um processo auto-similar como apresentado em [PRA96]. Ao considerar o tráfego de vídeo como um processo auto-similar [PRA96], este método de modelagem aproxima os resultados da simulação em função da autocorrelação teórica definida para o processo aleatório.

No final dessa seção, será apresentado o método utilizado para estimar o coeficiente de Hurst por tipo de quadro e por resolução de vídeo, assim como serão apresentados os valores das estimativas obtidas para os arquivos de vídeos apresentados em [FIT01].

### 5.2.1. F-ARIMA – Algoritmo da Modelagem de Tráfego

Em [GAR94] e [ANS02] é apresentado o algoritmo F-ARIMA, que será utilizado para modelar os vídeos MPEG. Os resultados estatísticos das simulações serão avaliados, bem como as respectivas funções de autocorrelação. As implementações do algoritmo F-ARIMA utilizadas nessas referências serão apresentadas nessa seção.

O algoritmo F-ARIMA, utilizado para modelagem e simulação de quadros de vídeo, utiliza a função de geração de números aleatórios  $N(m_k, v_k)$ , onde essa função é definida por uma distribuição normal geradora de valores aleatórios. O valor inicial de  $X_0$  é escolhido pela função de distribuição normal  $N(0, v_0)$ , onde são definidos  $N_0 = 0$  e  $D_0 = 1$ . A geração dos quadros é resultado da iteração de  $k = 1, 2, \dots, n$ , onde para cada iteração, os seguintes procedimentos devem ser calculados:

$$N_k = r(k) - \sum_{j=1}^{k-1} \phi_{k-1,j} r(k-j) \quad (5.4)$$

$$D_k = D_{k-1} - N_{k-1}^2 / D_{k-1} \quad (5.5)$$

$$\phi_{kk} = N_k / D_k \quad (5.6)$$

$$\phi_{kj} = \phi_{k-1,j} - \phi_{kk} \phi_{k-1,k-j}, \text{ para } j = 1, 2, \dots, k-1 \quad (5.7)$$

$$m_k = \sum_{j=1}^k \phi_{kj} X_{k-j} \quad (5.8)$$

$$v_k = (1 - \phi_{kk}^2) v_{k-1} \quad (5.9)$$

$$X_k = N(m_k, v_k) \quad (5.10)$$

A equação (5.4) utiliza a aproximação da autocorrelação definida por  $r(k)$ . A aproximação da autocorrelação deve considerar a equação proposta em [ANS02], onde o cálculo da autocorrelação é definido em (5.12), com caso particular (5.11), quando  $k = 0$ .

$$r(0) = 1 \quad (5.11)$$

$$r(k+1) = \frac{k+d}{k+1} r(k) \quad (5.12)$$

O valor  $d$  utilizado em (5.12) é obtido em função do coeficiente de Hurst, definido para a simulação e referenciado na equação (5.13).

$$d = H - 0,5 \quad (5.13)$$

O coeficiente de Hurst e o respectivo valor  $d$  podem assumir valores distintos para cada vídeo analisado como proposto em [GAR94]. Os ajustes desses valores seguem os valores da análise de dados empíricos coletados no processamento de vídeos reais.

A equação (5.12) é proposta por [ANS02] como uma equação de aproximação dos valores de autocorrelação utilizados no algoritmo F-ARIMA de simulação. A necessidade de



adaptar a fórmula matemática proposta em (5.12) na equação proposta (5.14), deve-se ao fato de satisfazer à condição (5.11) para delimitar o cálculo recursivo e não permitir que o cálculo ocorra indefinidamente.

$$r(k) = \frac{k+d}{k+1} \cdot r(k-1) \quad \therefore \quad r(0) = 1 \quad (5.14)$$

Dada uma entrada  $k$ , calcula-se recursivamente a autocorrelação aproximada. A distribuição dos dados, para esse modelo, segue a distribuição normal de Gauss [ANS02]. A aproximação recursiva, utilizada no cálculo da autocorrelação, define a complexidade do algoritmo em relação à carga de tempo e espaço.

A Figura 5.2 abaixo, apresenta a complexidade do algoritmo, definida como  $o(n^2)$  para gerar  $n$  valores. O gráfico apresenta o número de intervalos a processar em função da aproximação exponencial do tempo de processamento em segundos.

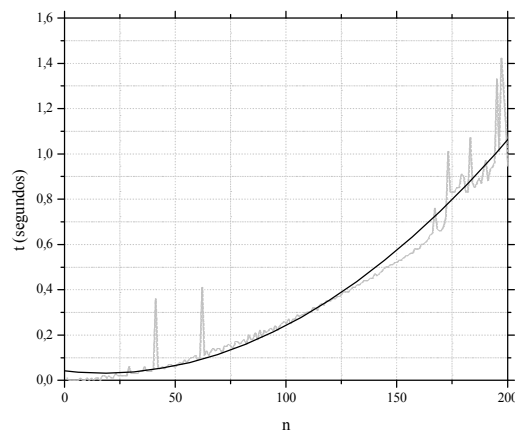


Figura 5.2: Complexidade do algoritmo F-ARIMA – Recursividade.

### 5.2.2. F-ARIMA – Adaptação Combinatória

Nessa seção será apresentado o algoritmo F-ARIMA proposto em [QIN94] para modelagem e simulação com adaptação combinatória. Em [QIN94] esse método é caracterizado como método F-ARIMA direto, o qual depende exclusivamente em função do coeficiente de Hurst definido para a simulação.

Como no método de modelagem e simulação apresentado na Seção 5.2.1, o algoritmo utiliza a função de distribuição normal para geração de números aleatórios, definida por  $N(\mu, \sigma^2)$ , onde  $\mu = 0$  e  $\sigma^2 = 1$ . Esses valores são definidos por serem variáveis aleatórias de distribuição normal e identicamente distribuídas [QIN94].

Para a modelagem de  $n$  quadros gerados na simulação, a equação (5.15) calcula a combinação entre  $d$  e  $k$ , onde  $k = 1, 2, \dots, n$ . O resultado do cálculo dessa combinação define a aproximação da autocorrelação teórica em função do parâmetro Hurst e também em relação aos  $n$  valores calculados anteriormente.

$$X_n = \sum_{k=1}^n \binom{d}{k} \cdot X_{n-k} + N(0,1) \quad n \geq 0 \quad (5.15)$$

Pela equação (5.13) apresentada anteriormente, obtém-se o valor de  $d$ . A função de autocorrelação teórica utilizada para comparação com o método proposto (5.15) é a equação (5.16).

$$r(k) = \frac{d(1+d) \cdots (k-1+d)}{(1-d)(2-d) \cdots (k-d)} \quad (k = 1, 2, \dots) \quad (5.16)$$

As aproximações de autocorrelação (5.16), (5.17) e (5.18) foram propostas em [GAR94], [BOR97] e [QIN94] respectivamente. Essas aproximações possuem a mesma característica da curva de autocorrelação teórica (5.12), a qual gera um gráfico com o traçado hiperbólico decrescente.

As curvas teóricas de autocorrelação de ambas as equações (5.16) e (5.17) são praticamente idênticas. Isso indica o alto grau de similaridade entre as equações teóricas de aproximação da autocorrelação.

$$r(k) = H(2H - 1) |k|^{2H-2} \quad k \rightarrow \infty \quad (5.17)$$

A equação (5.18) é uma particularidade da equação (5.16) enquanto  $k \rightarrow \infty$  [QIN94].

$$r(1) = \frac{d}{(1-d)} \quad r(k) \sim \frac{(-d)!}{(d-1)!} k^{2d-1} \quad k \rightarrow \infty \quad (5.18)$$

Inicialmente o cálculo da função (5.18) comprometeria o processamento dessa equação pelas características de recursividade para o cálculo fatorial. No entanto, ao considerar a equação (5.19), substitui-se o cálculo fatorial pela função *gamma* [QIN94].

$$r! = \Gamma(r+1) \quad (5.19)$$

A substituição do cálculo fatorial pela função *gamma* (5.20), proposta em [QIN94], simplifica o cálculo da autocorrelação e elimina necessidade de processamento recursivo, exclusivamente para o cálculo fatorial.

$$\Gamma(r) = \int t^{r-1} e^{-t} dt \quad (5.20)$$

A Figura 5.3 abaixo, apresenta a complexidade do algoritmo definida como  $o(n^2)$  para gerar  $n$  valores. O gráfico apresenta o número de intervalos a processar em função do tempo de processamento em segundos.

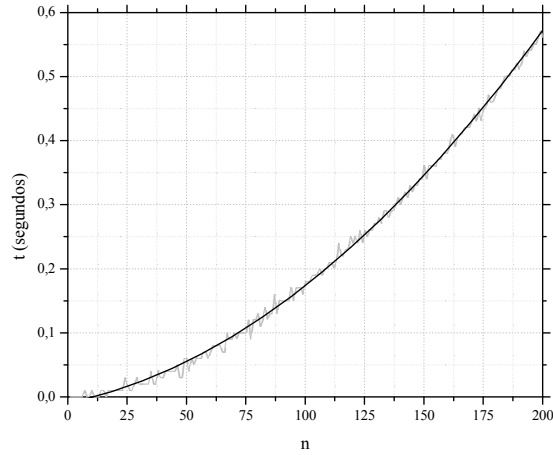


Figura 5.3: Complexidade do algoritmo F-ARIMA – Adaptação Combinatória.

### 5.2.3. F-ARIMA – Adaptação Hosking

Nessa seção será apresentado o algoritmo F-ARIMA proposto em [QIN94] para simulação com adaptação Hosking. Em [QIN94] esse método é descrito como um processo estacionário de distribuição normal.

O valor inicial de  $X_0$  é escolhido pela função de distribuição normal  $N(0, v_0)$ , da mesma forma como na proposta da Seção 5.2.1. A geração dos quadros é resultado da iteração de  $k=1,2,\dots,n$ , onde para cada iteração de  $k$ , são calculados os coeficientes parciais de correlação  $\phi_{kj}$  para  $j=1,2,\dots,k$ :

$$\phi_{kk} = \frac{d}{(k-d)} \quad (5.21)$$

$$\phi_{kj} = \phi_{k-1,j} - \phi_{kk}\phi_{k-1,k-j} \quad j=1,2,\dots,k-1 \quad (5.22)$$

$$m_k = \sum_{j=1}^k \phi_{kj} X_{k-j} \quad (5.23)$$

$$v_k = (1 - \phi_{kk}^2) v_{k-1} \quad (5.24)$$

$$X_k = N(m_k, v_k) \quad (5.25)$$

O valor de  $d$  é obtido pela equação (5.13), apresentada anteriormente. A função de autocorrelação teórica utilizada para comparação com o método proposto é a equação (5.16). Ambas as discussões para obtenção de  $d$  e a escolha da função teórica da autocorrelação são exatamente as mesmas características apresentadas para o modelo F-ARIMA direto.

A Figura 5.4, a seguir, apresenta a complexidade do algoritmo definida como  $o(n^2)$  para gerar  $n$  valores. O gráfico apresenta o número de intervalos a processar em função do tempo de processamento em segundos.

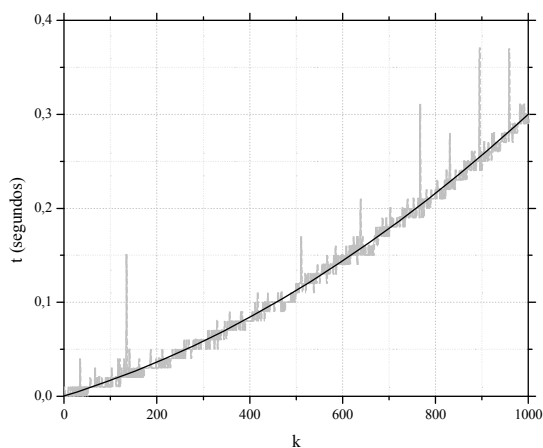


Figura 5.4: Complexidade do algoritmo F-ARIMA – Adaptação Hosking.

A implementação Hosking apresentou menor custo computacional para geração do tráfego de vídeo sintetizado. No entanto, o desempenho do algoritmo está condicionado à disponibilidade de memória para realização dos cálculos, devido a dependência dos valores para os quadros anteriores.

### 5.2.4. F-ARIMA – Diferença Fracional

Nessa seção será apresentado o algoritmo F-ARIMA proposto em [CRU98] para simulação com filtro de aproximação por diferença fracional. Ao invés de tratar cada elemento separado do algoritmo ARIMA, extrai-se o componente integrativo do processo aleatório [CRU98].

Sendo a série temporal  $X(k)$  é um processo aleatório de tamanho  $n$ , com valores aleatórios de distribuição normal, na qual os valores são gerados a partir da iteração de  $k = 1, 2, \dots, n$ , como apresentado na equação (5.26).

$$X(k) = N(0,1) \quad (5.26)$$

O processo aleatório resultante, gerado a partir da proposta em [CRU98], é resultado do filtro FIR (do inglês *Finite Impulse Response*) de aproximação por diferença fracional  $H(k)$ , apresentado na equação (5.27), sobre o processo aleatório  $X(k)$ .

$$H(k) = (1 - k^{-1})^d \quad (5.27)$$

O modelo ARIMA utiliza o processo de filtragem para aproximar os dados, a partir do filtro da aproximação  $H(k)$ , sobre o processo aleatório  $X(k)$ , gerando um processo aleatório auto-similar normalizado.

A função de autocorrelação teórica utilizada para comparação com o método proposto é a equação (5.28), como proposta em [CRU98]. O valor  $\beta$  é obtido pela equação (2.57), e o valor de  $d$  é obtido pela equação (5.13), como apresentado anteriormente.

$$r(k) = \frac{(k+1)^{2-\beta} - 2 \cdot k^{2-\beta} + (k-1)^{2-\beta}}{2} \quad (5.28)$$

A Figura 5.5, a seguir, apresenta a complexidade do algoritmo definida como  $o(n)$  para gerar  $n$  valores para o processo aleatório. O gráfico apresenta o número de intervalos a processar em função do tempo de processamento em segundos.

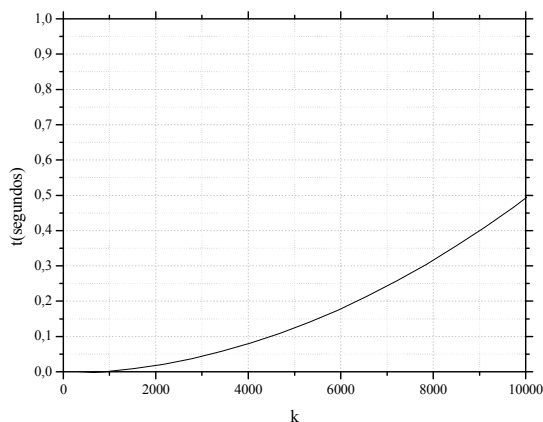


Figura 5.5: Complexidade do algoritmo F-ARIMA – Diferença Fracional.

A implementação F-ARIMA por diferença fracional apresentou menor custo computacional para geração do tráfego de vídeo sintetizado, pelo fato do algoritmo não requerer tratar os valores gerados nos quadros anteriores, reduzindo então o consumo de memória.

### 5.3. Agrupamento de Quadros

O agrupamento de quadros para o vídeo hierárquico deve seguir um conjunto de regras e considerações que serão tratadas nessa seção. O padrão da estrutura de um GOP considerado nas simulações é um padrão apresentado em vários modelos de tráfego de vídeo. No entanto alguns vídeos podem ser codificados, onde se pode definir um padrão de estrutura do GOP diferente do padrão de 12 quadros por GOP. Essa característica requer que o modelo de simulação seja ajustável a qualquer estrutura de GOP.

Considera-se que podem existir três tipos de quadros definidos no padrão MPEG contidos em um GOP, como apresentado na Seção 3.3. Os quadros do tipo I, P e B são os quadros que definem um GOP. A estrutura hierárquica considera os quadros do tipo B, como

quadros que armazenam as camadas de melhorias da imagem [FIT02]. Os quadros do tipo I e P são considerados como quadros da camada básica da estrutura MPEG.

O agrupamento GOP convencional considera uma estrutura típica de doze quadros em um GOP. Essa estrutura é definida pela seqüência IBBPBBPBBPBB, onde há um quadro do tipo I, para três quadros do tipo P e oito quadros do tipo B.

A primeira condição de verificação considera que sempre haverá ao menos um quadro do tipo I, sempre com referência ao próximo GOP, que define a estrutura de quadros mais básica de um GOP com tamanho unitário. Essa estrutura não considera o modelo hierárquico nem a estrutura base de quadros do tipo P.

Ao se considerar em um GOP o quadro I e somente os quadros do tipo B, os quais definem a camada de melhoria com referência aos quadros anteriores e posteriores, deve haver sempre ao menos dois quadros, devido à relação de interdependência entre os quadros do tipo I. Isso deve ocorrer mesmo que não exista nenhum quadro do tipo P. Constata-se então que o segundo agrupamento de um GOP possível é um agrupamento IBB, definido por três quadros.

Quando são considerados novos quadros do tipo P na estrutura, então se devem inserir dois novos quadros do tipo B para que a relação de interdependência entre os quadros seja garantida. O terceiro agrupamento de um GOP possível é um agrupamento IBBPBB, definido por seis quadros. Em conformidade com o padrão da estrutura MPEG, cada quadro do tipo P inserido na estrutura do GOP, deve existir dois novos quadros do tipo B.

A Tabela 5.1 apresenta a relação de números de quadros por tipo necessários para gerar na simulação, em função do tamanho de GOP definido por  $T_{GOP}$ .

Tabela 5.1: Relação de quadros para o modelo hierárquico.

$T_{GOP}$	$N_I$	$N_P$	$N_B$
1	1	0	0
3	1	0	2
6	1	1	4
9	1	2	6
12	1	3	8
15	1	4	12
18	1	5	14



Naturalmente a estimativa do número de quadros do tipo I necessários para se simular é o número de GOPs gerados na simulação, definido por  $N_{GOP}$ . Essa é uma consequência devido à característica dos GOPs possuírem somente um quadro do tipo I.

No entanto para os quadros do tipo P e B, o tamanho do GOP influencia no número de quadros necessários para a geração de dados. A equação (5.29) define a fórmula para estimativa de geração de número de quadros do tipo P. Para a estimativa do número de quadros  $N_p$  devem-se considerar somente valores no conjunto dos números inteiros e positivos, incluindo o zero.

$$N_p = \left( \frac{T_{GOP}}{3} \right) - 1 \quad N_p = \{0, 1, 2, 3, \dots\} \quad (5.29)$$

A equação (5.30) define a fórmula para estimativa de geração de número de quadros do tipo B. Para estimativa do número de quadros  $N_B$  devem-se considerar somente os valores presentes no conjunto dos números pares e positivos, incluindo o zero.

$$N_B = \left( \frac{T_{GOP}}{3} \right) \cdot 2 \quad N_B = \{0, 2, 4, 6, \dots\} \quad (5.30)$$

A soma do número de GOPs com a soma das equações (5.29) e (5.30), define o número de quadros a gerar no modelo de simulação. O resultado dessa soma servirá como referência para a distribuição dos quadros de acordo com a estrutura do GOP.

Para cada tipo de quadro, uma nova lista de quadros deverá ser gerada, com as suas próprias propriedades estatísticas. O tamanho dessas listas é definido de acordo com as equações apresentadas acima, e quanto ao número de GOPs para a lista de quadros do tipo I. O tamanho da lista da simulação hierárquica terá o tamanho  $N$  definido na equação (5.31).

$$N = N_{GOP} + N_p + N_B \quad (5.31)$$

Dados esses requisitos, define-se um contador  $n$  para controlar o tamanho total da lista de agrupamento de todos os quadros, definido por  $N$  na equação (5.31). Da mesma forma

define-se um contador  $g$  para controlar o número de quadros agrupados em um GOP, delimitados por  $T_{GOP}$ , como apresentado na Tabela 5.1.

Para cada iteração de  $n$  até o número total de quadros simulados  $N$ , deve-se verificar o contador de número de quadros processados por GOP, definido por  $g$ . Dessa forma se identifica quando o número de quadros processados  $n$  atingiu a um número total de quadros  $T_{GOP}$ . Nessa situação, o contador deve-se atribuir o valor 0 para o contador  $g$  com o intuito de indicar o início de um novo ciclo de agrupamento GOP.

Para cada tipo de quadro, existe um contador para controlar o número de quadros agrupados. Dessa forma se define  $i$  como contador de quadros do tipo I,  $p$  como contador de quadros do tipo P e  $b$  como contador de quadros do tipo B. Quando qualquer um desses contadores for superior ao tamanho da lista de simulação correspondente ao tipo de quadro, então não se deve mais agrupar quadros. Essa condição define o ponto de parada de agrupamento de quadros.

Quando o contador  $g$  referenciar o primeiro quadro, ou seja, quando  $g = 1$ , então se deve considerar a simulação gerada para os quadros do tipo I. Para se identificar os quadros do tipo P, então se deve considerar a equação lógica (5.32).

$$n = (p \cdot 3) + (i \cdot 3) + 4 \quad (5.32)$$

Caso essa condição lógica seja satisfeita, então o quadro é do tipo P e deve-se incrementar o contador  $p$ . Como serão considerados os 3 tipos de quadros, então caso nenhuma das duas condições não sejam satisfeitas, deve-se considerar como agrupamento de quadros do tipo B.

Essa proposta possibilita o agrupamento hierárquico de GOP para qualquer tamanho múltiplo de 3 ou igual a 1. Simulações de características especiais quanto à estrutura GOP podem se beneficiar dessa proposta, cujas características são de baixa complexidade e de baixo custo computacional.

## 5.4. Conclusões

A aproximação da curva teórica (5.1) definida para processos auto-similares com a curva de autocorrelação obtida a partir de um vídeo real, caracteriza um modelo auto-similar. As características definidas para processos auto-similares, apresentadas na Seção 2.7, devem ser consideradas no modelo de simulação.

Os algoritmos de modelagem e simulação para processos auto-similares geram valores aleatórios, a partir da aproximação teórica (5.1) da função de autocorrelação. Nos algoritmos F-ARIMA propostos são utilizados os parâmetros determinados pelo coeficiente de Hurst, o respectivo  $\beta$  (2.57) e  $d$  (5.13), assim como os valores da média (2.1) e da variância (2.3) para obtenção dos valores de autocorrelação (2.32).

Uma característica das propostas dos algoritmos é a complexidade estar definida para todas como  $o(n^2)$ , com exceção da proposta por diferença fracional [CRU98], onde a complexidade é  $o(n)$ . Essa característica é essencialmente crítica quando se trata de processos auto-similares possuírem grande armazenamento de memória e de longa duração. O modelo deve prever uma maior geração de quadros para se aproximar a curva de autocorrelação com a curva teórica definida para o modelo. Essa constatação indica que a complexidade é uma característica importante para a construção do modelo, a partir das propostas de codificação do algoritmo F-ARIMA. A Tabela 5.2 apresenta as características dos algoritmos apresentados de acordo com as características de tempo de processamento, memória, precisão dos dados gerados e em função da complexidade do algoritmo.

Tabela 5.2: Características das complexidades dos algoritmos.

<b>Método</b>	<b>Processamento</b>	<b>Memória</b>	<b>Precisão</b>	<b>Complexidade</b>
<b>Recursivo</b>	***	***	**	***
<b>Combinatória</b>	***	*	*	**
<b>Hosking</b>	*	***	***	**
<b>Fracional</b>	*	*	***	*

A proposta de agrupamento dos tipos de quadros define a estratégia para reconstrução do tráfego de vídeo hierárquico, realizada independentemente do tipo de quadro. A proposta de agrupamento independe do tamanho definido para o GOP, uma vez que esse tamanho seja múltiplo de 3 ou unitário. No entanto, para agrupamento de tamanho unitário,

não é considerado geração de modelagem hierárquica pelo fato de não se definir camadas de melhoria a ser aplicada na decodificação da imagem.

## Capítulo 6

### Simulações de Tráfego de Vídeo Hierárquico

Os conceitos estatísticos apresentados na Seção 4.2 definem as informações requeridas para se avaliar o comportamento de um tráfego de vídeo. A partir das considerações estatísticas de auto-similaridade apresentadas na Seção 4.3, alguns modelos foram apresentados na Seção 5.2. Esses modelos são baseados nas características de auto-similaridade dos processos aleatórios, conforme apresentado na Seção 2.7.

Esses modelos dependem essencialmente das características estatísticas e dos coeficientes de similaridade. As características atribuídas ao coeficiente de Hurst determinam a curva da autocorrelação e, conseqüentemente, as propriedades de auto-similaridade entre as comparações da autocorrelação de tráfego real e simulada. Esses valores são fundamentais para se calibrar a curva de autocorrelação em relação às características dos tipos de quadros e de resolução de vídeo que se deseja simular.

Nesse capítulo serão apresentadas as avaliações dos algoritmos de simulação F-ARIMA e os resultados das simulações realizadas considerando os modelos de geração apresentados no capítulo anterior. Os processos aleatórios de tráfego de vídeo real, onde serão calculados os valores da autocorrelação, devem ser comparados com o resultado da autocorrelação obtida na simulação. Serão apresentadas também as estatísticas dos vídeos simulados, assim como foi realizada a análise para um arquivo real.

## 6.1. Avaliação dos Algoritmos de Simulação

Com a finalidade de validar os algoritmos de modelagem e simulação apresentados na Seção 5.2, nessa seção serão apresentadas as comparações das curvas de autocorrelação (2.32) geradas a partir do resultado das simulações, em relação à aproximação da curva de autocorrelação teórica definida para o modelo. A implementação do algoritmo F-ARIMA, a utilizar nesse experimento, dependerá dos resultados obtidos nessa avaliação.

As diferenças entre as curvas da autocorrelação das simulações em relação às curvas de autocorrelação teóricas estão restritas à complexidade do algoritmo, ao número de quadros a gerar e ao número de repetições da simulação. Devido à característica de dependência de longa duração, será necessário gerar um volume mais expressivo de quadros na simulação. O volume necessário dependerá da aproximação da curva da autocorrelação do processo aleatório, em comparação à curva de autocorrelação teórica. Quando as curvas estiverem próximas, então não será necessário gerar mais dados para o modelo.

Nas simulações apresentadas a seguir, definiu-se como constante o intervalo de geração de 1000 iterações, definido por  $k$ , sendo que o algoritmo de simulação foi repetido em 10 ocorrências. Como apresentado em [QIN94], a curva da autocorrelação da simulação se aproxima da curva de autocorrelação teórica, enquanto se considera o valor da média da autocorrelação, resultante das repetições das amostras geradas na simulação. No entanto, deve-se considerar que as repetições devem ser utilizadas somente para avaliação da curva de autocorrelação. O valor da variância é alterado durante as repetições, quando a variância é diferente de 1. Essa situação influencia no cálculo dos valores aleatórios, utilizado no algoritmo F-ARIMA.

O coeficiente de Hurst utilizado nas simulações é constante e definido em 0,7000. O respectivo valor de  $\beta$  (2.57), obtido para o cálculo da autocorrelação da função teórica auto-similar (5.1), foi de 0,6000 e o valor de  $D$  (5.13) igual a 0,2000, utilizado no cálculo na função teórica de autocorrelação (5.16).

A Figura 6.1 apresenta o comparativo da autocorrelação teórica (5.1) de um processo aleatório auto-similar em comparação à autocorrelação da simulação de um processo aleatório (2.32) gerada pelo algoritmo F-ARIMA apresentado na Seção 5.2.1.

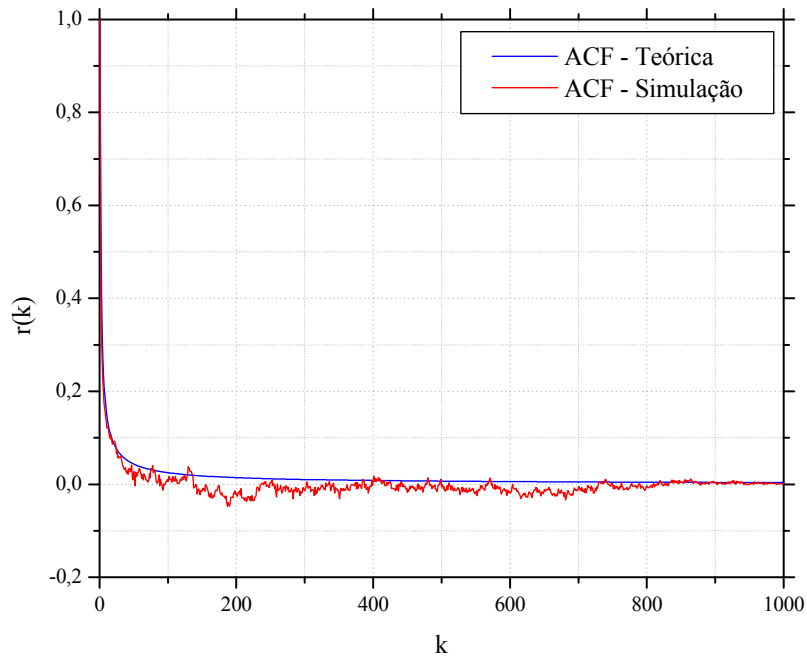


Figura 6.1: Autocorrelação de Simulação em comparação à autocorrelação teórica.

A função de aproximação da autocorrelação utilizada na equação (5.4) foi substituída por (5.1), a fim de evitar o uso recursivo de aproximação da autocorrelação em função de Hurst.

A complexidade desta proposta é exponencial. Essa característica sucede ao fato de utilizar valores recursivos em (5.7). Deve-se considerar também a substituição da função de aproximação da autocorrelação recursiva pela teórica na equação (5.4), com o objetivo de reduzir o tempo de processamento na geração do modelo e manter a aproximação da autocorrelação.

No entanto, em função da característica de dependência de longa duração e da complexidade do algoritmo F-ARIMA devido à recursividade, um grande esforço computacional ainda é necessário para utilização do modelo de simulação com o algoritmo apresentado na Seção 5.2.1.

De forma a avaliar outros modelos de desenvolvimento, outras duas propostas do algoritmo ARIMA foram apresentadas na Seção 5.2.2 e 5.2.3 [QIN94]. Essas duas novas

propostas são adaptações do algoritmo F-ARIMA, as quais consideram respectivamente a análise combinatória e o cálculo parcial de correlação (Hosking).

Como no algoritmo da Seção 5.2.1, estas propostas também procuram manter as aproximações de autocorrelação no modelo de simulação, com o intuito de preservar as características de processos auto-similares. Basicamente estas novas propostas simplificam os cálculos de aproximação da autocorrelação em substituição dos cálculos recursivos.

A Figura 6.2 apresenta a comparação do cálculo de autocorrelação das simulações de acordo com a adaptação de Hosking, a adaptação combinatória e a autocorrelação da equação teórica (5.16).

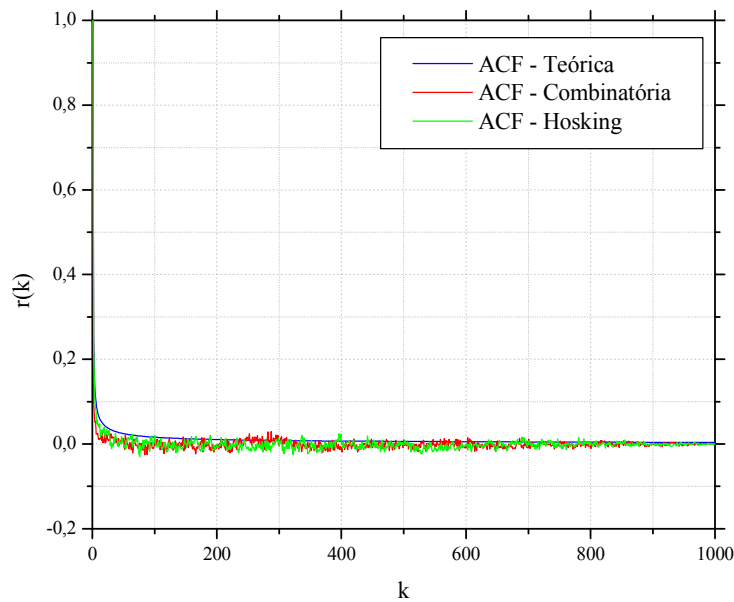


Figura 6.2: Autocorrelação de Simulação Hosking e Combinatória em comparação à autocorrelação teórica.

As curvas de autocorrelação na Figura 6.2 são praticamente as mesmas quando comparadas às propostas de desenvolvimento F-ARIMA com adaptação combinatória e de Hosking. A Figura 6.3 apresenta a comparação do cálculo de autocorrelação da simulação com a adaptação de Hosking em comparação com a autocorrelação teórica (5.16).



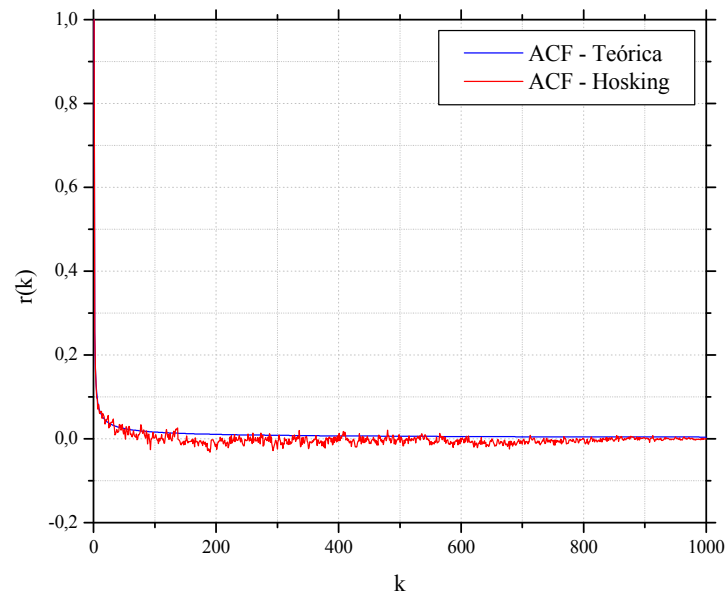


Figura 6.3: Autocorrelação de Simulação de Hosking em comparação com a teórica.

Em ambas as propostas se utilizou o mesmo valor de Hurst para a geração da simulação e da curva teórica. Outros valores de Hurst foram utilizados para validar o algoritmo de simulação. No entanto, constata-se que quanto maior o valor de Hurst, maior o número de quadros necessários para o modelo gerar. Essa característica representa a dependência de longa duração, definido na Seção 2.7, nas propriedades dos processos auto-similares [QIN94].

De forma a gerar valores com menor custo computacional, e com maior precisão, em relação às informações estatísticas coletadas, a implementação proposta do algoritmo F-ARIMA apresentada na Seção 5.2.4 por [CRU98], viabiliza um modelo de tráfego de vídeo de baixa complexidade. A Figura 6.4 apresenta a comparação do cálculo de autocorrelação da simulação com a adaptação por diferença fracional, em comparação com a autocorrelação teórica (5.28).

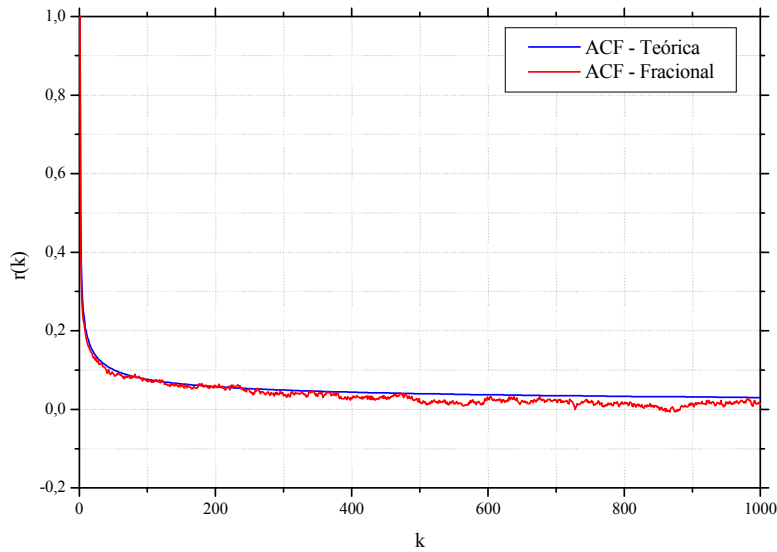


Figura 6.4: Comparação da Autocorrelação Teórica e de Simulação por diferença fracional.

Para os parâmetros utilizados para essa simulação, assumiu-se o coeficiente de Hurst em  $0,8000$ . O respectivo valor de  $\beta$  (2.57), obtido para o cálculo da autocorrelação da função teórica auto-similar (5.1), foi de  $0,4000$  e o valor de  $D$  (5.13) igual a  $0,3000$ , utilizado no cálculo na função teórica de autocorrelação (5.28).

Uma vez considerada a equivalência das curvas de autocorrelação geradas a partir dos modelos de simulação, então se considera a proposta de menor complexidade para utilização no modelo. O modelo selecionado é a adaptação F-ARIMA com diferença fracional, proposta em [CRU98] e detalhada na Seção 5.2.4.

## 6.2. Geração de Simulação

As simulações de vídeo foram realizadas considerando a proposta do método de estimativa do coeficiente de auto-similaridade, ou coeficiente Hurst, para um vídeo real na Seção 4.3. No entanto, o coeficiente de Hurst não deve depender somente da análise de um vídeo real. Na simulação de tráfego de vídeo nem sempre existirá um vídeo real para se estimar o coeficiente de Hurst para então gerar a simulação.

Para isso devem ser considerados valores aproximados para o coeficiente de Hurst, considerando o tipo de quadro, a resolução do vídeo e a complexidade da imagem. Como observado nos experimentos realizados (Tabela 4.8), cada tipo de vídeo possui um valor diferente de aproximação do coeficiente de Hurst. Isso é refletido também nos tipos de quadros da imagem.

Através da aproximação do cálculo da autocorrelação, os valores podem ser comparados com os gráficos artificiais de aproximação e com os valores do que seria um fluxo de um vídeo real.

Os valores estimados de Hurst utilizados nas simulações foram obtidos por análise dos próprios arquivos de vídeo real, como apresentado na Seção 4.1. Dessa maneira, calcula-se o valor da autocorrelação teórica a partir da obtenção do valor  $\beta$ , resultado da equação (2.57).

Como os valores da autocorrelação tendem ao infinito, no experimento considerou-se o comportamento de um modelo de simulação em um intervalo definido. Esse intervalo foi utilizado para o cálculo da estimativa de Hurst. Consideraram-se então para as simulações 2000 quadros do tipo I, 6000 quadros do tipo P e 16000 quadros do tipo B. O procedimento de cálculo desses números é apresentado na Seção 5.3.

A Figura 6.5 apresenta a autocorrelação da simulação de quadros do tipo I, em comparação com o vídeo *Star Wars IV* em alta resolução [FIT01] e com a autocorrelação teórica. O coeficiente de Hurst geral obtido e utilizado nesse experimento foi de  $0,7459$ .

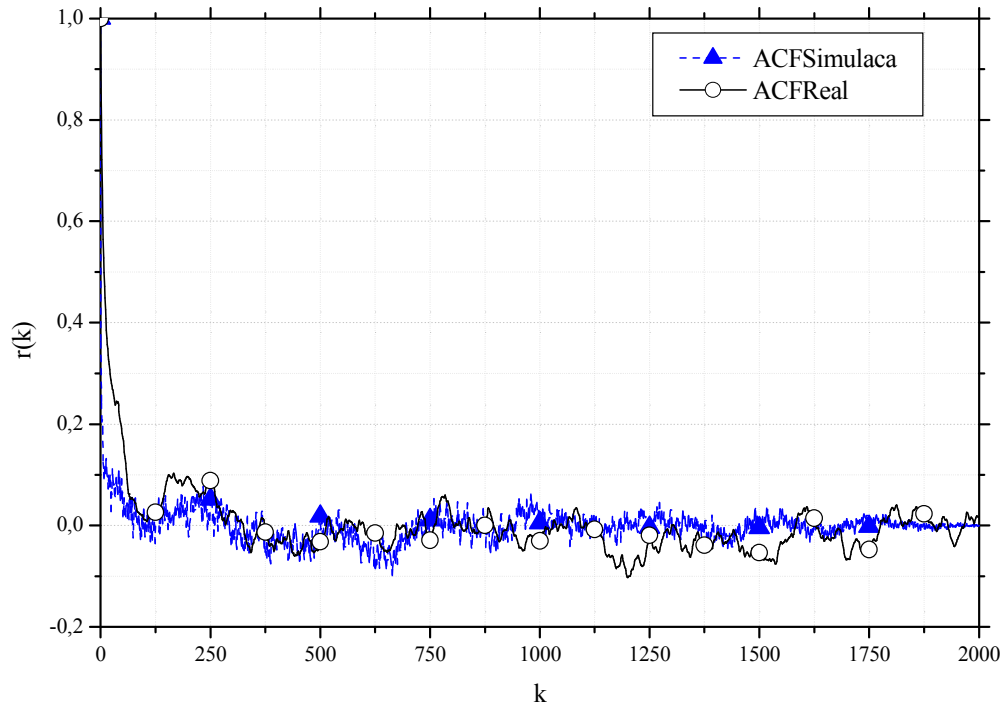


Figura 6.5: Autocorrelação da simulação de quadros I em alta resolução.

A Figura 6.6 apresenta a autocorrelação da simulação de quadros do tipo P, em comparação com a autocorrelação do vídeo *Star Wars IV* em alta resolução [FIT01] e a autocorrelação teórica. O coeficiente de Hurst obtido nesse experimento foi de  $0,8038$ .

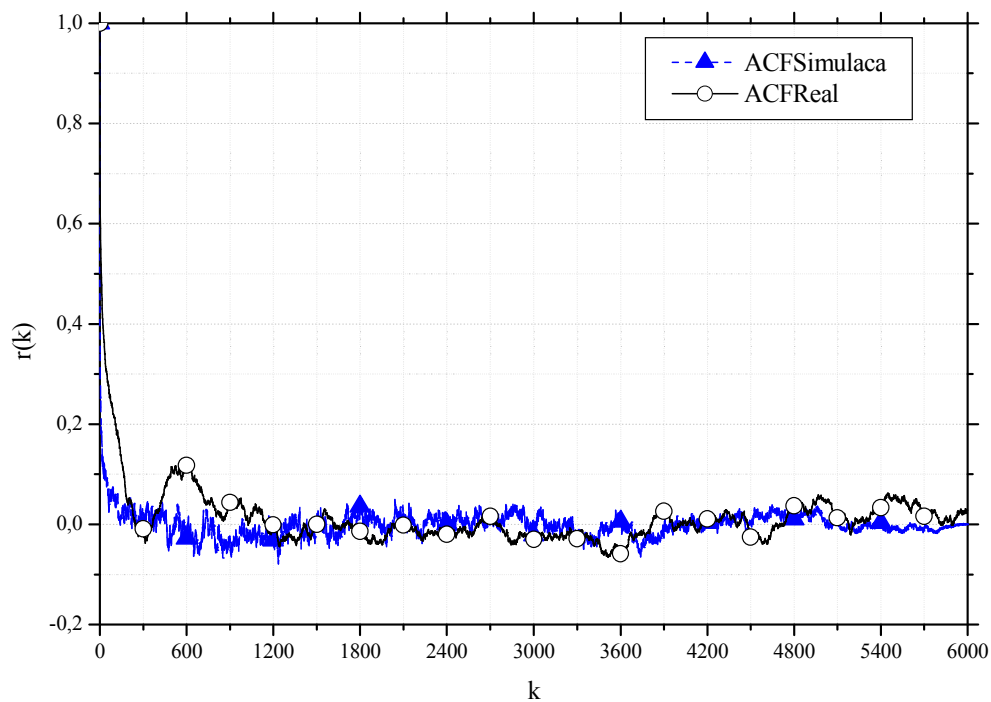


Figura 6.6: Autocorrelação da simulação de quadros P em alta resolução.

A Figura 6.7 apresenta a autocorrelação da simulação de quadros do tipo B, em comparação com a autocorrelação do vídeo *Star Wars IV* em alta resolução [FIT01] e a autocorrelação teórica. O coeficiente de Hurst obtido nesse experimento foi de  $0,8294$ .

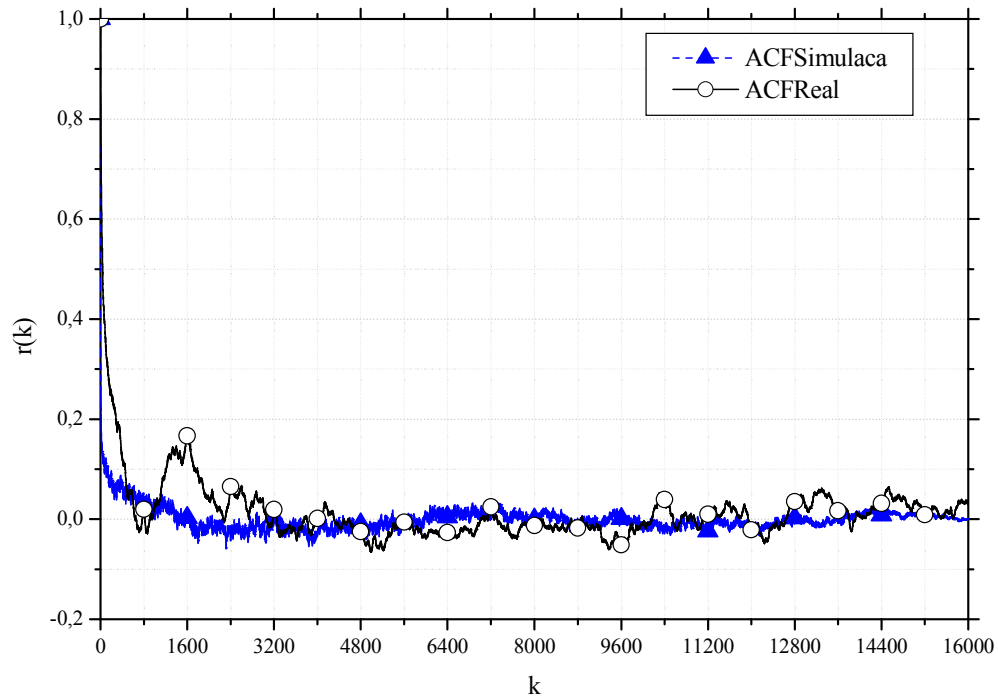


Figura 6.7: Autocorrelação da simulação de quadros B em alta resolução.

A Figura 6.8 apresenta a autocorrelação da simulação de quadros do tipo I, em comparação com a autocorrelação do vídeo *Alladin* em média resolução [FIT01] e com a autocorrelação teórica. O coeficiente de Hurst obtido nesse experimento foi de  $0,8111$ .

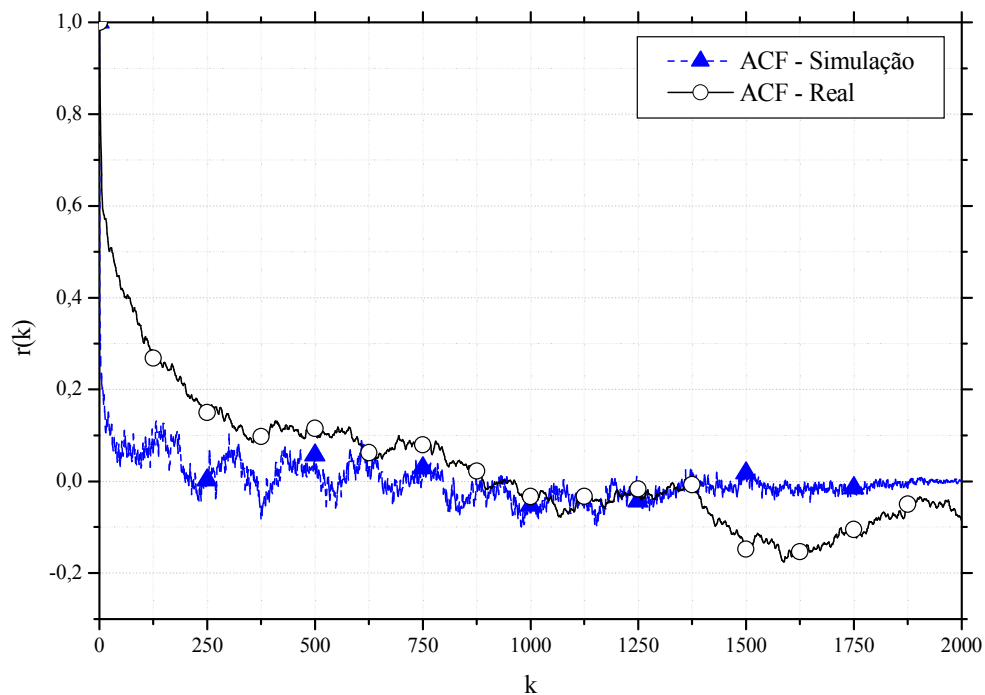


Figura 6.8: Autocorrelação da simulação de quadros I em média resolução.

A Figura 6.9 apresenta a autocorrelação da simulação de quadros do tipo P, em comparação com o vídeo *Aladdin* em média resolução [FIT01], em comparação com a autocorrelação teórica e real do vídeo. O coeficiente de Hurst obtido nesse experimento foi de 0,7721.

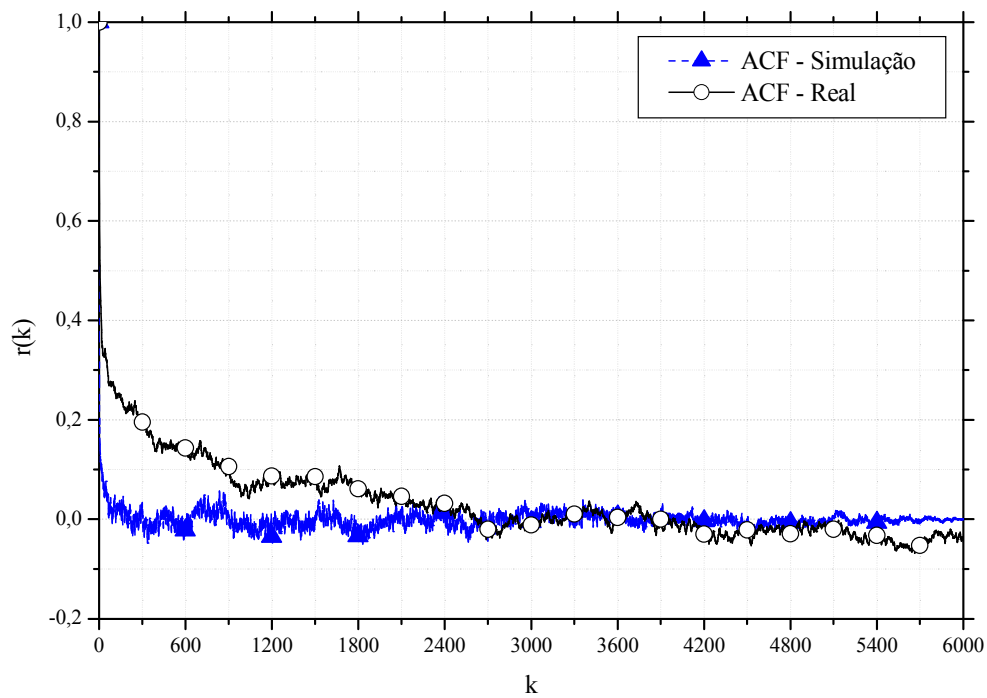


Figura 6.9: Autocorrelação da simulação de quadros P em média resolução.



A Figura 6.10 apresenta a autocorrelação da simulação de quadros do tipo B, em comparação com o vídeo *Aladdin* em média resolução [FIT01], em comparação com a autocorrelação teórica e real do vídeo. O coeficiente de Hurst obtido nesse experimento foi de 0,7373.

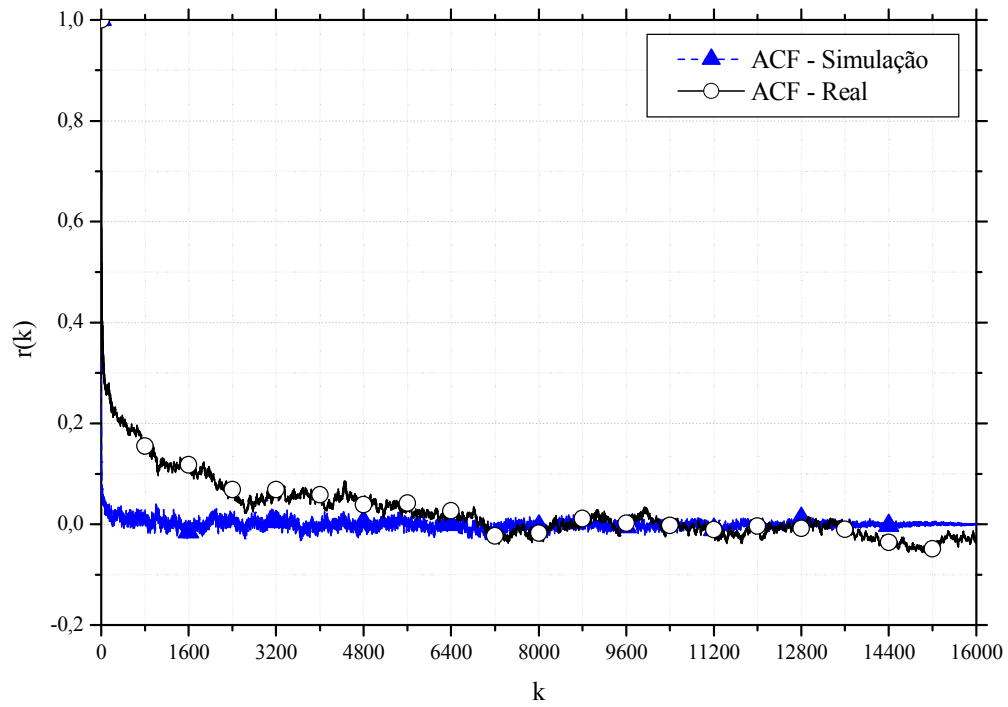


Figura 6.10: Autocorrelação da simulação de quadros B em média resolução.

A Figura 6.11 apresenta a autocorrelação da simulação de quadros do tipo I, em comparação com a autocorrelação do vídeo *Jurassic Park* em baixa resolução [FIT01] e com a autocorrelação teórica. O coeficiente de Hurst obtido nesse experimento foi de  $0,7782$ .

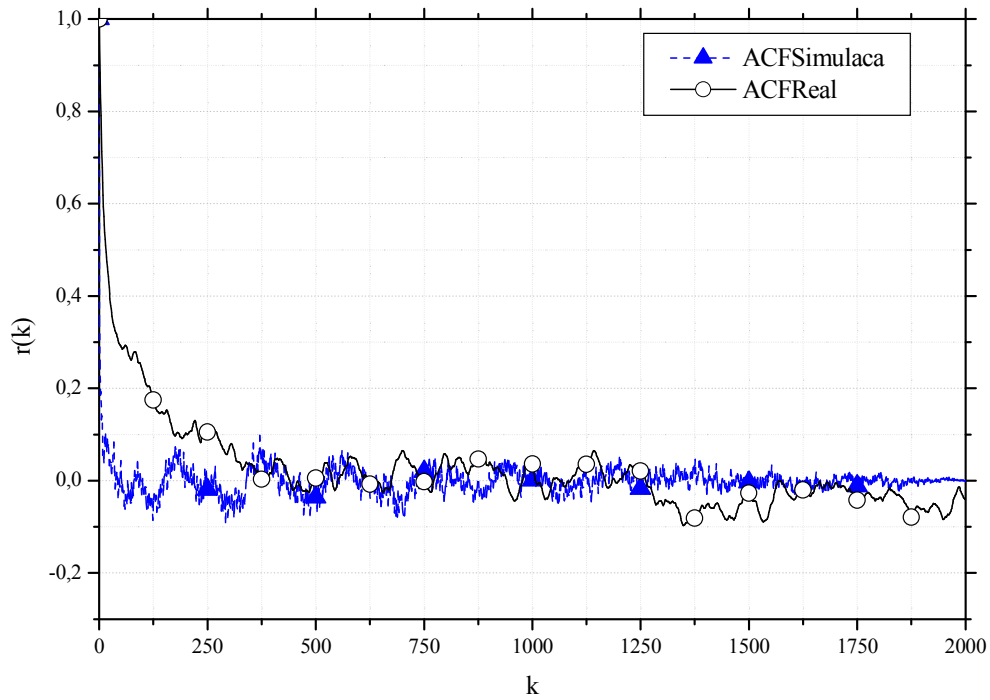


Figura 6.11: Autocorrelação da simulação de quadros I em baixa resolução.

A Figura 6.12 apresenta a autocorrelação da simulação de quadros do tipo P, em comparação com a autocorrelação do vídeo *Jurassic Park* em baixa resolução [FIT01] e com a autocorrelação teórica. O coeficiente de Hurst obtido nesse experimento foi de  $0,7700$ .

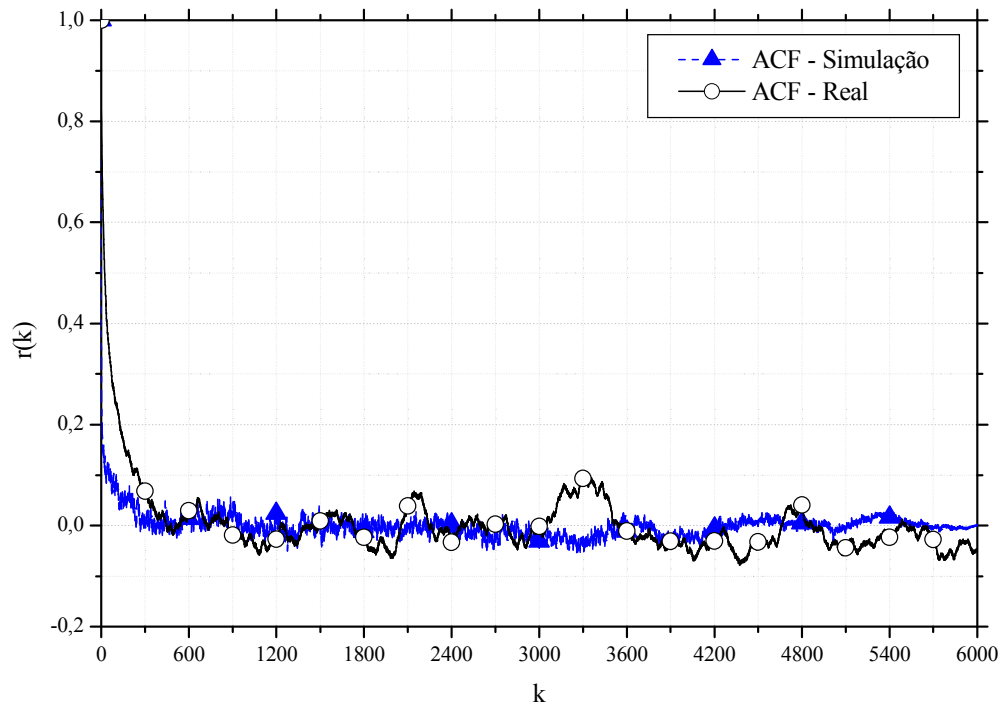


Figura 6.12: Autocorrelação da simulação de quadros P em baixa resolução.

A Figura 6.13 apresenta a autocorrelação da simulação de quadros do tipo B, em comparação com a autocorrelação do vídeo *Jurassic Park* em baixa resolução [FIT01] e com a autocorrelação teórica. O coeficiente de Hurst obtido nesse experimento foi de 0,7799.

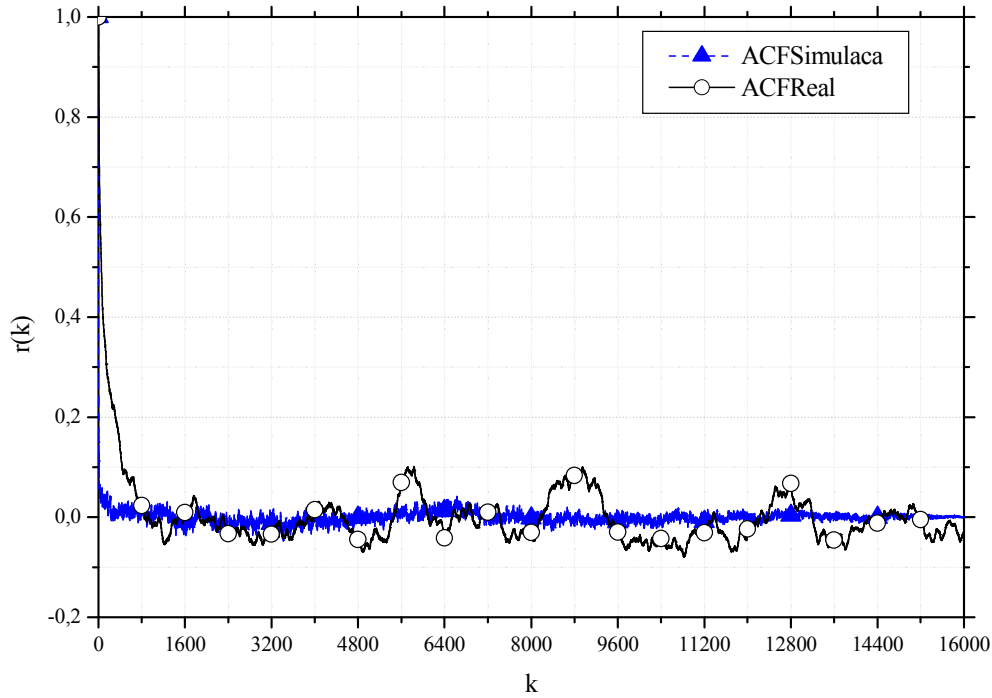


Figura 6.13: Autocorrelação da simulação de quadros B em baixa resolução.

As curvas de autocorrelação da simulação apresentam resultados próximos em relação às curvas de autocorrelação dos arquivos de vídeo real. Pequenas variações no coeficiente de auto-similaridade, não influenciam nos dados gerados pelo modelo.

Os coeficientes estimados de Hurst, os quais foram utilizados nas simulações, foram obtidos através da aproximação da média dos coeficientes de Hurst. Para a geração dessa simulação foram utilizados somente os parâmetros de tamanho dos quadros, resolução do vídeo e coeficiente de auto-similaridade.

### 6.3. Modelagem Hierárquica

A modelagem hierárquica tem como objetivo propor um método de geração de tráfego de vídeo em camadas, baseado nas simulações realizadas isoladamente para cada tipo de resolução de vídeo e tipo de quadro.

Na Seção 6.2, as simulações foram avaliadas como processos aleatórios auto-similares para cada resolução de vídeo, onde foi considerado somente o tamanho da simulação e o coeficiente de auto-similaridade, estimado para cada tipo de quadro. Foram efetuadas as comparações da curva de autocorrelação entre o resultado da simulação e a autocorrelação do arquivo compactado de vídeo.

O resultado da simulação da Seção 6.2, realizado isoladamente por tipo de resolução de vídeo e tipo de quadro é insumo para o modelo proposto de simulação hierárquica, onde se considera também os valores estatísticos da média e da variância por resolução de vídeo e tipo de quadro.

Outra característica para o modelo é a definição do tamanho da estrutura hierárquica para agrupamento dos quadros. Para as simulações realizadas nesse trabalho o agrupamento de quadros definido para as simulações foi de 12 quadros por GOP.

No modelo proposto, a primeira definição é a resolução do vídeo a ser gerado. Como conseguinte devem ser obtidos os resultados e avaliar as análises estatísticas para o tipo de vídeo selecionado para simulação. Com esses valores calculados é possível estimar o coeficiente de auto-similaridade por tipo de quadro.

A definição da estrutura do GOP deve considerar o número de quadros a gerar na simulação, uma vez que devem existir quadros suficientes nos intervalos da simulação por tipo de quadro para o agrupamento do vídeo hierárquico. A Tabela 6.1 apresenta os parâmetros de simulação utilizados para a simulação hierárquica.

Tabela 6.1: Parâmetros definidos para o modelo de simulação de tráfego de vídeo.

Resolução	Quadro	Média ( $\mu \times 10^3$ ) bytes	Variância ( $\sigma^2 \times 10^6$ ) bytes	Hurst	Número de Quadros	GOP
Alta	I	3,42473	1,01868	0,74585	1750	12
Alta	P	1,55134	0,62416	0,80376	5250	12
Alta	B	1,05488	0,22377	0,82944	14000	12
Média	I	2,21342	0,89688	0,81110	1750	12
Média	P	0,91768	0,47912	0,77207	5250	12
Média	B	0,54143	0,25263	0,73731	14000	12
Baixa	I	3,64596	1,72063	0,77819	1750	12
Baixa	P	0,96686	0,44366	0,77002	5250	12
Baixa	B	0,33466	0,08824	0,77988	14000	12

A Figura 6.14 apresenta o processo de geração de tráfego de vídeo hierárquico, proposto pelo modelo. A definição dos valores da Tabela 6.1 é determinante para o modelo, pois em todo o processo são considerados esses valores para geração do tráfego de vídeo.

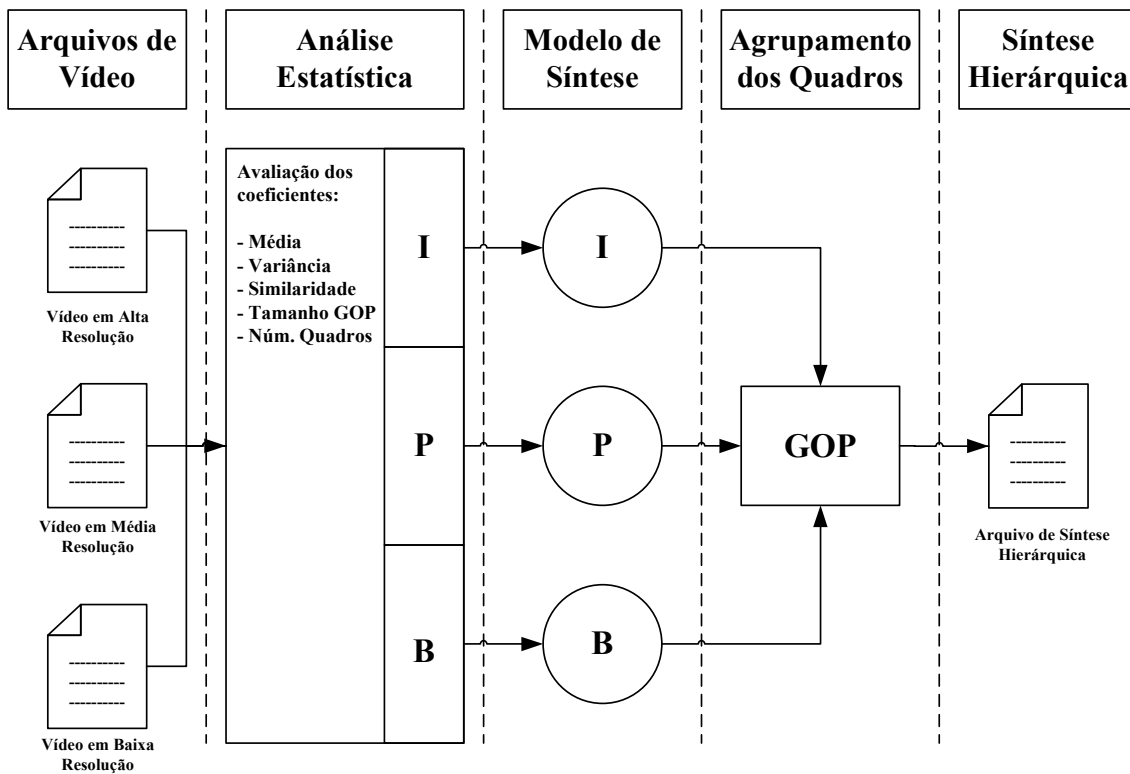


Figura 6.14: Modelo para geração de tráfego de vídeo hierárquico.

Os arquivos de vídeo real devem ser avaliados de acordo com a qualidade da resolução dos vídeos selecionados para simulação. A avaliação desses vídeos produz os coeficientes de simulação que serão utilizados pelo modelo. Os coeficientes de simulação envolvem os valores da média e da variância, a estimativa do coeficiente de Hurst, o número de quadros e o tamanho do GOP.

A geração dos quadros sintetizados deve ocorrer independentemente para cada tipo de quadro, de acordo com os coeficientes de simulação obtidos na análise do vídeo selecionado. Esses resultados são agrupados de acordo com a estrutura do GOP definida pelo número de quadros por GOP.

As variâncias são utilizadas para multiplicar e as médias para somar com os valores obtidos na simulação. Esse procedimento é requerido para se estabelecer a aproximação das informações da simulação com os valores de tendência central e de dispersão. Depois de agrupado, esse resultado é arredondado para valores inteiros, para representar a medida dos tamanhos dos quadros em *bytes*.

A Figura 6.15, a Figura 6.16 e a Figura 6.17 apresentam a autocorrelação geral para alta, média e baixa resolução respectivamente.

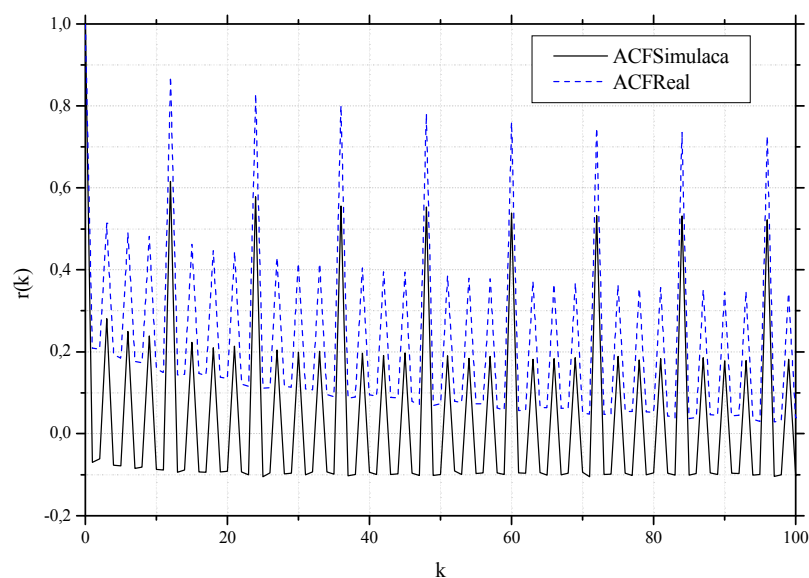


Figura 6.15: Autocorrelação geral da simulação para alta resolução.

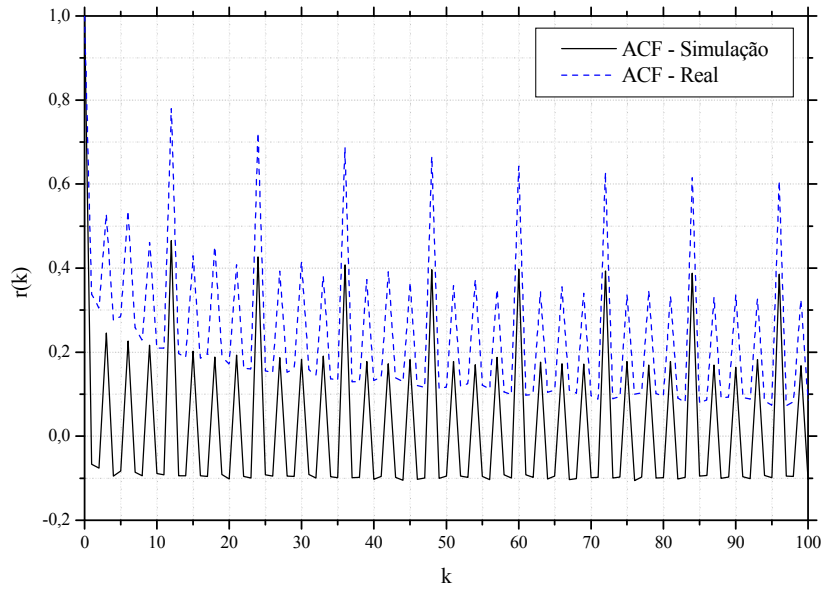


Figura 6.16: Autocorrelação geral da simulação para média resolução.

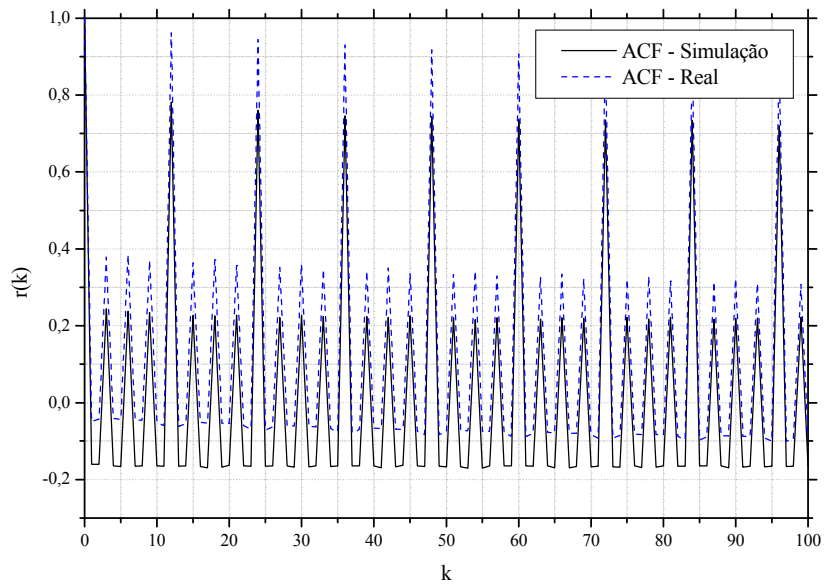


Figura 6.17: Autocorrelação geral da simulação para baixa resolução.



As estatísticas das simulações, como a função de densidade cumulativa e a função de densidade de probabilidade, possuem os valores relativamente correlatos às estatísticas geradas de arquivos reais. As simulações tendem a manter os valores das estatísticas dos arquivos reais e a distribuição dos quadros da mesma forma.

Os valores da simulação são multiplicados pelo desvio padrão e somados à média dos arquivos reais avaliados. Essa operação aproxima os valores estatísticos no processo aleatório de simulação, em relação às informações estatísticas encontradas no processo aleatório real.

#### **6.4. Conclusões**

Independentemente da resolução do vídeo, as simulações seguem o padrão da curva de autocorrelação avaliada a partir de um vídeo real. As simulações consideram os parâmetros definidos para o modelo, tais como o tamanho de GOP, os cálculos estatísticos da média e da variância para cálculo da autocorrelação e as estimativas do coeficiente de Hurst a partir da análise de auto-similaridade.

O modelo hierárquico foi obtido a partir do agrupamento de simulações isoladas para cada tipo de quadro e para cada tipo de resolução. Outra informação necessária para o agrupamento é o número de quadros a gerar na simulação e também o tamanho de cada GOP, a definir a estrutura de interdependência entre os quadros do tipo I, P e B.

A comparação com os resultados obtidos na simulação em relação ao vídeo real ocorre na comparação das curvas de autocorrelação. A curva de autocorrelação depende das definições dos parâmetros utilizados no modelo e também das características inerentes aos processos auto-similares. Quanto maior o coeficiente de Hurst, maior o número de quadros necessário para gerar durante a modelagem e simulação, com a finalidade de aproximar a curva de autocorrelação sintetizada em relação à curva de autocorrelação teórica.



## Capítulo 7

### Conclusão

A metodologia para a proposta do modelo de simulação é formada a partir do levantamento de informações estatísticas para processos aleatórios e processos auto-similares, estudo das características do método de compressão MPEG para vídeo hierárquico, avaliação de métodos de simulação F-ARIMA, proposta de estruturação de vídeo hierárquico e análise dos resultados obtidos das simulações.

As análises estatísticas realizadas, a partir de arquivos compactados de vídeo real, apresentaram as informações necessárias para a geração do modelo de simulação. Esses resultados foram obtidos por análise estatística em função de processos aleatórios e de características auto-similares. Também foram utilizados esses valores para validação das características de similaridade em comparação aos resultados gerados pela simulação.

A análise de similaridade também considerou um método para estimativa do coeficiente de auto-similaridade a partir dos valores estatísticos calculados. Os valores estimados de Hurst possuem valores distintos para diferentes intervalos definidos para essa análise. A partir de um intervalo definido, existe um valor que praticamente estabiliza o valor estimado do coeficiente de auto-similaridade. Os valores estáveis para os intervalos de estimativa foram considerados para as simulações. Esses valores variam em função do tipo de quadro e da resolução de vídeo apresentado.

Propostas de construção do algoritmo F-ARIMA foram avaliadas com o objetivo de utilizar o método mais eficiente à proposta de modelagem de simulações. Os algoritmos apresentaram resultados em função das características de similaridade com grande precisão, como observado nas curvas de autocorrelação. A precisão das implementações F-ARIMA foi avaliada ao comparar a curva de autocorrelação teórica, com a autocorrelação do resultado da simulação.

As características de complexidade desses algoritmos e também a relação de espaço e tempo são elementos críticos, quando se trata da característica de similaridade de longa duração. Foi observado que, quanto maior o grau de similaridade, maior o número de quadros requeridos na simulação. Essa característica é determinada pelo valor do coeficiente de auto-similaridade, com o objetivo de aproximar a curva de autocorrelação da curva teórica. Quanto maior o número de quadros gerados na simulação, mais próximas são as curvas de autocorrelação calculadas a partir dos dados da simulação, em comparação com as curvas de autocorrelação teórica.

Independentemente do tipo de quadro e da resolução de vídeo, as simulações caracterizam um processo aleatório auto-similar. No entanto, o modelo de tráfego de vídeo hierárquico requer a avaliação dessas simulações agrupadas. O modelo de simulação hierárquica propõe o agrupamento das simulações parciais por tipo de quadro e por resolução de vídeo. Cada simulação parcial é gerada a partir de um conjunto de informações obtidas pelas análises estatísticas do vídeo real.

Após o procedimento de agrupamento das simulações parciais, a avaliação da simulação completa caracteriza o modelo de tráfego de vídeo hierárquico. O resultado desse agrupamento gera uma nova análise de similaridade, onde se identificam as características da estrutura do GOP.

As aproximações da curva de autocorrelação calculada a partir da simulação hierárquica, em comparação com a curva de autocorrelação calculada para os arquivos compactados de vídeo real, identificam as características dos processos auto-similares. As características de similaridade e estatística mantêm-se também para os processos simulados.

O resultado gerado pelo modelo de simulação pode ser utilizado em trabalhos futuros relacionados à avaliação de redes de computadores e de telecomunicações, qualidade de serviço e demais características de rede. Para essa avaliação podem ser utilizadas ferramentas de simulação de tráfego em rede ou mesmo em ferramentas de avaliação de redes já implementadas.

## Referências Bibliográficas

- [ANS02] ANSARI, Nirwan, LIU, Hai, SHI, Yun Q., ZHAO, Hong; *On Modeling MPEG Video Traffics*. IEEE Transaction and Broadcasting. Department of Electrical and Computer Engineering - University of New Jersey, USA, 2002.
- [BOL96] BOLOT, Jean-Chrysostome, TURLETTI, Thierry; *Adaptive Error Control for Packet Video in the Internet*. IEEE. INRIA France / MIT LCS USA, 1996.
- [CRU98] CRUZ, Luis J., PALLARES, Esteve, ALINS, Juan J., MATA, Jorge; *Self-Similar Traffic Generation Using a Fractional ARIMA Model. Application to the VBR MPEG Video Traffic*. Department of Applied Mathematics and Telematics at Polytechnic University of Catalonia, Spain, 1998, p.102-107.
- [COM00] COMER, Douglas E.; *Internetworking with TCP/IP volume 1, principles, protocols, and architectures*. 2a. Edição. Editora: Prentice-Hall, 2000.
- [DEV01] DEVADHAR, Siddhartha, KRUMBEIN, Cederic, LIU, Kim Man; *MPEG Background*. <http://bmrc.berkeley.edu/frame/research/mpeg>, Berkeley Research Centre at Berkeley University, USA, 2001.
- [FIT01] FITZEK, F. H., REISSLEIN, M.; *MPEG-4 and H.263 Video Traces for Network Performance Evaluation*. <http://www-tkn.ee.tu-berlin.de/research/trace/trace.html>, Department of Electrical Engineering at the Technical University of Berlin and Arizona State University, Germany, 2001, p.40-54.
- [FIT02] FITZEK, Frank H., REISSLEIN, Martin, LASSETTER, Jeremy, LOTFALLAH, Osama, PANCHANATHAN, Sethuraman; *Traffic and Quality Characterization of Scalable Encoded Video: A Large-Scale Trace-Based Study*. Department of Electrical Engineering at Arizona State University, USA, 2002, p.1-26.

- [FIT03] FITZEK, Frank H., REISSLEIN, Martin, SEELING, Patrick, ZORZI, Michele; *Video and Audio Trace Files of Pre-encoded Video Content for Network Performance Measurements*. Department of Electrical Engineering at Arizona State University, USA, and DipInge Università di Ferrara, Italy, 2003, p.11.
- [GAR94] GARRETT, Mark W., WILLINGER, Walter; *Analysis, Modeling and Generation of Self-Similar VBR Video Traffic*. ACM SigComm, London, 1994.
- [JEO98] JEONG, Hae-Duck J., MCNICKLE, Don, PAWLIKOWSKI, Krzysztof; *A Comparative Study of Three Self-Similar Tele-Traffic Generator*. Department of Computer Science and Management - University of Canterbury, New Zealand, 1998.
- [LEE99] LEE, Y. B.; *UVoD: A Unified Architecture for Video-on-Demand Services*. IEEE Communications Letters. Department of Information Engineering, University of Hong Kong, China, 1999.
- [MAR03] MARTINS, Ricardo F., LEITE, Carlos A., FARINES, Jean M.; *Towards Quality Evaluation and Improvement of a MPEG Video Stream*. The 3rd IEEE Latin American Network Operations and Management Symposium. Iguassu Falls, Brazil, 2003.
- [PRA96] PRASAD, Anand R., STAVROV, Borut, SCHOUTE C., Frits; *Generation and Testing of Self-Similar Traffic in ATM Networks*. IEEE Telecommunications and Traffic Control Systems Group. Delft University of Technology, Netherland, 1996, p.200-205.
- [QIN94] YONG-QING, Lu, PETR, David W., FROST, Victor; *Characterization and Modeling of Long-Range Dependent Telecommunication Traffic*. Telecommunications and Information Sciences Laboratory, Department of Eletrical Engineering and Computer Science, USA. University of Kansas, USA, 1994.
- [ROD96] RODEN, Tino V.; *H.261 and MPEG1 - a comparison*. Praktische Informatik IV, D-68131 Mannheim, Germany. University of Mannheim, Germany, 1996.

- [ROS97] ROSS, Sheldon M.; *Simulation*. 2nd Edition. Hartcourt and Academic Press, USA, 1997.
- [ROS00] ROSS, Sheldon M.; *Introduction to Probability Models*. 7th Edition. Hartcourt and Academic Press, 2000.
- [SCH01] SCHREMMER, Claudia, KUHMÜNCH, Christoph, EFFELSBERG, Wolfgang; *Layered Wavelet Coding for Video*. 11th International Packet Video Workshop (PV 2001), pp. 42 ff., Kyongju, Korea, April 2001.
- [SEE04] SEELING Patrick, REISSLEIN, Martin, KULAPALA Beshan; *Network Performance Evaluation Using Frame Size and Quality Traces of Single-Layer and Two-Layer Video: a Tutorial*. IEEE Communications Surveys and Tutorials. Arizona State University, USA, 2004, p.58-78.
- [SHA74] SHAMBLIN James E, STEVENS Jr., G. T.; *Operations Research*. 1st Edition. McGraw-Hill Inc., USA, 1974, p.18-28, 37-38, 67-73.
- [WAG72] WAGNER Harvey M.; *Principals of Operations Research*. 1st Edition. Prentice-Hall International Inc., UK, 1972.