



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

# INTERPOLAÇÃO DE IMAGENS ATRAVÉS DE EXTRAPOLAÇÃO DOS COEFICIENTES DA DSTR

Dissertação submetida à Faculdade de Engenharia Elétrica e de Computação da Universidade Estadual de Campinas, Departamento de Telecomunicações como parte dos requisitos exigidos para a obtenção o título de Mestre em Engenharia Elétrica.

Autor

Patrick Biesdorf

Bacharel em Ciência da Computação pela UNIOESTE em 2000

Orientador

Prof. Dr. Yuzo Iano

Doutor em Engenharia Elétrica pela UNICAMP em 1986

Banca Examinadora

Prof. Dr. Yuzo Iano - FEEC/UNICAMP - Presidente

Prof. Dr. Messias Meneguette Júnior - FCT/UNESP-PP

Prof. Dr. Dalton Soares Arantes - FEEC/UNICAMP

Prof. Dr. Edson Moschim - FEEC/UNICAMP

Campinas, 2003

*Ofereço este trabalho a Deus, a minha família e especialmente a Paula Brito, pela sua paciência e amor.*

# Agradecimentos

À Deus acima de tudo.

Ao meu pai pelo apoio em todos os sentidos. A minha mãe pelas orações e incentivo.

A minha companheira Paula Brito.

Ao Prof. Yuzo Iano, que me deu a oportunidade de ingressar nesta jornada e me guiou pelos caminhos incertos.

Aos colegas de aula e companheiros de moradia, por serem amigos, confiantes e auxiliares.

À UNICAMP - Universidade Estadual de Campinas, à UNIOESTE - Universidade Estadual do Oeste do Paraná e seus professores, por fornecerem os recursos que me permitiram realizar este trabalho.

Ao CNPq por fornecer as condições financeiras tão escassas em nosso país.

# Conteúdo

<b>Agradecimentos</b>	<b>iii</b>
<b>Conteúdo</b>	<b>iv</b>
<b>Lista de Tabelas</b>	<b>vi</b>
<b>Lista de Figuras</b>	<b>vii</b>
<b>Siglas</b>	<b>viii</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Problema . . . . .	5
1.2 Objetivo . . . . .	6
1.3 Organização dos Capítulos . . . . .	6
<b>2 Interpolação</b>	<b>9</b>
2.1 Teoria de Amostragem . . . . .	9
2.1.1 Amostragem Periódica . . . . .	10
2.1.2 Representação da Amostragem no Domínio da Frequência . . . . .	11
2.1.3 Teorema de Amostragem de Nyquist . . . . .	11
2.1.4 Reconstrução de uma Função Limitada em Banda a partir de suas Amostras . . . . .	11
2.1.5 Aumento da Taxa de Amostragem por um Fator Inteiro (Processamento Discreto) . . . . .	12
2.1.6 Considerações Gerais Sobre Imagens Digitais . . . . .	14
2.2 Interpolação no Domínio Espacial . . . . .	15
2.2.1 Método Clássico . . . . .	15
2.2.2 Método Generalizado . . . . .	17
2.2.3 Teoria de Aproximação . . . . .	20
2.2.4 Casos Particulares . . . . .	27
2.3 Interpolação no Domínio da Transformada . . . . .	37

2.3.1	DFT . . . . .	38
2.3.2	DCT . . . . .	39
2.3.3	DSTr . . . . .	43
<b>3</b>	<b>Extrapolação</b>	<b>51</b>
3.1	Projeções Ortogonais . . . . .	53
3.2	Energia Mínima . . . . .	59
3.2.1	Redução da Energia do Erro . . . . .	59
3.2.2	Norma Mínima e Mínimos Quadrados . . . . .	60
3.2.3	Mínimos Quadrados Ponderados . . . . .	63
3.2.4	Modificação Iterativa da Norma Mínima Ponderada . . . . .	64
3.3	Dimensão Reduzida . . . . .	66
3.3.1	Estimação Limitada em Banda e Concentrada em Tempo . . . . .	67
3.4	Concentração de Energia . . . . .	69
3.5	Dimensão Essencial . . . . .	73
<b>4</b>	<b>Aplicação da Extrapolação nos Coeficientes da DSTr</b>	<b>77</b>
4.1	Considerações Sobre o Algoritmo de Extrapolação . . . . .	77
4.2	Descrição do Método Proposto . . . . .	81
<b>5</b>	<b>Resultados Obtidos</b>	<b>87</b>
5.1	Comparação . . . . .	87
<b>6</b>	<b>Conclusão</b>	<b>99</b>
6.1	Considerações e Conclusão . . . . .	99
6.2	Trabalhos Futuros . . . . .	101
6.3	Publicações . . . . .	103
	<b>Referências Bibliográficas</b>	<b>105</b>
<b>A</b>	<b>Seqüências Esferoidais Estendidas</b>	<b>109</b>
A.1	Funções Esferoidais Estendidas . . . . .	109
A.2	Seqüências Esferoidais Estendidas Discretas . . . . .	110
A.3	Seq. Esferoidais Estendidas Discretas Periódicas . . . . .	112
A.4	Expansão de Karhunen Loéve . . . . .	114

# Lista de Tabelas

5.1	SNR e PSNR para DSTr e DSTrEx. Imagem 'Lena' . . . . .	89
5.2	SNR e PSNR para DSTr e DSTrEx. Imagem 'Peppers' . . . . .	89
5.3	SNR e PSNR para DSTr e DSTrEx. Imagem 'Baboo' . . . . .	89
5.4	SNR e PSNR para DSTr e DSTrEx. Imagem 'Rabbit' . . . . .	92
5.5	SNR e PSNR para DSTr e DSTrEx. Imagem 'Figuras Geométricas' . . . . .	94
5.6	SNR e PSNR para DSTr e DSTrEx coeficientes originais. Imagem 'Lena' . . . . .	94
5.7	SNR e PSNR para DSTr e DSTrEx coeficientes originais. Imagem 'Peppers' . . . . .	94
5.8	SNR e PSNR para DSTr e DSTrEx coeficientes originais. Imagem 'Baboo' . . . . .	94
5.9	SNR e PSNR para DSTr e DSTrEx coeficientes originais. Imagem 'Rabbit' . . . . .	95
5.10	SNR e PSNR para DSTr e DSTrEx coeficientes originais. Imagem 'Figuras Geométricas' . . . . .	96

# Lista de Figuras

2.1	Interpolação da função $x$ a partir de suas amostras. . . . .	13
2.2	Núcleos de aproximação em Fourier. Sinc X Vizinho mais próximo . . . . .	27
2.3	B-spline cúbica. . . . .	31
2.4	Núcleo de aproximação para B-splines de vários graus. . . . .	32
2.5	Construção de uma seqüência de simetria par. . . . .	41
2.6	Extensão periódica de um bloco. . . . .	42
2.7	Correlação Residual versus Coeficiente de Correlação para $N = 16$ . . . . .	44
2.8	Seqüência de simetria ímpar. . . . .	45
2.9	Rotação de eixos de um bloco. . . . .	47
3.1	Redução da Energia do Erro - Esquema de blocos . . . . .	60
3.2	Primeira iteração do algoritmo. . . . .	61
4.1	Ilustração da seqüência de interpolação de um bloco. . . . .	82
4.2	Diagrama de blocos representando o método proposto. . . . .	83
5.1	Imagem original "Lena" $512 \times 512$ . . . . .	88
5.2	Imagem original "Peppers". . . . .	90
5.3	Imagem original "Baboo". . . . .	91
5.4	Imagem original "Rabbit". . . . .	92
5.5	Imagem original "Figuras Geométricas". . . . .	93
5.6	Gráfico SNR $\times$ quantidade de coeficientes. Imagem "Lena". . . . .	95
5.7	Gráfico SNR $\times$ quantidade de coeficientes. Imagem "Peppers". . . . .	96
5.8	Gráfico SNR $\times$ quantidade de coeficientes. Imagem "Baboo". . . . .	97

# Siglas

**BLTC** - *Band-Limited Time Concentrated*

**DCT** - *Discrete Cosine Transform*

**DFT** - *Discrete Fourier Transform*

**DPSS** - *Discrete Prolate Spheroidal Sequences*

**DST** - *Discrete Sine Transform*

**DSTr** - *Discrete Sine Transform with Axis Rotation*

**EC** - *Energy Concentration*

**ER** - *Error Reduction*

**JPEG** - *Joint Picture Experts Group*

**MNLS** - *Minimum Norm Least Square*

**MPEG** - *Motion Picture Experts Group*

**P-DPSS** - *Periodic Discrete Prolate Spheroidal Sequences*

**PSF** - *Prolate Spheroidal Functions*

**PSNR** - *Peak Signal to Noise Ratio*

**SNR** - *Signal to Noise Ratio*

**WLS** - *Weighted Least Squares*

# Capítulo 1

## Introdução

O termo processamento digital de imagens se refere ao tratamento de imagens por circuitos digitais, incluindo computadores, processadores de sinais digitais (*Digital Signal Processors* - DSPs), ou outro dispositivo específico. Para que uma imagem seja tratada por um desses dispositivos, ela precisa estar na forma digital. Uma imagem digital é considerada um sinal bidimensional discretizado tanto em amplitude quanto em espaço. Nos dispositivos digitais elas são geralmente representadas como matrizes cujos valores podem ser inteiros, reais ou complexos, porém com precisão finita, limitada à quantidade de bits utilizados para se representar esses números. Cada ponto da imagem é chamado de pixel (*picture element*) e seu valor pode ter diversos significados, de acordo com o padrão em que a imagem é armazenada, como por exemplo, uma cor em uma imagem com mapa de cores, um nível de intensidade de branco em uma imagem em escala de cinza, a intensidade da componente (vermelho, verde ou azul) em uma imagem RGB (*Red-Green-Blue*) ou até mesmo um coeficiente complexo de uma transformada em uma imagem codificada ou algum outro tipo de significado em uma dada aplicação.

As imagens digitais podem ser naturais ou sintéticas. As imagens naturais geralmente são provenientes de amostragem de imagens contínuas usando-se algum dispositivo de captura como *scanner*, placas capturadoras de vídeo, máquinas filmadoras ou fotográficas

digitais, entre outros. As imagens sintéticas são imagens geradas por dispositivos digitais e programas que geram os pixels, independentes de uma imagem contínua, como por exemplo as imagens geradas por editores de gráficos bidimensionais e tridimensionais e programas de computação gráfica, entre outros.

Por ser um sinal bidimensional, imagens digitais são grandes consumidoras de recursos como armazenamento, processamento ou transmissão, tornando seu processamento muito custoso e muitas vezes impraticável. Felizmente, nas últimas décadas vem-se observando um ciclo de desenvolvimento da tecnologia de circuitos digitais que tem possibilitado a queda nos custos de fabricação de dispositivos eletrônicos, como por exemplo os computadores pessoais e seus periféricos. A queda dos custos, e em consequência dos preços desses dispositivos, possibilitou que consumidores os adquiram em maior quantidade ou invistam nas suas melhorias (*upgrade*) tornando-os cada vez mais populares e permitindo que sejam utilizados em um número crescente de aplicações. Ao mesmo tempo, com a demanda maior desses dispositivos, as empresas passam a investir mais no desenvolvimento da tecnologia, diminuindo ainda mais os custos.

Dentro desse ciclo, observa-se a participação do processamento digital de imagens, podendo ser cada vez mais utilizado, visto que os recursos necessários para a sua aplicação se tornam cada vez mais acessíveis. Por outro lado, se os recursos são mais abundantes, novas aplicações em processamento de imagens podem ser criadas. Quando essas novas aplicações são consideradas necessárias, mais recursos computacionais são adquiridos para que elas possam ser executadas com melhor desempenho.

Dessa forma, cada vez mais o processamento de imagens tem sido necessário, para uma série de aplicações, tais como:

- Sensoriamento Remoto - satélites que transmitem imagens com uma série de informações, como temperatura, umidade, tipo de vegetação e outras, que são uti-

lizadas na meteorologia (previsão do tempo) e monitoramento do meio ambiente.

- Radar e sonar - Utilizados em aplicações militares, aeroespaciais, navegação e até mesmo pesca (sonar).
- Geo-processamento: geração por computador de mapas com diferentes níveis de detalhamento. Também utilizado no Sistema de Posicionamento Global (*Global Positioning System* - GPS).
- Imagens médicas - Nesta área há uma variedade de imagens como raio-X, tomografia e ultra-som que auxiliam o médico no diagnóstico.
- Controle industrial automatizado (visão robótica) - controles automáticos baseados em informações extraídas de imagens na linha de produção de indústrias.
- Monitoramento de segurança - Cameras de vigilância que são acionadas ou acionam algum outro dispositivo a partir de características pré-definidas do vídeo, como movimento e mudanças de temperatura (imagens infravermelho).
- Reconhecimento automático de características pessoais - identificação de características pessoais únicas (digital, face, retina) a partir da comparação com imagens armazenadas em um banco de dados.
- Entretenimento - Nesse campo pode-se citar jogos de computadores que utilizam recursos extremos de computação gráfica, arquivos de fotografias e vídeos e processadores de imagens.

O Processamento Digital de Imagens divide-se em diversas áreas com diferentes características e funções. Entre elas pode-se citar:

- Codificação e Compressão - como as imagens digitais utilizam um conjunto grande de dados, faz-se necessário que sua informação seja comprimida para diminuir a necessidade de recursos para a sua transmissão ou armazenamento.
- Restauração - imagens distorcidas por algum problema na captura ou transmissão podem ser restauradas se o motivo da distorção for conhecido. Por exemplo, é possível restaurar digitalmente imagens fora de foco ou borradas por deslocamento.
- Enriquecimento - também é possível destacar informações aparentemente escondidas em imagens utilizando-se enriquecimento, como por exemplo, aumentando-se o contraste.
- Segmentação e Classificação - essa área estuda a prática de extração e identificação automática de informação de imagens.
- Filtragem - utilizada para extrair ou realçar informações da imagem no domínio da frequência. Também utilizada para se reduzir o ruído contido na imagem.
- Interpolação e Decimação - em alguns casos é necessário visualizar a imagem em uma escala diferente. Geralmente a interpolação é utilizada para que detalhes, de uma determinada região da imagem analisada, possam ser visualizados.

A interpolação apresenta uma série de aplicações, que vai desde o entretenimento, onde um usuário amplia uma imagem para preencher a tela de seu computador, até o uso médico onde o profissional precisa visualizar os detalhes de um raio-X, por exemplo, para poder determinar o diagnóstico. Essas aplicações tem demandado cada vez mais interpolações de boa qualidade. Porém, a perda de informações quando a imagem é digitalizada ou decimada, impõe um limite ao processo de interpolação que necessita ser contornado.

Como foi dito, imagens digitais podem ser provenientes de uma infinidade de origens. Apesar de serem geralmente naturais (digitalizadas a partir de imagens contínuas) ou sintéticas (produzidas por computador), dentro desses conjuntos, elas podem ser divididas em subconjuntos com características em comum. Por exemplo, dentro do conjunto de imagens naturais, existem as imagens médicas, as de satélites e as de fotografias. As fotografias podem ser de paisagens, de pessoas ou somente de faces.

Para limitar o escopo do problema, será utilizado neste trabalho, somente imagens naturais de fotografias monocromáticas digitalizadas com 256 níveis de cinza, com tamanhos variados medidos em pixels. Entretanto, a aplicação do método proposto em outro tipos de imagens não apresenta nenhuma limitação a princípio.

## 1.1 Problema

A interpolação de sinais discretos pode ser vista como um aumento na taxa de amostragem do sinal contínuo. Portanto, uma forma de se conseguir, do ponto de vista teórico, um sinal interpolado, seria recuperar o sinal contínuo e amostrá-lo novamente a uma nova taxa [1]. Porém, sabe-se, a partir da teoria de amostragem, que para que isso seja possível, o sinal discreto deve possuir amostras suficientes de acordo com sua frequência máxima. No entanto, esse sinal, que provavelmente não possui banda limitada (frequência máxima desconhecida), pode ter sido filtrado antes do processo de aquisição ou decimação, para evitar sobreposição espectral (*aliasing*), no momento em que foi amostrado ou decimado.

O problema abordado neste trabalho então é a perda das componentes de alta frequência excluídas no processo de filtragem, levando a uma perda de resolução. Essa perda de resolução resulta em perda de detalhes nas imagens digitais. Ao se interpolar essa imagem, o resultado apresentará serrilhamento nas bordas dos objetos e distorção nas regiões de textura, entre outros efeitos.

## 1.2 Objetivo

O objetivo deste trabalho é buscar uma forma de tentar recuperar as componentes de alta frequência perdidas, restaurando a resolução de imagens digitais no processo de interpolação. Gerchberg [2] afirma que, se o espectro do sinal for extrapolado, é possível alcançar o que ele chama de super-resolução. Acreditando nessa afirmação, o objetivo deste trabalho é fazer um estudo dos algoritmos de extrapolção de sinais para aplicá-los sobre os coeficientes de baixa frequência para se encontrar os coeficientes de alta frequência e verificar a eficiência desse procedimento no processo de interpolação de imagens digitais.

## 1.3 Organização dos Capítulos

### Capítulo 2

No Capítulo 2 é apresentada uma visão geral da teoria de amostragem de sinais enfatizando-se as características resultantes da mudança de taxa de amostragem. Na sequência são apresentados os conceitos mais relevantes no processo de interpolação de sinais digitais e uma variedade de métodos de interpolação, tanto no domínio do tempo quanto no domínio da transformada, identificando-se as características peculiares de cada método.

### Capítulo 3

No Capítulo 3 é introduzido o conceito de super-resolução e apresentado o problema de extrapolção de sinais. São apresentados os operadores ortogonais utilizados no processo de extrapolção. Em seguida são apresentados diversos algoritmos para extrapolção de sinais limitados em banda.

### Capítulo 4

No Capítulo 4 é apresentado o método proposto para tentar alcançar super-resolução de uma imagem, extrapolando-se os coeficientes da transformada utilizada no processo

de interpolação, buscando-se recuperar as componentes de alta frequência. São apresentadas as características específicas do ambiente do problema e é comentada a escolha do algoritmo de extrapolação utilizado, baseado nas limitações do ambiente do problema. O método é explicado em detalhes, comentando-se as características específicas de cada passo.

### **Capítulo 5**

No Capítulo 5 são apresentadas as comparações que foram realizadas e os resultados obtidos. Em seguida é apresentada a conclusão sobre os resultados e comentários adicionais sobre o método.

### **Capítulo 6**

No Capítulo 6 é apresentada a conclusão do trabalho e são indicadas sugestões para a continuidade do mesmo.



# Capítulo 2

## Interpolação

Interpolação é a intercalação de termos, numa série de números ou observações, determinados pelo cálculo baseados nos termos a sua volta. A diferença entre a interpolação e a extrapolação é a posição do dado a ser calculado. Na extrapolação, esse dado se encontra fora dos limites das coordenadas dos dados existentes.

O processo de interpolação possui muitas aplicações e pode ser muito útil para se conseguir mais detalhes de uma série de dados, como por exemplo, imagens digitais.

A seguir será apresentado uma visão da teoria envolvida no processo de interpolação e alguns dos métodos mais importantes.

### 2.1 Teoria de Amostragem

Interpolação pode ser vista teoricamente, como um processo em que uma função  $f$  é reconstruída a partir de suas amostras e em seguida amostrada novamente com uma taxa de amostragem maior. Para entendermos esse procedimento, precisamos conhecer as propriedades e efeitos conseqüentes da amostragem, assim como as regras de amostragem que possibilitem a reconstrução da função  $f$ .

### 2.1.1 Amostragem Periódica

A forma mais comum de representar digitalmente uma função aleatória é através de suas amostras. Uma seqüência de amostras  $x[n]$  é obtida a partir de uma função contínua  $f_c(t)$  de acordo com a relação

$$x[n] = f_c(nT), \quad -\infty < n < \infty \quad (2.1.1)$$

onde  $T$  é o período (ou passo) de amostragem e seu recíproco,  $f_s = 1/T$  é a freqüência de amostragem. A notação de colchetes indica que a seqüência possui valores somente em índices inteiros. O sistema que implementa esta operação é denominado conversor ideal contínuo/discreto, que é geralmente implementado na prática através de um conversor analógico/digital.

Em geral, a operação de amostragem não é inversível, visto que muitas funções contínuas podem produzir uma mesma seqüência de amostras, dependendo da taxa de amostragem. Felizmente, alguns cuidados podem ser tomados no processo de amostragem, a fim de que a função contínua seja reconstruída ou pelo menos uma boa aproximação seja conseguida.

Para analisarmos os efeitos da amostragem, é conveniente representar matematicamente o processo de amostragem como uma modulação da função por um trem de impulsos resultando em  $x_s(t)$  e depois, transformar o resultado em uma seqüência  $x[n]$ . A principal diferença entre  $x_s(t)$  e  $x[n]$  é que  $x_s(t)$  é contínua e possui valor zero exceto em múltiplos de  $T$ . A seqüência  $x[n]$  por outro lado, é indexada pela variável inteira  $n$ , que em efeito, implica em uma normalização do eixo das abcissas, ou seja, não existe nenhuma informação explícita sobre a taxa de amostragem.

### 2.1.2 Representação da Amostragem no Domínio da Freqüência

O processo de modulação no domínio temporal é correspondente a uma operação de convolução no domínio da freqüência, ou seja, entre  $F_c(j\Omega)$  e  $S(j\Omega)$ , as transformadas de Fourier da função  $f_c(t)$  e do trem de impulsos, respectivamente. A transformada de Fourier do trem de impulsos resulta em outro trem de impulsos com impulsos em  $k\Omega_s$ , ou seja, múltiplos inteiros da freqüência de amostragem  $\Omega_s = 2\pi/T$ .

O resultado da convolução de  $F_c(j\Omega)$  e  $S(j\Omega)$  resulta em cópias de  $F_c(j\Omega)$  igualmente espaçada, centralizadas em  $k\Omega_s$ . Para que não haja sobreposição entre os valores não nulos das réplicas de  $F_c(j\Omega)$ , esta deve ser limitada em freqüência máxima  $\Omega_N$  e  $\Omega_s - \Omega_N > \Omega_N$  ou  $\Omega_s > 2\Omega_N$ . Quando esta regra não é seguida, surge o fenômeno conhecido como sobreposição espectral ou "*aliasing*". Esse fenômeno resulta em erros na recuperação da função contínua.

### 2.1.3 Teorema de Amostragem de Nyquist

O fenômeno explicado acima nos leva ao conhecido Teorema de Amostragem de Nyquist, que diz que uma função limitada em banda de freqüência  $\Omega_N$ , para ser completamente recuperada a partir de suas amostras, deve ser amostrada com uma freqüência de amostragem  $f_s = 2\pi/T > \Omega_N$ .

### 2.1.4 Reconstrução de uma Função Limitada em Banda a partir de suas Amostras

Uma vez reconhecido que uma função pode ser recuperada sem erros se forem tomadas amostras suficientes, e isto implica que a função deve ser limitada em banda, podemos então proceder para o processo de reconstrução. Como vimos, o processo de amostragem

periódica resulta em réplicas do espectro da função no eixo das frequências. Portanto, para recuperarmos a função contínua, precisamos somente da réplica do espectro localizado na origem, eliminando as demais réplicas. Isto pode ser conseguido utilizando um filtro passa baixas ideal contínuo com frequência de corte  $\Omega_N$ .

O processo de filtragem no domínio da transformada de Fourier corresponde à multiplicação do espectro por uma função que possui valores unitário na faixa de frequência desejada e valores zero na faixa de frequências a ser rejeitada. No domínio temporal esta operação corresponde a uma convolução entre o sinal discreto e a resposta impulsiva do filtro. A resposta impulsiva é a transformada inversa do filtro no domínio da frequência. No caso do filtro passa-baixas ideal, a resposta impulsiva é a função sinc, definida por

$$h_r(t) = \text{sinc}(t) = \frac{\sin(\pi t/T)}{\pi t/T}. \quad (2.1.2)$$

Esta função tem muita importância no processo de análise de interpolação, sendo considerado o interpolador ideal do ponto de vista teórico. A Figura 2.1 mostra um função limitada em banda e suas amostras e a interpolação através da função sinc.

### 2.1.5 Aumento da Taxa de Amostragem por um Fator Inteiro (Processamento Discreto)

Apesar de podermos considerar a interpolação de um sinal discreto através da reconstrução da função contínua, seguido de uma nova amostragem a uma taxa superior, também é possível aumentar a taxa de amostragem utilizando somente procedimentos no campo discreto. Esse procedimento é muito importante, visto que não seria prático implementar a reconstrução da função contínua em um circuito eletrônico.

Para aumentarmos a taxa de amostragem do sinal por um valor inteiro  $L$  tal que  $T' = T/L$  ( $T'$  é o novo passo de amostragem e  $T$  é o passo de amostragem atual), o

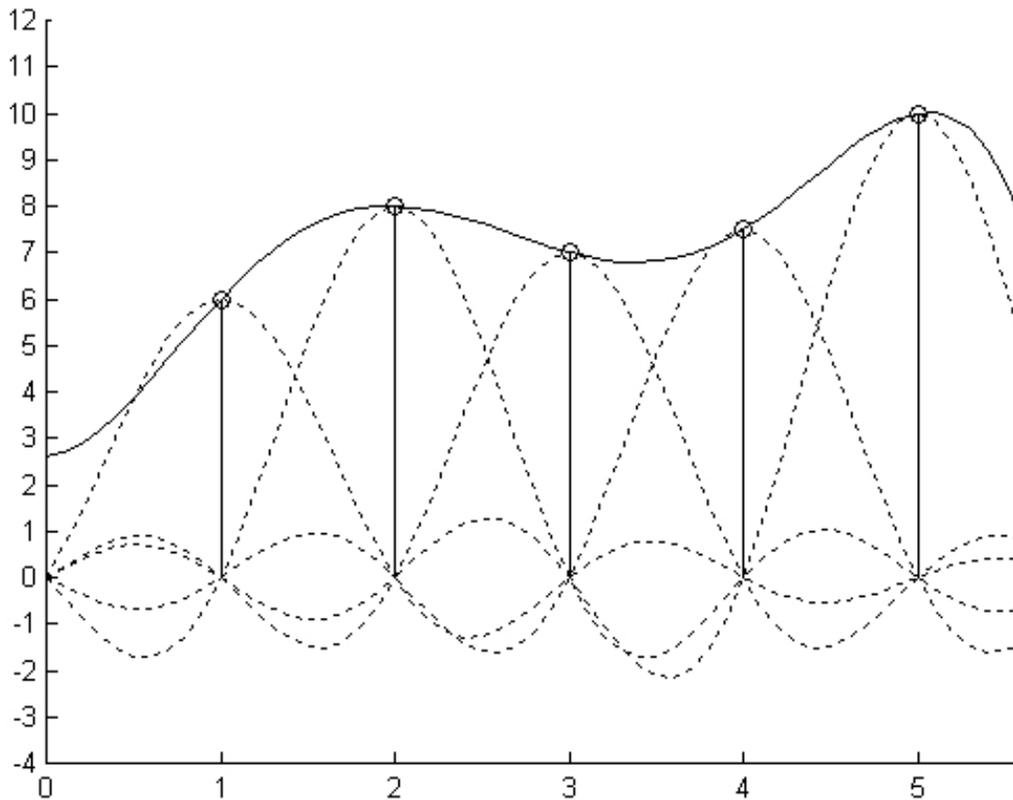


Figura 2.1: Interpolação da função  $x$  a partir de suas amostras.

processo consiste em interpolar o sinal discreto com  $L$  zeros entre as amostras e em seguida filtrar o sinal resultante com um filtro passa-baixas discreto de frequência de corte  $\pi/L$  e ganho  $L$ .

Para mudar a taxa de amostragem por um valor não inteiro  $T' = TM/L$ , é necessário primeiro aumentar a taxa de amostragem por um fator  $L$  e depois reduzir por um fator  $M$ . Porém é preciso tomar um certo cuidado no processo de redução da taxa de amostragem. Para entendermos porque, é preciso analisar o processo no domínio da frequência. Quando o sinal é interpolado com zeros no processo de aumento da taxa de amostragem, o espectro do sinal comprime, diminuindo a frequência normalizada  $\Omega_N$ . Já no processo de redução

da taxa de amostragem, que é feito amostrando novamente o sinal discreto  $x[n]$  tal que  $x_d[n] = x[Mn]$ , o espectro do sinal expande, podendo haver sobreposição espectral caso a taxa de redução seja maior do que  $\Omega_N$ . Para evitar a sobreposição espectral neste caso, é necessário filtrar o sinal com um filtro passa-baixas discreto com frequência de corte  $\pi/M$  e ganho 1. Esse fato faz com que conteúdo espectral das frequências altas sejam perdidos, não podendo ser recuperados no processo de aumento de taxa de amostragem.

## 2.1.6 Considerações Gerais Sobre Imagens Digitais

Visto os efeitos da amostragem em um sinal contínuo, deve-se levantar algumas considerações desses efeitos no caso específico de imagens digitais, lembrando-se que mesmo dentro desse conjunto de sinais, ainda existe outros subconjuntos (ex. imagens naturais, sintéticas, médicas, de sensoriamento remoto, etc.) com características mais específicas que fogem do interesse deste trabalho.

As bordas dos objetos são as características que contém maior quantidade de informação em uma imagem. Elas são extremamente localizadas, portanto apresentam conteúdo espectral nas frequências altas.

As imagens de um modo geral são consideradas limitadas em banda, porque são amostradas a uma taxa elevada, dada uma certa escala.

Nenhum sinal pode ser ao mesmo tempo finito (truncado no espaço) e limitado em frequência. Esse fato gera o problema com os limites da área da imagem que geralmente é retangular e finita. Também gera problemas quando a imagem é processada em blocos.

Quando a escala da imagem contínua aumenta, como no caso onde é feito um "zoom", é provável que surgirão mais detalhes perceptíveis ao olho, que resulta no aumento de conteúdo espectral de altas frequências. Deste modo não é possível determinar com antecedência a frequência máxima  $\Omega_N$  do sinal a ser amostrado.

Para evitar o problema de sobreposição espectral, na prática, as imagens são filtradas antes da amostragem, tendo seu conteúdo espectral de altas frequências perdidos. Esse mesmo problema ocorre em aplicações onde a imagem digitalizada é decimada. Caso a original não esteja disponível, ela não pode ser recuperada perfeitamente a partir da imagem reduzida disponível

Limites de características psico-visuais (olho humano) são estudados para determinar o mínimo de informação necessária para que uma qualidade seja considerada aceitável a uma dada aplicação. Entre as características mais relevantes, podemos citar a resolução espacial, geralmente dada em pontos por polegadas (dots per inch - dpi), e os níveis de intensidade da luminosidade (luminância). Imagens monocromáticas digitais de boa qualidade geralmente apresentam valores mínimos de 300 dpi e 256 níveis de intensidade em cada ponto (do preto ao branco passando pelos tons de cinza).

De modo geral as variáveis de amostragem geralmente são impostas pelo hardware de aquisição (scanner, câmera digital, placa digitalizadora, etc.), não havendo muita flexibilidade. Lembrando que a qualidade da imagem digitalizada está diretamente ligada ao custo do hardware tanto para a aquisição quanto para o armazenamento.

## **2.2 Interpolação no Domínio Espacial**

### **2.2.1 Método Clássico**

Apesar da interpolação de sinais possuir diversos algoritmos diferentes com diferentes formas de implementação, utilizaremos o algoritmo linear para ilustrar os diversos algoritmos de interpolação no domínio espacial, mantendo uma forma única e concisa. O algoritmo

linear apresenta a seguinte equação:

$$f(\mathbf{x}) = \sum_{\mathbf{k} \in \mathbb{Z}^q} f_{\mathbf{k}} \varphi_{int}(\mathbf{x} - \mathbf{k}), \quad \forall \mathbf{x} = (x_1, x_2, \dots, x_q) \in \mathbb{R}^q \quad (2.2.1)$$

onde um valor interpolado  $f(\mathbf{x})$  em alguma coordenada  $\mathbf{x}$  em um espaço de dimensão  $q$  é expresso como uma combinação linear das amostras  $f_{\mathbf{k}}$  avaliado em coordenadas inteiras  $\mathbf{k} = (k_1, k_2, \dots, k_q) \in \mathbb{Z}^q$ , sendo os pesos dados pelos valores da função  $\varphi_{int}(\mathbf{x} - \mathbf{k})$ . Valores típicos da dimensão do espaço correspondem a imagens bidimensionais (2D), com  $q = 2$ , e volumes tridimensionais (3D), com  $q = 3$ . O passo de amostragem é assumido ser unitário sem perda de generalidade. A função  $\varphi_{int}$  deve satisfazer a propriedade de interpolação, ou seja, deve ser zero para todos os argumento inteiros exceto a origem, onde deve assumir o valor unitário. Um exemplo clássico de função de síntese é a função sinc, no qual todas as funções sintetizadas são limitadas em banda de frequência.

Como foi expresso na Equação 2.2.1, a soma é realizada sobre todas as coordenadas inteiras  $\mathbf{k} \in \mathbb{Z}^q$ , cobrindo toda a grade cartesiana de locais de amostragem, havendo ou não uma amostra  $f_{\mathbf{k}_0}$ , fisicamente adquirida em algum  $\mathbf{k}_0$  específico. No entanto, na prática, o número de amostras conhecidas é sempre finito, portanto, de modo a satisfazer a convenção formal em 2.2.1, temos que estender esse número finito para o infinito, estabelecendo condições de bordas apropriadas sobre o sinal interpolado usando, por exemplo, simetria par [1, 3]. Agora que todos os valores de  $f_{\mathbf{k}}$  foram definidos, a soma é realizada sobre os infinitos valores.

Considere a avaliação de 2.2.1 no caso específico quando todas as coordenadas de  $\mathbf{x} = \mathbf{k}_0$  são inteiros:

$$f_{\mathbf{k}_0} = \sum_{\mathbf{k} \in \mathbb{Z}^q} f_{\mathbf{k}} \varphi_{int}(\mathbf{k}_0 - \mathbf{k}), \quad \forall \mathbf{k}_0 \in \mathbb{Z}^q \quad (2.2.2)$$

Esta equação é conhecida como a restrição de interpolação e pode ser percebido que 2.2.2

é uma convolução discreta [1], podendo ser rescrita como:

$$f_{\mathbf{k}_0} = (f * p)_{\mathbf{k}_0}, \quad \forall \mathbf{k}_0 \in \mathbb{Z}^q \quad (2.2.3)$$

onde  $*$  é o símbolo da operação de convolução e introduzimos a notação  $p_{\mathbf{k}} = \varphi_{int}(\mathbf{k})$  para enfatizar o fato que apenas discutiremos convolução entre seqüências que tenham a propriedade de serem discretas. Portanto 2.2.1 não é uma convolução discreta neste senso, porque  $\varphi_{int}$  é avaliada possivelmente em valores não inteiros. De agora em diante,  $f_{\mathbf{k}}$  será usada para descrever as amostras de  $f$  e  $p_{\mathbf{k}}$  para descrever os valores tomados por  $\varphi_{int}$  para argumentos inteiros. Obviamente temos que 2.2.2 é equivalente a  $p_{\mathbf{k}} = \delta_{\mathbf{k}}$  onde  $\delta_{\mathbf{k}}$  é o símbolo de Kronecker [1] que é caracterizado por ser uma seqüência com valores igual a zero em todas as coordenadas exceto na origem que possui valor unitário (valor central  $\delta_0 = 1$ ). Esta função de argumentos inteiros tem o papel de elemento neutro na operação de convolução discreta.

## 2.2.2 Método Generalizado

Podemos considerar como um algoritmo alternativo a forma:

$$f(\mathbf{x}) = \sum_{\mathbf{k} \in \mathbb{Z}^q} c_{\mathbf{k}} \varphi(\mathbf{x} - \mathbf{k}), \quad \forall \mathbf{x} \in \mathbb{R}^q \quad (2.2.4)$$

A diferença crucial entre a formulação clássica 2.2.1 e a formulação generalizada 2.2.4 é a introdução dos coeficientes  $c_{\mathbf{k}}$  no lugar das amostras  $f_{\mathbf{k}}$ , oferecendo novas possibilidades no sentido de que a interpolação pode ser realizada em dois passos separados: primeiramente o cálculo dos coeficientes  $c_{\mathbf{k}}$  a partir das amostras  $f_{\mathbf{k}}$ , em seguida o cálculo dos valores desejados de  $f(\mathbf{x})$  a partir dos coeficientes  $c_{\mathbf{k}}$ . O benefício desta separação é permitir uma escolha maior de funções de síntese, algumas com propriedades melhores do que as disponíveis no restrito caso clássico, onde  $c_{\mathbf{k}} = f_{\mathbf{k}}$ . A desvantagem aparente é a necessidade

de um passo adicional. Veremos mais tarde que esta desvantagem é altamente compensada pelo ganho em qualidade resultante da maior opção de escolha de funções de síntese.

Na abordagem apresentada, a função de síntese não é necessariamente de suporte finito nem requer que satisfaça a propriedade de interpolação. No entanto, é essencial associar um valor específico para as amostras  $f_{\mathbf{k}}$  desconhecidas que estão fora do intervalo conhecido. Na prática, esta associação é implícita, e os dados desconhecidos são assumidos serem a extensão periódica dos dados após espelhamento (simetria par).

### Determinação dos Coeficientes

Suponha que queiramos reforçar a condição de restrição de interpolação semelhante a 2.2.2, no contexto de interpolação generalizada. Considerando novamente apenas argumentos inteiros  $\mathbf{x} = \mathbf{k}_0$ , temos

$$f_{\mathbf{k}_0} = \sum_{\mathbf{k} \in \mathbb{Z}^q} c_{\mathbf{k}} p_{\mathbf{k}_0 - \mathbf{k}}, \quad \forall \mathbf{k}_0 \in \mathbb{Z}^q, \quad (2.2.5)$$

onde  $p_{\mathbf{k}} = \varphi(\mathbf{k})$ . Dada alguma função  $\varphi$  que é conhecida a priori, esta expressão não é mais do que um sistema de equações lineares em termos dos coeficientes  $c_{\mathbf{k}}$  desconhecidos. De acordo com a Equação 2.2.5, a dimensão desse sistema é infinito, tanto com respeito ao número de equações (porque todos os argumentos  $\mathbf{k} \in \mathbb{Z}^q$  são considerados) quanto ao número de coeficientes desconhecidos (porque todos os índices  $\mathbf{k} \in \mathbb{Z}^q$  são considerados na soma).

Mas na prática a quantidade de amostras  $\mathbf{k}_0$  conhecidas é finita, o que limita o número de equações, e ao mesmo tempo, a função  $\varphi$  é restrita a ser de suporte finito, o que limita o número de coeficientes desconhecidos. O problema pode ser escrito da forma  $c = P^{-1}f$ , onde vários esforços já foram realizados (ex.: [4]) com o objetivo de desenvolver técnicas eficientes para inverter a matriz  $P$  de funções de síntese  $\varphi$  específicas.

Outra estratégia para reduzir o tamanho do sistema é reconhecer que 2.2.5 é uma equação de convolução discreta que pode ser escrita como:

$$f_{\mathbf{k}_0} = (c * p)_{\mathbf{k}_0}, \quad \forall \mathbf{k}_0 \in \mathbb{Z}^q \quad (2.2.6)$$

Portanto a seqüência infinita de coeficientes  $\{c_{\mathbf{k}}\}$  pode ser obtida convoluindo a seqüência infinita  $\{f_{\mathbf{k}}\}$  pela convolução inversa  $(p)^{-1}$ , que é simplesmente uma seqüência de números  $\{(p)^{-1}_{\mathbf{k}}\}$  tal que  $(p * (p)^{-1})_{\mathbf{k}} = \delta_{\mathbf{k}}$ . Esta seqüência possui definição única e geralmente existe nos casos de interesse. Convoluindo ambos os lados de 2.2.6 por  $(p)^{-1}_{\mathbf{k}}$ , temos que:

$$c_{\mathbf{k}_0} = ((p)^{-1} * f)_{\mathbf{k}_0}, \quad \forall \mathbf{k}_0 \in \mathbb{Z}^q \quad (2.2.7)$$

Visto que convolução discreta pode ser implementada através de uma operação de filtragem digital, isto sugere que pode ser uma solução alternativa para a inversão de matriz utilizada na determinação da seqüência de coeficientes  $\{c_{\mathbf{k}}\}$  necessária para reforçar a restrição 2.2.5 que é desejável. Um algoritmo muito eficiente para realizar esse cálculo para uma classe de funções de síntese pode ser encontrado em [5, 6].

Comparando-se a equação 2.2.1 com a equação 2.2.4 nota-se que a interpolação clássica é um caso especial da interpolação generalizada sendo  $c_{\mathbf{k}} = f_{\mathbf{k}}$  e  $\varphi = \varphi_{int}$ . Pode ser mostrado que o contrário também é verdade, já que é possível interpretar a interpolação generalizada como um caso da interpolação clássica. Para tanto, deve se determinar a função interpoladora  $\varphi_{int}$  a partir de sua contrapartida  $\varphi$ . A partir das equações 2.2.4 e 2.2.7, temos:

$$f(\mathbf{x}) = \sum_{\mathbf{k}_1 \in \mathbb{Z}^q} ((p)^{-1}_{\mathbf{k}} * f)_{\mathbf{k}_1} \varphi(\mathbf{x} - \mathbf{k}_1) = \sum_{\mathbf{k}_1 \in \mathbb{Z}^q} \sum_{\mathbf{k}_2 \in \mathbb{Z}^q} (p)^{-1}_{\mathbf{k}_2} f_{\mathbf{k}_1 - \mathbf{k}_2} \varphi(\mathbf{x} - \mathbf{k}_1) \quad (2.2.8)$$

A função  $\varphi_{int}$ , que fica escondida atrás da função não interpoladora  $\varphi$ , pode finalmente ser determinada por:

$$\varphi_{int}(\mathbf{x}) = \sum_{\mathbf{k} \in \mathbb{Z}^q} (p)^{-1}_{\mathbf{k}} \varphi(\mathbf{x} - \mathbf{k}) \quad (2.2.9)$$

É importante perceber que esta equivalência permite o manuseio exato e eficiente de uma função interpoladora  $\varphi_{int}$  de suporte infinito através da realização de operações apenas com funções  $\varphi$  não interpoladoras de suporte finito. A liberdade para escolher uma função de síntese é muito maior depois que a restrição de interpolação é removida. Isto possibilita o uso de funções  $\varphi$  que oferecem melhor desempenho do que qualquer função explicitamente interpoladora  $\varphi_{int}$  (a um mesmo custo computacional).

### 2.2.3 Teoria de Aproximação

Até então citamos as funções  $\varphi$  e  $\varphi_{int}$  de forma genérica sem defini-las ou especificá-las. Iremos agora citar as características básicas a estas funções e mostraremos como é possível medir sua eficiência.

**Separabilidade:** determina que os dados possam ser processados em suas dimensões separadamente, ou seja a função não depende dos dados de uma dimensão para calcular os dados em outra dimensão. Por exemplo, uma função separável permite que uma imagem bidimensional possa ser processada somente nas linhas e então somente nas colunas. Esta propriedade resulta em uma menor complexidade computacional. Para ilustrar, consideremos uma função com região de suporte 5. Para o cálculo de dados tridimensionais seriam necessários  $5^3 = 125$  amostras para uma função não separável e somente  $3 \times 5 = 15$  amostras para uma função separável.

**Simetria:** uma função simétrica é tal que  $\varphi(\mathbf{x}) = \varphi(-\mathbf{x})$ . Esta propriedade resulta que o processo de interpolação, que pode ser considerado como um processo de filtragem digital, possua uma resposta de fase sem degradação. A preservação da fase é uma propriedade muito importante em processamento de imagens.

**Divisão da unidade:** é definida pela equação:

$$1 = \sum_{\mathbf{k} \in \mathbb{Z}^q} \varphi_{int}(\mathbf{x} - \mathbf{k}) \quad (2.2.10)$$

e resulta que a função interpoladora possui a propriedade de reproduzir a constante, ou seja para um segmento de sinal composto de valores constantes  $f_{\mathbf{k}} = f_0$ , as coordenadas interpoladas possuirão o mesmo valor. Esta propriedade também é muito importante no processamento de imagens, visto que a maioria das imagens digitais tem sua energia concentrada nas baixas frequências assim como os sinais de valores constantes.

Para avaliar a eficiência de uma função no processo de interpolação realiza-se o seguinte experimento:

1. Obter uma função integrável  $f(x)$  e escolher um passo de amostragem  $h > 0$ ;
2. Criar um conjunto de amostras  $f(hk)$ ;
3. Desse conjunto, utilizando 2.2.1 ou 2.2.4, construir uma função interpolada  $f_h(\mathbf{x}) = \sum c_{\mathbf{k}} \varphi(\frac{1}{h}\mathbf{x} - \mathbf{k})$ ;
4. Comparar  $f$  e  $f_h$  usando alguma norma, como por exemplo norma média quadrática (ou  $L_2$ )  $\varepsilon^2(h) = \|f - f_h\|_{L_2}^2$ ;

Quando o passo de amostragem  $h$  fica menor, mais detalhes de  $f$  podem ser capturados. Então, é razoável desejar que o erro de aproximação  $\varepsilon(h)$  fique menor também. As questões fundamentais são: quanto menor, qual a influencia tem a escolha da função  $f$  e que papel tem a função de síntese  $\varphi$  que é usada para construir  $f_h$ ?

Estas perguntas são respondida apresentando uma fórmula de predição do erro de aproximação no domínio da transformada de Fourier [7]:

$$\eta^2(h) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} |\hat{f}(\omega)|^2 E(\omega h) d\omega_1 \cdots d\omega_q \quad (2.2.11)$$

onde  $\hat{f}(\omega)$  é a transformada de Fourier de uma função arbitrária  $f(x)$ , e onde  $E$  é um núcleo de erro que depende somente da função de síntese e dado por:

$$E(\omega) = \frac{(|\sum_{\mathbf{k} \in \mathbb{Z}_*^q} \hat{\varphi}(\omega + 2\pi\mathbf{k})|^2 + |\sum_{\mathbf{k} \in \mathbb{Z}_*^q} \hat{\varphi}(\omega + 2\pi\mathbf{k})|^2)}{|\sum_{\mathbf{k} \in \mathbb{Z}^q} \hat{\varphi}(\omega + 2\pi\mathbf{k})|^2} \quad (2.2.12)$$

A equivalência  $\varepsilon = \eta$  se mantém para funções limitadas em banda . Para as funções que não pertencem a esta classe, o erro estimado  $\eta(h)$  deve ser entendido como o erro médio sobre todos o conjunto de amostras  $f(h\mathbf{k} + \Delta)$ , onde  $\Delta = (\Delta_1, \Delta_2, \dots, \Delta_q)$  é algum termo de fase com  $\Delta_i \in [0, h[$ . Quando  $q = 1$ , para funções limitadas em banda e quando a função de síntese  $\varphi_{int}$  é interpoladora, esse núcleo de erro se reduz ao núcleo proposto em [8].

Observando-se a equação 2.2.11 vemos que um decréscimo no passo de amostragem  $h$  resultará em um decréscimo do argumento de  $E$ . Como a função  $f$  é arbitrária, e já que se deseja que o erro de aproximação  $\varepsilon(h)$  seja minimizado para um passo de amostragem minimizado, o núcleo de erro também deve minimizar na origem. Portanto é interessante desenvolver  $E$  em uma série de Mac-Laurin em torno da origem (para simplificar, consideraremos somente o caso unidimensional). Já que esta função possui simetria par ( $E(x) = E(-x)$ ), apenas os fatores pares precisam ser considerados e o desenvolvimento fica:

$$E(\omega) = \sum_{n \in N} \frac{E^{(2n)}(0)}{(2n)!} \quad (2.2.13)$$

onde  $E^{(2n)}$  é a 2n-ésima derivada do núcleo de erro. Por definição, a ordem de diferenciação  $L$  para o qual  $E^{(2L)}(0) \neq 0$  e  $E^{(2n)}(0) = 0 \quad \forall n \in [0, L - 1]$ , é chamado de ordem de aproximação de  $\varphi$ .

Portanto para uma função de síntese de ordem  $L$ , a expansão infinita de Mac-Laurin é dada por:

$$E(\omega) = (C_\varphi)^2 \omega^{2L} + \left( \sum_{n=L+1}^{\infty} \frac{E^{(2n)}(0)}{(2n)!} \right) \quad (2.2.14)$$

onde  $C_\varphi$  depende somente de  $\varphi$ . Quando o passo de amostragem é pequeno o suficiente, podemos ignorar os termos de maior ordem da expansão. A aplicação em 2.2.11 da expressão de  $E$  resultante leva a:

$$\eta^2(h) = (C_\varphi)^2 h^{2L} \left( \frac{1}{2\pi} \int_{-\infty}^{\infty} |\omega^L \hat{f}(\omega)|^2 d\omega \right) \quad h \rightarrow 0, \quad (2.2.15)$$

onde a expressão entre parênteses pode ser reconhecida como a norma da  $L$ -ésima derivada da função  $f$  suave da qual começamos.

Finalmente para uma função de síntese de ordem de aproximação  $L$ , temos que:

$$\eta(h) = \|f - f_h\|_{L_2} = C_\varphi h^L \|f^{(L)}\|_{L_2} \quad h \rightarrow 0 \quad (2.2.16)$$

Esse resultado mostra que podemos associar a qualquer  $\varphi$  um número  $L$  e uma constante  $C_\varphi$  tal que o erro de aproximação  $\varepsilon$  predito por  $\eta$  decai com o inverso do passo de amostragem  $h$ , quando  $h$  é pequeno o suficiente. Visto que o decréscimo pode ser descrito com sendo da ordem  $O(h^L)$ , o número  $L$  é chamado de ordem de aproximação da função de síntese  $\varphi$ . Esse processo independente da função  $f$ .

### Equivalência de Strang-Fix

Caso o interesse seja somente na ordem de aproximação  $L$  da função de síntese  $\varphi$ , sem levar em consideração os detalhes de  $E$ , não é necessário o cálculo explícito de 2.2.12. Strang-Fix [9] propôs uma série de condições que são equivalentes a 2.2.16. O interessante destas condições equivalentes é que elas podem ser facilmente testadas. Elas são válidas para todas as funções de síntese com decaimento suficiente; a função sinc é um dos raros casos onde estas condições não são satisfeitas. Mencionaremos três condições equivalentes para o caso unidimensional.

1. Zeros de  $L$ -ésima ordem no domínio de Fourier

$$\left\{ \begin{array}{l} \hat{\varphi}(0) = 1 \\ \hat{\varphi}^{(n)}(2\pi k) = 0 \quad k \in \mathbb{Z}_* \quad n \in [0, L-1] \end{array} \right. ;$$

2. Reprodução de todos os monômios de grau  $n \leq N = L - 1$

$$\forall n \in [0, N] \quad \exists \{ \dots, c_{-1}^{(n)}, c_0^{(n)}, c_1^{(n)}, \dots \} \quad | \quad \sum_{k \in \mathbb{Z}} c_k^{(n)} \varphi(x - k) = x^n$$

3. Momentos discretos

$$\sum_{k \in \mathbb{Z}} (x - k)^n \varphi(x - k) = \mu_n \quad \forall n \in [0, L - 1]$$

onde  $\mu_n$  depende somente de  $n$

Sob hipóteses fracas, pode-se dizer que qualquer umas destas condições é equivalente a:

$$\varepsilon(h) \leq \text{const} \times h^L \|f^{(L)}\|_{L_2}$$

## Reprodução de Polinômios

A ordem de aproximação  $L$  só é relevante quando a função é super-amostrada o que é comum em imagens digitais cotidianas. A idéia trazida pela teoria de Strang-Fix é o fato de que é equivalente pensar tanto em termos de conteúdo de frequência quanto em termos de polinômios, desconsiderando-se alguns detalhes técnicos. Intuitivamente, é razoável pensar que quando a função amostrada é suave a uma escala  $h$ , podemos modelá-la por polinômios sem introduzir muito erro. Isto foi formalizado como o teorema de aproximação de Weierstrass. O que a teoria de Strang-Fix nos diz é que existe uma relação precisa entre a ordem de aproximação e o grau máximo do polinômio que a função de síntese pode reproduzir exatamente. Por exemplo, começamos esta discussão com a estimativa teórica da qualidade de qualquer função de síntese investigando a reprodução da constante. Completamos agora um ciclo completo e voltamos à reprodução de polinômios, do qual a constante é um caso particular.

## Regularidade

Considere que temos uma função suave  $f$  e as suas amostras. A partir das amostras  $f(hk)$  e por interpolação reconstruímos uma  $f_h$  que é uma aproximação de  $f$ . Visto que  $f$  é suave, intuitivamente deseja-se que  $f_h$  seja suave também. Intuitivamente, também se imagina, que para assegurar esta suavidade,  $\varphi$  seja suave. Estas considerações podem levar aos seguintes silogismos:

1. A ordem de aproximação  $L$  requer a reprodução de monômios de grau  $L - 1$ ;
2. Monômios de grau  $N = L - 1$  são funções que são pelo menos  $N$  vezes deriváveis;
3. Se requer uma função de síntese derivável  $N$  vezes para se conseguir uma  $L$ -ésima ordem de aproximação.

Porém a intuição algumas vezes pode enganar. Uma função que é pelo menos  $n$  vezes derivável é dita ter regularidade  $C^n$ . Uma função contínua, mas no entanto não derivável é rotulada de  $C^0$ , enquanto que uma função descontínua é dita não possuir regularidade. Alguns autores insistem que a regularidade de uma função de síntese é importante [4]. Isto pode ser verdade quando a diferenciação de  $f_h$  é necessária, mas derivar uma função mais do que duas vezes é incomum em aplicações atuais. Frequentemente somente o gradiente de uma imagem é necessário, portanto, não é realmente necessário limitar a escolha da função de síntese àquelas com alto grau de regularidade. Na realidade, a conclusão do silogismo acima é incorreto e uma função de síntese não precisa ser derivável  $N$  vezes para ter uma ordem de aproximação  $N + 1$ .

Para mostrar um exemplo que ilustra a afirmação acima, podemos citar Schaum [10] que propôs uma família de funções interpoladoras inspirada na interpolação Lagrangiana. Um membro desta família é composto de pedaços de polinômios emendados. Apesar

do fato de que esta função é descontínua, ela possui uma ordem de aproximação  $L = 3$ . Portanto, uma soma linear destas funções com coeficientes bem escolhidos é capaz de reproduzir exatamente uma constante, uma rampa e uma quadrática. Neste caso específico, a função de síntese é interpoladora.

### Exemplo de Núcleo de Aproximação

A comparação entre a função sinc com a interpolação do "vizinho-mais-próximo" nos dá um bom caso de teste para compreender o poder de predição de  $E$ . Como a transformada de Fourier da função sinc é simplesmente uma função retangular que tem valores unitários no intervalo  $[-\pi, \pi]$  e zero fora desse intervalo, o denominador em 2.2.12 é unitário para qualquer frequência  $\omega$  (porque um único termo do domínio principal contribui para a soma infinita). Por outro lado, já que a soma é realizada sobre inteiros não nulos  $k \in Z_*$ , não há termos que contribuem no numerador e  $E$  é zero no domínio principal  $[-\pi, \pi]$ . Tomando um raciocínio similar, o valor de  $E$  é dois fora do domínio principal. Isto corresponde ao fato bem conhecido de que a função de síntese sinc pode representar uma função limitada em banda de frequência sem erros e ao mesmo tempo sofre muito com a sobreposição espectral (alising) quando a função não é limitada em banda de frequência.

A interpolação por "vizinho-mais-próximo" é caracterizada por uma função de síntese retangular, cuja transformada de Fourier é uma função sinc (esta situação é o inverso do caso anterior). Infelizmente, a expressão 2.2.12 agora é menos maleável. Podemos porém traçar uma estimativa numérica de 2.2.11. Os núcleo de aproximação resultantes estão representados na Figura 2.2.

Na origem, quando  $\omega = 0$ , é visível pela Figura 2.2 que tanto a função sinc quanto a interpolação por "vizinho-mais-próximo" não produzem nenhum erro, portanto elas reproduzem a constante. De modo mais genérico, o grau de "achatamento" na origem dá

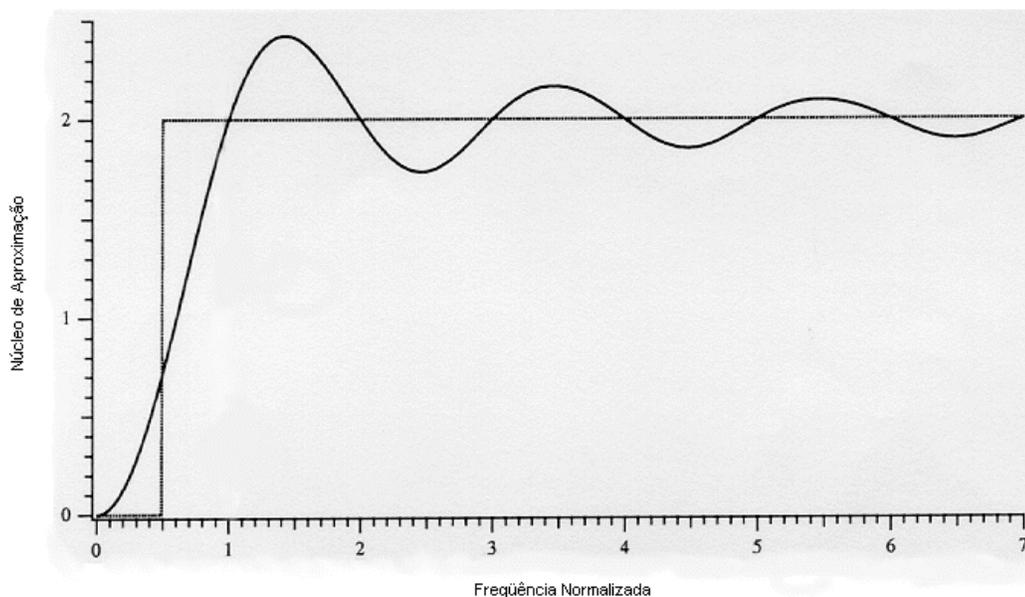


Figura 2.2: Núcleos de aproximação em Fourier. Sinc X Vizinho mais próximo

diretamente a ordem de aproximação da função de síntese - sendo uma linha reta no caso da função sinc, esse achatamento é infinito assim como a ordem de aproximação. Quando  $\omega$  cresce, a interpolação por "vizinho-mais-próximo" é pior do que a interpolação por sinc, que é perfeita até a frequência de Nyquist. O que não se imagina, e se pode perceber pela Figura 2.2 é que a interpolação por "vizinho-mais-próximo" é melhor do que a função sinc para algumas (não todas) funções que não são limitadas em banda.

## 2.2.4 Casos Particulares

A seguir apresentaremos algumas funções de síntese mais relevantes, incluindo algumas de suas propriedades, sua forma explícita, região de suporte, aparência e forma interpoladora equivalente quando não forem interpoladoras. Também incluímos a ordem de aproximação, núcleo de aproximação e regularidade.

### Vizinho-Mais-Próximo

A função de síntese associada à interpolação por vizinho-mais-próximo é a mais simples de todas, sendo um pulso quadrado. Sua região de suporte é unitário, é interpoladora e satisfaz a divisão da unidade contanto que tenha uma pequena assimetria introduzida nas bordas do pulso quadrado. A ordem de aproximação é um (reproduz sem erro no máximo a constante). É descontínua, portanto não tem regularidade. Sua expressão é dada por:

$$\varphi^0(x) = \begin{cases} 0 & x < \frac{-1}{2} \\ 1 & \frac{-1}{2} \leq x < \frac{1}{2} \\ 0 & \frac{1}{2} \leq x \end{cases} \quad (2.2.17)$$

A maior vantagem desta função de síntese é sua simplicidade, que resulta na implementação mais eficiente de todas. De fato, para cada coordenada  $x$  onde se deseje computar o valor da função interpolada  $f$ , há apenas uma amostra  $f_{\mathbf{k}}$  que contribui, não importa quantas dimensões  $q$  estão envolvidas. O preço pago é a severa perda de qualidade.

### Linear

A interpolação linear possui grande popularidade porque a complexidade de sua implementação é muito baixa, perdendo somente para a interpolação por vizinho-mais-próximo. Além do mais, muitos consideram que ela satisfaz o princípio de Occam por ser a interpolação mais simples que alguém possa imaginar, capaz de construir uma função contínua  $f$  a partir de uma seqüência de amostras discretas. Ela é feita da convolução (contínua) de um pulso quadrado com ele mesmo, que resulta em um triângulo, algumas vezes chamado de função de chapéu ou de tenda. Sua região de suporte é de duas unidades, é interpoladora e sua ordem de aproximação é 2 (reproduz linhas retas de qualquer rampa finita). Sua regularidade é  $C^0$ , que expressa que ela é contínua mas não é derivável. No caso

unidimensional, esta interpolação requer no máximo duas amostras para produzir um valor interpolado. No caso bidimensional, também conhecido como interpolação bilinear, sua implementação separável requer quatro amostras e seis no caso tridimensional (oito amostras na implementação não separável onde é chamado de interpolação trilinear [11, 12]). A expressão da função de síntese para o caso unidimensional é dada por:

$$\beta^1(x) = \begin{cases} 1 - |x| & |x| < 1 \\ 0 & 1 \leq x \end{cases} \quad (2.2.18)$$

## B-splines

Existe uma família inteira de funções de síntese feitas de B-splines. Por convenção sua representação simbólica é  $\beta^n$ , onde  $n \in \mathbb{N}$  não é uma potência, mas sim um índice chamado de grau da spline. Estas funções de síntese são compostas por pedaços de polinômios de grau  $n$ . Elas são globalmente simétricas e  $(n - 1)$  vezes continuamente deriváveis. Portanto, sua regularidade é  $C^{n-1}$ . Suas expressões são dadas por:

$$\beta^0(x) = \begin{cases} 1 & |x| < 1/2 \\ 1/2 & |x| = 1/2 \\ 0 & |x| > 1/2 \end{cases} \quad (2.2.19)$$

e

$$\beta^n(x) = \sum_{k=0}^{n+1} \frac{(-1)^k (n+1)}{(n+1-k)!k!} \left( \frac{n+1}{2} + x - k \right)_+^n \quad \forall x \in \mathbb{R}, \quad \forall n \in \mathbb{N}_* \quad (2.2.20)$$

onde por definição:

$$(x)_+^n = (\max(0, x))^n \quad n > 0 \quad (2.2.21)$$

Tanto a região de suporte quanto a ordem de aproximação destas funções são um a mais do que o grau, ou seja  $n + 1$ . Elas possuem muitas outras propriedades interessantes que estão fora do escopo deste trabalho, exceto talvez o fato de que uma B-spline derivada pode ser computada recursivamente por:

$$\frac{d}{dx} \beta^n(x) = \beta^{n-1}(x + 1/2) - \beta^{n-1}(x - 1/2) \quad n > 0 \quad (2.2.22)$$

Portanto, computar o gradiente exato de um sinal dado por uma seqüência discreta de coeficientes de interpolação  $\{c_{\mathbf{k}}\}$  pode ser feito da seguinte maneira:

$$\frac{d}{dx}f(x) = \sum_{\mathbf{k} \in \mathbb{Z}} c_{\mathbf{k}} \frac{d}{dx} \beta^n(\mathbf{x} - \mathbf{k}) = \sum_{\mathbf{k} \in \mathbb{Z}} (c_{\mathbf{k}} - c_{\mathbf{k}-1}) \beta^{n-1}(\mathbf{x} - \mathbf{k} + 1/2) \quad (2.2.23)$$

onde a derivação contínua de  $n$  vezes assegura que a função resultante é suave quando  $n \geq 3$ , ou pelo menos contínua quando  $n \geq 2$ .

### **Grau $n = 0$**

A B-spline de menor grau  $n = 0$  é praticamente idêntica à função de síntese da interpolação por vizinho-mais-próximo. Elas diferem uma da outra apenas nos valores de transição, onde temos que  $\beta^0$  é simétrica em relação à origem e satisfaz a propriedade de divisão da unidade. Portanto, contrário ao caso da interpolação por vizinho mais próximo, acontece em alguns casos excepcionais (estimativas em coordenada de meios inteiros) que a interpolação por  $\beta^0$  requer o cálculo entre duas amostras. No resto, esta função é indistinguível da função de vizinho mais próximo.

### **Grau $n = 1$**

A função B-spline  $\beta^1$  é exatamente idêntica ao caso linear.

### **Graus $n > 1$**

Nenhuma spline  $\beta^n$  de grau  $n > 1$  se beneficia da propriedade de ser interpoladora. Portanto, grande cuidado deve ser tomado para nunca usá-las no contexto da Equação 2.2.1, devendo-se usar a Equação 2.2.4 no lugar.

A B-spline cúbica é a mais utilizada na prática. Sua expressão é dada por:

$$\beta^3(x) = \begin{cases} \frac{2}{3} - \frac{1}{2}|x|^2(2 - |x|) & 0 \leq |x| < 1 \\ \frac{1}{6}(2 - |x|)^3 & 1 \leq |x| < 2 \\ 0 & 2 \leq |x| \end{cases} \quad (2.2.24)$$

Esta função de síntese não é interpoladora. No entanto, como foi explicado na equação 2.2.9, é possível que ela possua uma função de síntese com região de suporte infinito

$\varphi_{int} = \beta_{card}^3$  que permite construir exatamente a mesma função interpolada  $f$ . Para dar um exemplo concreto desse fato, mostramos na Figura 2.3 a B-spline cúbica  $\beta^3$  não interpoladora e sua função de síntese interpoladora equivalente, chamada de spline cúbica cardinal  $\beta_{card}^3$ . Graficamente a B-spline parece similar a uma Gaussiana. Isto não é por acaso, visto que uma B-spline converge para uma Gaussiana de variância infinita quando o grau da B-spline aumenta. Para um simples grau pequeno como  $n = 3$ , a aproximação é impressionantemente próxima, sendo que o erro relativo entre uma B-spline cúbica e uma Gaussiana com variâncias idênticas é de apenas por volta de 3,5%. No lado direito da Figura 2.3, a spline cardinal mostra oscilações decrescentes, que é o remanescente de uma função sinc. Isto também não é por acaso, já que uma spline cardinal converge para uma função sinc quando o grau da B-spline aumenta [13, 14]. Mostramos na Figura 2.4 o núcleo de aproximação para B-splines de vários graus. Claramente, quanto maior o grau, mais próximo a uma sinc estará a spline cardinal equivalente e melhor será seu desempenho.

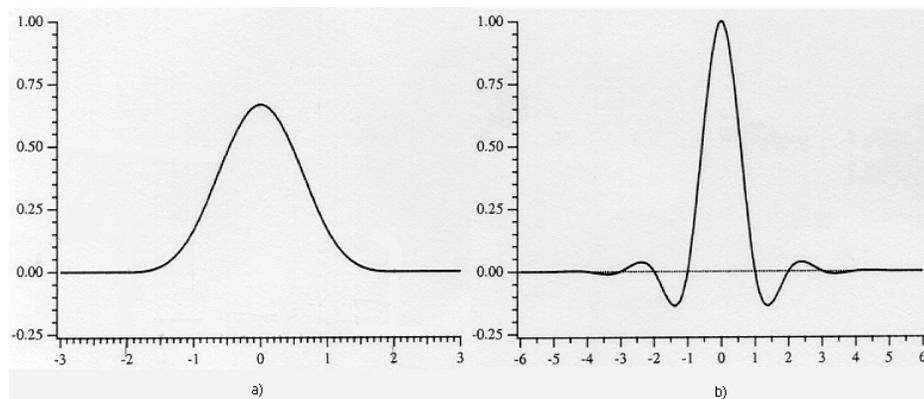


Figura 2.3: B-spline cúbica.

As funções B-spline desfrutam de uma ordem máxima de aproximação para uma região de suporte com um número inteiro dado. Reciprocamente, desfruta de uma região de

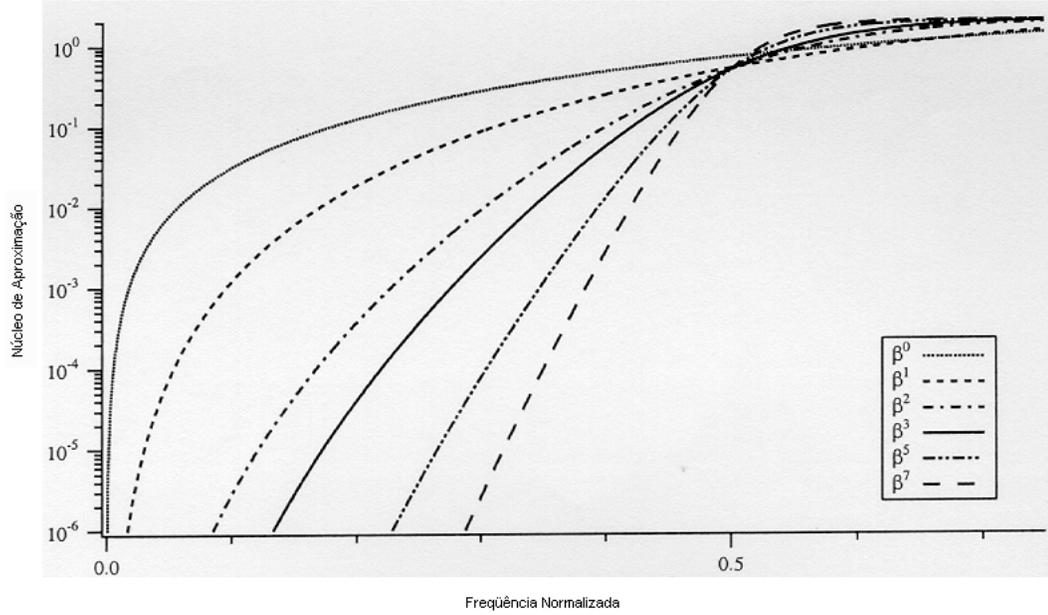


Figura 2.4: Núcleo de aproximação para B-splines de vários graus.

suporte mínimo para uma ordem de aproximação dada. Portanto elas pertencem a uma família de funções que possuem Máxima Ordem e Mínimo Suporte (Moms). Pode ser mostrado que qualquer uma destas funções pode ser expressa com uma soma ponderada de uma B-spline e suas derivadas [15].

$$\text{Moms}^n = \beta^n(x) + \sum_{m=1}^n c_m \frac{d^m}{dx^m} \beta^n(x) \quad (2.2.25)$$

B-splines são as funções Moms de diferenciação máxima. São apresentadas a seguir dois outros membros desta família. As funções o-Moms são tais que sua constante de aproximação quadrática mínima  $C_\varphi$  é mínima, enquanto as funções de Schaum são interpoladoras mas tem a constante de aproximação sub-ótima que é pior do que as o-Moms e as B-splines.

## o-Moms

As funções o-Moms são indexadas por seu grau polinomial  $n$ . São simétricas e suas dificuldades são idênticas às das B-splines das quais elas descendem. Além do mais, tem a mesma região de suporte que as  $\beta^n$ , que é  $W = n + 1$ , esta região de suporte é a menor possível para uma função de síntese com ordem de aproximação  $L = n + 1$ . Apesar de serem idênticas às B-splines de mesmo grau, sua constante de erro de aproximação  $C_\varphi$  é muito menor. De fato, as funções o-Moms são as que alcançam a menor constante quadrática média. Neste sentido, elas são assintoticamente aproximações ótimas, sendo as menores para uma dada região de suporte, com a maior ordem de aproximação, e com a menor constante de erro de aproximação [15].

Esta funções não são interpoladoras, portanto é necessário computar a seqüência de coeficientes  $\{c_k\}$  requeridos para a implementação da Equação 2.2.4.

A funções o-Mons de grau 0 e 1 são idênticas à  $\beta^0$  e  $\beta^1$  respectivamente. As funções o-Mons de graus maiores podem ser determinados recursivamente utilizando Fourier. Mostramos aqui a expressão para  $n = 3$ .

$$\text{oMoms}^3 = \beta^3(x) + \frac{1}{42} \frac{d^2}{dx^2} \beta^3(x) = \begin{cases} 1/2|x|^3 - |x|^2 + 1/14|x| + 13/21 & 0 \leq |x| < 1 \\ -1/6|x|^3 + |x|^2 - 85/42|x| + 29/21 & 1 \leq |x| < 2 \\ 0 & 2 \leq |x| \end{cases} \quad (2.2.26)$$

Só por curiosidade, podemos citar que estas funções de síntese possuem uma descontinuidade na origem, portanto sua regularidade é  $C^0$ . Também possuem outras descontinuidades para  $|x| = 1$  e  $|x| = 2$ .

## Funções de Schaum

Assim como as funções o-Moms, os núcleos pseudo-Lagrangianos propostos por Schaum em [10] também podem ser representados como uma soma de B-splines e suas derivadas de ordem par. Elas tem a mesma ordem e região de suporte que as B-splines e o-Moms. Sua vantagem é que elas são interpoladoras, mas em relação às B-splines e as o-Moms possuem a desvantagem de possuir a menor constante de aproximação  $C_\varphi$ . Por exemplo, para uma mesma ordem de aproximação  $L = 4$ , o valor mínimo é alcançado pela o-Moms com  $C_\varphi = 0,000627$ ; o valor da constante para a spline cúbica é mais que o dobro com  $C_\varphi = 0,00166$ , enquanto que a constante da Schaum cúbica tem valor  $C_\varphi = 0,01685$ . As funções de Schaum não possuem regularidade (são descontínuas) para graus pares e são  $C^0$  para graus ímpares.

## Funções de Key

A principal razão para a popularidade que a família de funções de Key desfruta [16] é em razão do fato de que elas possuem melhor desempenho do que a interpolação linear, sendo interpoladoras, dispensando o cálculo dos coeficientes de interpolação, podendo ser aplicadas utilizando-se a Equação 2.2.1. Estas funções são feitas de pedaços de polinômios cúbicos e dependem de um parâmetro  $a$ . A expressão geral é dada por:

$$u_a(x) = \begin{cases} (a+2)|x|^3 - (a+3)|x|^2 + 1 & 0 \leq |x| < 1 \\ a|x|^3 - 5a|x|^2 + 8a|x| - 4a & 1 \leq |x| < 2 \\ 0 & 2 \leq |x| \end{cases} \quad (2.2.27)$$

Comparando esta expressão com a da spline cúbica, percebemos que ambas demandam o cálculo de pedaços de polinômios de mesma região de suporte. No entanto, suas ordens de aproximação são diferentes: a melhor ordem de aproximação que as funções de Key podem alcançar é três, para o valor específico  $a = -1/2$ , enquanto que a spline cúbica

tem ordem quatro. Esta ordem a mais para  $\beta^3$  é conseguida ao custo do cálculo da seqüência de coeficientes  $\{c_k\}$  a serem usados na Equação 2.2.4. No entanto, utilizando uma abordagem de filtragem digital, esse custo pode se tornar desprezível. De modo geral, o ganho em velocidade oferecido pelas funções de Key não é suficiente para contrabalancear a perda de qualidade quando comparada com  $\beta^3$  e  $u_{-1/2}$ . Além do mais, a regularidade das funções de Key é  $C^1$ , que é uma a menos que a da spline cúbica.

### Função Sinc

A interpolação utilizando a função sinc, que corresponde à filtragem ideal, é a interpolação perfeita do ponto de vista teórico, para funções limitadas em banda. Porém na prática ela é irrealizável, visto que possui região de suporte infinita, sendo utilizada somente em casos muito específicos (ex.: translação de sinais periódicos por operações utilizando Transformada Discreta de Fourier). Em geral, é implementada uma aproximação da função sinc, introduzindo assim, uma certa quantidade de sobreposição espectral e/ou borramento (*blur*), dependendo da qualidade da aproximação. Também não é possível ajustar o desempenho a uma aplicação específica.

Como foi citado, a interpolação por sinc propicia uma interpolação livre de erros (teoricamente) para sinais limitados em banda. Mas esta classe de sinais representa apenas uma fração de todos os possíveis sinais. E também, afim de se reduzir a quantidade de memória necessária para armazenar um sinal, nem sempre uma função limitada em banda é amostrada com taxa de amostragem acima da frequência de Nyquist.

Em alguns problemas, uma região de suporte infinito não representa grandes dificuldades, desde que haja um algoritmo eficiente para implementar a interpolação com outra função de síntese de suporte finito, como é feito com as B-splines e o-Moms. Infelizmente nenhuma função pode ser ao mesmo tempo de região de suporte finito e limitada em

banda, o que elimina qualquer esperança de se encontrar uma função de síntese  $\varphi$  com região de suporte finito equivalente à função sinc. Portanto a solução clássica para se utilizar a função sinc é truncá-la, multiplicando-a por uma janela de região de suporte finito. Esse processo é chamado de janelamento ou truncamento. Um grande catálogo de janelas de truncamento pode ser encontrado em [10], juntamente com uma análise detalhada.

Por construção, todas as funções truncadas são interpoladoras. Enquanto a regularidade da função sinc não truncada é infinita, em geral a regularidade de sua versão truncada depende da janela de truncamento. Em particular, a regularidade é maximizada fazendo coincidir a região de suporte da janela com um par de pontos onde a função sinc é zero, ou seja cruza o eixo das abcissas. Isto resulta em efeito de blocos reduzido. Em teoria, qualquer janela de truncamento é admissível, incluindo, por exemplo, uma janela  $w_u$  tal que  $w_u(x)\text{sinc}(x) = u_{-1/2}(x)$ , onde  $u_{-1/2}(x)$  é a função de Key. Na prática, as únicas janelas que são consideradas, tem em geral, uma aparência Gaussiana e são frequentemente construídas de polinômios trigonométricos. Vamos citar dois a seguir.

### **Truncamento de Dirichlet (Janela Retangular)**

O truncamento de Dirichlet é o mais simples, sendo uma janela retangular de largura  $W$  correspondente a uma versão espichada de  $\beta^0$ , que não requer nenhum esforço computacional além de um teste para indicar se o ponto está incluído na região de suporte. A função de síntese truncada é dada por:

$$\text{sinc}_W^D(x) = \frac{\sin(\pi x)}{\pi x} \beta^0(x/W) \quad (2.2.28)$$

onde  $W$  é um inteiro par. O preço a ser pago pela simplicidade é a baixa qualidade. A regularidade desta função é baixa, já que não é derivável. Mais importante, sua ordem de aproximação é tão ruim quanto  $L = 0$ , e esta função nem mesmo satisfaz a divisão da

unidade. Isto significa que um acréscimo na taxa de amostragem não implica em redução no erro de interpolação.

### Truncamento de Hanning

O truncamento é definido como uma multiplicação da função sinc com alguma janela. Isto corresponde a um convolução no domínio da transformada de Fourier. A janela de Hanning é uma entre várias tentativas de se projetar uma janela que tenha propriedades favoráveis no domínio da Fourier. Sua expressão é dada por:

$$\text{sinc}_W^H(x) = \text{sinc}_W^D(x)(1/2 + 1/2 \cos(2\pi x/W)) \quad (2.2.29)$$

cuja função de síntese possui ordem de aproximação  $L = 0$ , que não é melhor do que o truncamento de Dirichlet. Mas a constante de  $C_\varphi$  é melhorada significativamente, sendo que para  $\text{sinc}_4^D$  possui o valor  $C_\varphi = 0,1076$  e para  $\text{sinc}_4^H$ ,  $C_\varphi = 0,0153$ . Como ela é continuamente derivável, Hanning é mais regular do que Dirichlet.

## 2.3 Interpolação no Domínio da Transformada

Com o surgimento do algoritmo para a Transformada Rápida de Fourier (*Fast Fourier Transform* - FFT), que possui complexidade computacional da ordem de  $O(N \log N)$  ( $N$  é o número de amostras de um sinal), muitas aplicações se tornaram implementáveis em computador, principalmente as aplicações sobre imagens digitais, que possuem um número elevado de amostras. Outras transformadas que podem ser implementadas através de uma relação com a Transformada Discreta de Fourier (*Discrete Fourier Transform* - DFT) como a Transformada Discreta do Coseno (*Discrete Cosine Transform* - DCT) ou a Transformada Discreta do Seno (*Discrete Sine Transform* - DST) também desfrutaram desse ganho em velocidade computacional e vêem sua aplicação favorecida.

Como vimos anteriormente um sinal discreto pode ter sua taxa de amostragem aumentada, ou seja, ser interpolado, através de procedimentos discretos. O princípio básico da interpolação utilizando transformadas é a compressão do espectro de frequências, resultante de um aumento na taxa de amostragem, seguido de uma filtragem passa-baixas ideal. Esse método pode ser implementado discretamente de forma bastante simplificada, acrescentando zeros de forma apropriada aos coeficientes da transformada discreta do sinal e aplicando a transformada discreta inversa. Apresentaremos o método de forma específica para cada transformada.

### 2.3.1 DFT

A Transformada Discreta de Fourier é uma representação, no domínio da frequência, de uma seqüência finita  $x[n]$ ,  $n = 0, 1, \dots, N - 1$ , de comprimento  $N$  resultando também em uma seqüência finita de coeficientes que são as amostras da Transformada de Fourier contínua, nas frequências discretas normalizadas  $\omega = 2\pi k/N$  para  $0 \leq \omega < 2\pi$ . A DFT é definida matematicamente como [1, 2, 17]:

$$X[k] = \sum_{n=0}^{N-1} x[n] \exp\left(\frac{-j2\pi k n}{N}\right), \quad \text{para } k = 0, 1, \dots, N - 1 \quad (2.3.1)$$

e a transformada inversa é dada por:

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] \exp\left(\frac{j2\pi k n}{N}\right), \quad \text{para } n = 0, 1, \dots, N - 1 \quad (2.3.2)$$

onde  $j$  é o número imaginário  $\sqrt{-1}$ .

O espectro da transformada de Fourier contínua de um sinal discreto é periódico, com período  $\omega = 2\pi$ , e seus valores no intervalo  $\omega = [0, \pi[$  são complexos conjugados dos valores no intervalo  $\omega = [0, -\pi]$ , assim como suas réplicas. Devido a esta periodicidade o espectro da DFT é limitado em  $\omega = [0, \pi]$ . Potranto a interpolação do sinal é dada

adicionando os valores zero no meio dos coeficientes da DFT, de acordo com a expressão:

$$X_i[k] = \begin{cases} X[k], & 0 \leq k < \lfloor N/2 \rfloor \\ 0, & \lfloor N/2 \rfloor \leq k < LN - \lfloor N/2 \rfloor \\ X[k - N], & LN - \lfloor N/2 \rfloor \leq k < LN \end{cases} \quad \text{para } 0 \leq k < 2N \quad (2.3.3)$$

A seqüência interpolada  $x_i[n]$  de tamanho  $LN$ , é conseguido aplicando a DFT inversa sobre os coeficientes resultantes  $X_i[n]$ .

### 2.3.2 DCT

A Transformada Discreta do Coseno é a transformada mais utilizada no processo de codificação de imagens sendo adotada nos padrões JPEG (*Joint Picture Experts Group*) e MPEG (*Motion Picture Experts Group*). Esta transformada apresenta um conjunto de características superiores a um grande número de transformadas. A DCT é a transformada que mais se aproxima em muitos aspectos à Transformada de Karhunen-Loève [18], considerada a transformada ótima do ponto de vista estatístico. Entre esses aspectos, podemos citar a eficiência na compactação de energia do sinal, a correlação residual e a taxa de distorção em relação à quantidade de bits utilizados na codificação. A DCT de uma seqüência finita  $x[n]$ ,  $n = 0, 1, \dots, N - 1$ , de comprimento  $N$  é definida pelo par de transformadas:

$$X^{c2}[k] = 2 \sum_{n=0}^{N-1} x[n] \cos \left( \frac{(2n+1)\pi k}{2N} \right), \quad 0 \leq k \leq N-1 \quad (2.3.4)$$

$$x[n] = \frac{1}{N-1} \sum_{k=0}^{N-1} B[k] X^{c2}[k] \cos \left( \frac{(2n+1)\pi k}{2N} \right), \quad 0 \leq n \leq N-1 \quad (2.3.5)$$

onde a transformada inversa envolve a função de peso  $B[k]$  dada por:

$$B[k] = \begin{cases} \frac{1}{2}, & k = 0 \\ 1, & 1 \leq k \leq N-1 \end{cases} \quad (2.3.6)$$

Também é possível expressar a DCT de uma seqüência finita  $x[n]$  em termos da DFT. A implementação da DFT assume que um sinal finito a ser transformado é periódico. A DCT também assume o sinal como sendo periódico, mas também que possui simetria par. Para entendermos a relação entre a DCT e a DFT construímos uma seqüência com simetria par  $x_p[n]$  de tamanho  $2N$  a partir de uma seqüência  $x[n]$  através da equação:

$$x_p[n] = x[n] + x[2N - 1 - n] = \begin{cases} x[n], & 0 \leq n \leq N - 1 \\ x[2N - 1 - n], & N \leq n \leq 2N - 1 \end{cases} \quad (2.3.7)$$

A Figura 2.5 mostra um exemplo da construção da seqüência simétrica. A DFT de tamanho  $2N$  da seqüência  $x_p[n]$  também de tamanho  $2N$  é:

$$X_p[k] = \sum_{n=0}^{2N-1} x_p[n] W_{2N}^{kn}, \quad \text{para } 0 \leq k \leq 2N - 1 \quad (2.3.8)$$

onde

$$W_{2N}^{kn} = \exp\left(\frac{-j2\pi kn}{2N}\right) \quad (2.3.9)$$

Então obtem-se:

$$X_p[k] = \sum_{n=0}^{N-1} x[n] W_{2N}^{kn} + \sum_{n=N}^{2N-1} x[2N - 1 - n] W_{2N}^{kn}, \quad \text{para } 0 \leq k \leq 2N - 1 \quad (2.3.10)$$

e

$$X_p[k] = W_{2N}^{-k/2} \sum_{n=0}^{N-1} 2x[n] \cos\left(\frac{(2n+1)\pi k}{2N}\right), \quad \text{para } 0 \leq k \leq 2N - 1 \quad (2.3.11)$$

A DCT de  $N$  pontos da seqüência  $x[n]$ , é obtida a partir de  $X_p[k]$  por:

$$X^{c2}[k] = \begin{cases} W_{2N}^{k/2} X_p[k], & \text{para } 0 \leq k \leq N - 1 \\ 0, & \text{para os demais valores} \end{cases} \quad (2.3.12)$$

A transformada inversa utilizando DFT é conseguida obtendo-se os coeficientes da DFT de  $2N$  pontos a partir dos coeficientes da DCT pela relação:

$$X_p[k] = \begin{cases} X^{c2}[0], & k = 0 \\ W_{2N}^{-k} X^{c2}[k], & 1 \leq k \leq N - 1 \\ 0, & k = N \\ -W_{2N}^{-k} X^{c2}[2N - k,] & N + 1 \leq k \leq 2N - 1 \end{cases} \quad (2.3.13)$$

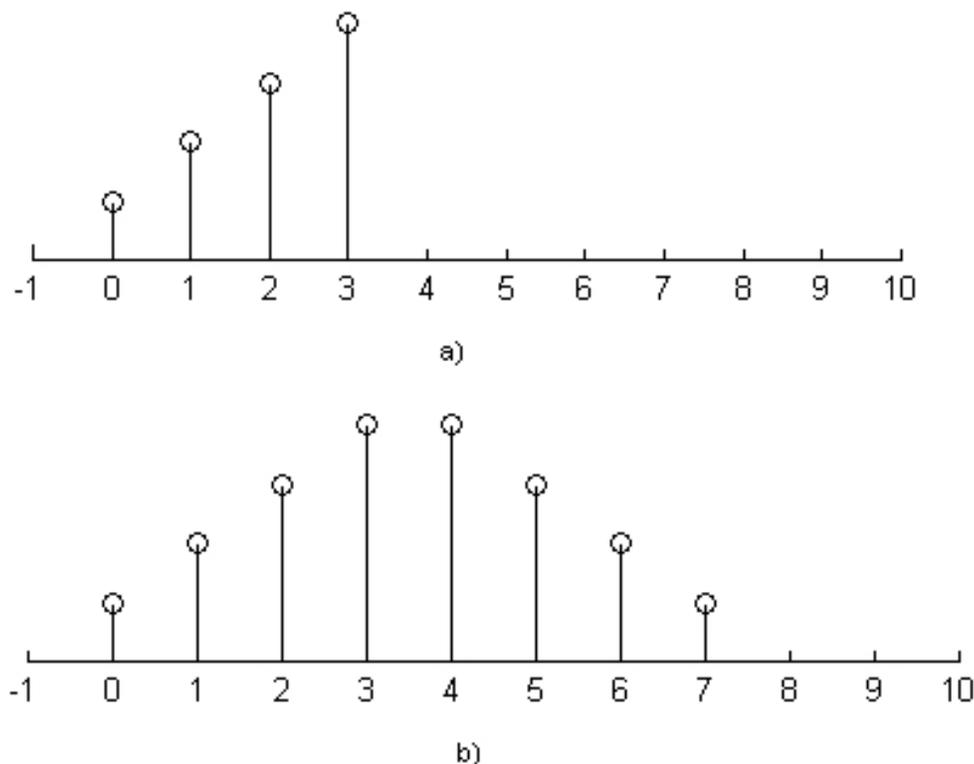


Figura 2.5: Construção de uma seqüência de simetria par de tamanho  $2N$ . a) Seqüência original. b) Seqüência com simetria par construída a partir da seqüência original.

Utilizando-se a DFT inversa calcula-se a seqüência  $x_p[n]$  de onde extrai-se a seqüência  $x[n] = x_p[n]$  para  $n = 0 \dots N - 1$ .

O procedimento de calcular a DCT utilizando a DFT permite a utilização do algoritmo rápido para a transformada de Fourier (FFT). A DCT também possui um algoritmo rápido [18]. Porém esta relação nos dá uma noção da vantagem que a DCT tem sobre a DFT no processamento de sinais por blocos. Quando o sinal é muito extenso, limitações físicas impõe que o sinal seja dividido em blocos de tamanho reduzido, diminuindo o tempo de processamento do sinal. Esse procedimento é muito comum em imagens digitais, onde a imagem é dividida em blocos usualmente quadrados de tamanho 8 ou 16. Como foi citado,

a DFT é implementada sobre o princípio de que o sinal finito é considerado periódico. Ao se quebrar esse sinal em blocos, a extensão periódica do bloco  $\tilde{x}[n]$  poderá, muito provavelmente, apresentar em suas bordas, uma descontinuidade (o sinal não é suave) como mostra a Figura 2.6, o que corresponde a um acréscimo de frequências altas. Esta descontinuidade é suavizada significativamente quando o sinal possui simetria par, como acontece na DCT.

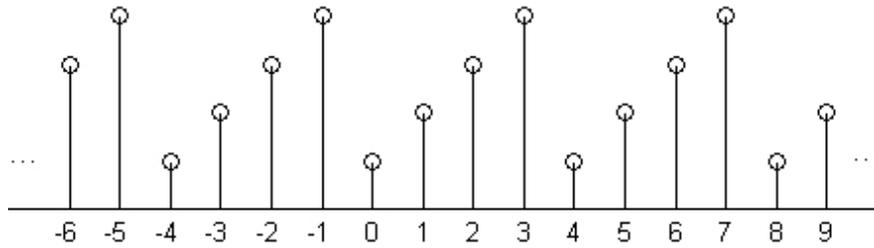


Figura 2.6: Extensão periódica de um bloco.

O processo de interpolação de sinais utilizando a DCT é bastante simples. Assim como no processo de interpolação por DFT, bastam ser adicionados zeros aos coeficientes da DCT. Como os coeficientes da DCT não apresentam simetria em torno da origem, os zeros são acrescentados no final da seqüência, da seguinte forma:

$$X_i^{c2}[k] = \begin{cases} X^{c2}[k], & 0 \leq k \leq N - 1 \\ 0, & N \leq k \leq LN - 1 \end{cases} \quad (2.3.14)$$

onde  $L$  é a taxa interpolação com valor inteiro. A seqüência interpolada de tamanho  $LN$  é conseguida aplicando a transformada inversa sobre a seqüência de coeficientes  $X_i^{c2}$  e multiplicando por  $\sqrt{L}$ , o ganho do filtro passa-baixas.

### 2.3.3 DSTr

Visando reduzir o erro médio quadrático na interpolação por DFT, Fraser [19] propôs o janelamento do sinal e a eliminação das amostras dos extremos da seqüência. Araújo [20] propôs com o mesmo o objetivo, a rotação do eixo da seqüência antes e após a aplicação da transformada. Essa rotação, de pequena complexidade, produziu melhorias na interpolação com filtragem pela DFT. Em outro trabalho, Araújo [21] apresentou um novo método para interpolação usando rotação de eixos, porém agora utilizando a DST, derivando assim a DSTr, que se mostrou bastante eficiente. Pelaes [22] apresentou uma metodologia para aplicação de rotação de eixos combinada com as transformadas DCT e DST aplicadas no processo de interpolação de imagens, mostrando resultados superiores para a DSTr.

A DST (tipo I) assim como a DCT é composta por bases ortogonais e é definida por [18]:

$$X^{s1}[k] = \sqrt{\frac{2}{N}} \sum_{n=1}^{N-1} x[n] \text{sen} \left( \frac{\pi kn}{N} \right), \quad \text{para } k = 1, \dots, N-1. \quad (2.3.15)$$

Note que a transformada é calculada somente sobre os pontos  $n = 1, \dots, N-1$  para uma seqüência de  $N$  pontos com  $n = 0, \dots, N-1$ . A mesma equação para a transformada direta pode ser utilizada para a transformada inversa, ou seja,  $x[n] = \text{DST} \{ \text{DST} \{ x[n] \} \}$  aplicando a DST duas vezes obtém-se a mesma seqüência.

A complexidade computacional da DST e da DCT são correspondentes, assim como algumas de suas propriedades como compactação de energia e taxa de distorção. Porém, o comportamento da DST-I em relação à correlação residual é inverso ao da DCT. Correlação residual mede a qualidade com que uma transformada é capaz de decorrelacionar uma seqüência. Hamidi e Pearl [23] propuseram uma medida chamada correlação fracional que é deixada por uma transformada. Aplicando esta medida a um processo de

Markov-1, com diferentes coeficientes de correlação, a várias transformadas, percebe-se que a DCT-II apresenta menor correlação residual do que todas as outras transformadas, exceto a DST-I. Porém a correlação residual da DCT-II decai inversamente proporcional ao coeficiente de correlação, enquanto que a correlação residual da DST aumenta. O ponto de compensação entre as duas transformadas depende do tamanho  $N$  da seqüência, sendo próximo a 0,7 para uma seqüência de tamanho 16, conforme mostra o gráfico da Figura 2.7. Isto mostra que para algumas aplicações a DST pode ser superior a todas as outras transformadas. Porém sua dificuldade de aplicação reside no fato dela não ser definida para todos os pontos da seqüência a ser transformada.

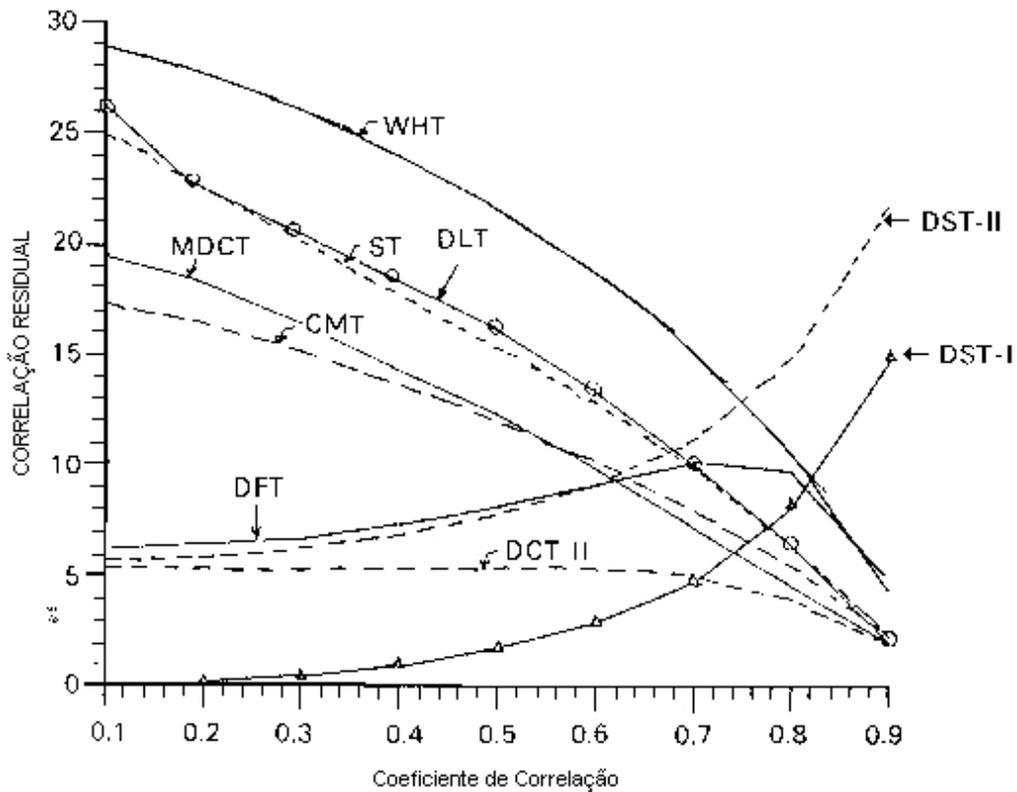


Figura 2.7: Correlação Residual versus Coeficiente de Correlação para  $N = 16$ .

Da mesma forma que a DCT, a DST também possui uma relação íntima com a DFT e pode ser calculada utilizando a FFT. Na DST o sinal finito a ser transformado é considerado possuir simetria ímpar, portanto para calcular a DST  $X^{s1}[k]$ , de uma seqüência  $x[n]$ , utilizando a DFT, uma seqüência de simetria ímpar  $x_{im}[n]$  é construída a partir da seqüência original  $x[n]$  através da relação:

$$x_{im}[n] = \begin{cases} 0, & n = 0 \\ -x[N - n], & 1 \leq n \leq N \\ 0, & n = N + 1 \\ x[n - N - 2], & N + 2 \leq n \leq 2N + 1 \end{cases} \quad (2.3.16)$$

A Figura 2.8 mostra um exemplo de como a seqüência é construída

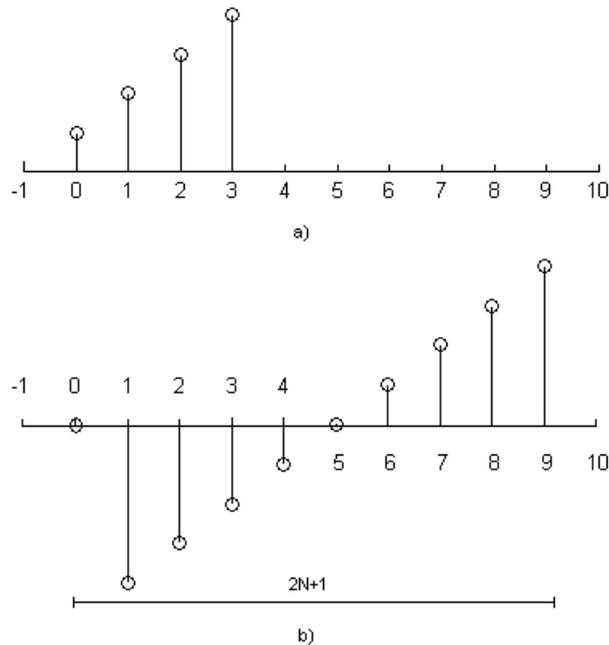


Figura 2.8: Sequencia de simetria ímpar. a)Seqüência original. b)Seqüência com simetria ímpar construída a partir da seqüência original.

Aplicando a DFT sobre a seqüência  $x_{im}[n]$  obtem-se a seqüência  $X_{im}[k]$ :

$$X_{im}[k] = \sum_{n=0}^{2N+1} x_{im}[n] W_{2N+2}^{kn}, \quad \text{para } 0 \leq k \leq 2N+1 \quad (2.3.17)$$

$$X_{im}[k] = \sum_{n=1}^N -x_{im}[N-n] W_{2N+2}^{kn} + \sum_{n=N+2}^{2N+1} x_{im}[n-N-2] W_{2N+2}^{kn}, \quad \text{para } 0 \leq k \leq 2N+1 \quad (2.3.18)$$

A DST  $X^{s1}[k]$  de  $x[n]$  é obtida tomando-se  $N$  pontos de  $X_{im}[k]$  com a seguinte relação:

$$X^{s1}[k] = \begin{cases} X_{im}[k], & \text{para } 1 \leq k \leq N \\ 0, & \text{para os demais valores} \end{cases} \quad (2.3.19)$$

Com esse procedimento mostramos que a DST também pode ser implementada utilizando-se o algoritmo FFT. A DST também possui um algoritmo rápido [18] similar ao da DCT. A apresentação desta relação também serve para mostrar o comportamento da transformada no caso do processamento por blocos. A simetria ímpar, imposta pela DST, não tem o mesmo efeito positivo na redução de descontinuidades da extensão periódica. Em alguns casos, quando a seqüência é suave, a simetria ímpar pode acrescentar descontinuidades, que nem sequer a DFT acrescenta. Portanto a DST é péssima quando utilizada no processamento por blocos.

O procedimento de rotação de eixos, proposto por Araújo [20], geralmente aplicado aos blocos de um sinal, é definido por:

$$x_r[n] = x[n] - r[n] \quad (2.3.20)$$

onde a seqüência  $r[n]$  definida por:

$$r[n] = \frac{x[N-1] - x[0]}{N-1} n + x[0] \quad (2.3.21)$$

é uma linha reta que passa pelos pontos  $x[0]$  (o primeiro ponto do bloco) e pelo ponto  $x[N]$  (o último ponto do bloco) como é mostrado na Figura 2.9. Para desfazer a rotação,

o primeiro e último ponto do bloco devem ser armazenados. Para reduzir o número de pontos a serem armazenados, o último ponto de um bloco pode ser o primeiro ponto do próximo bloco adjacente, produzindo uma pequena sobreposição de uma amostra entre os blocos. O processo de rotação inverso é definido pela expressão:

$$x[n] = x_r[n] + r[n] \quad (2.3.22)$$

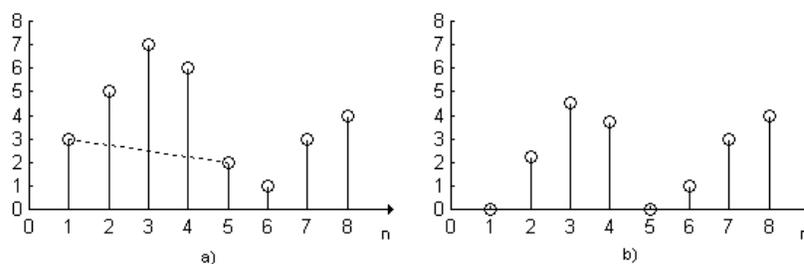


Figura 2.9: Rotação de eixos de um bloco de tamanho  $N = 4$ . a) Seqüência original e reta que passa pelos pontos  $x[0]$  e  $x[N + 1]$ . b) Bloco rotacionado.

O procedimento de rotação de eixos possui dupla vantagem: reduz as descontinuidades da extensão periódica quando o sinal é dividido em blocos; e permite a utilização da DST. Um bloco com seus pontos rotacionados tem as descontinuidades de suas extensões periódicas diminuídas em todos os caso mostrados até agora (periodicidade simples, com simetria par e com simetria ímpar). Portanto, com a rotação de eixos, a DST não tem seu desempenho degradado, independente do sinal a ser transformado. Outro detalhe importante é que o bloco rotacionado terá sempre o seu primeiro ponto igual a zero, proporcionando a aplicação da DST, já que esse ponto pode ser desprezado.

A aplicação da DST sobre um sinal é realizada com o seguinte procedimento: o sinal é dividido em blocos de tamanho  $N$ . Toma-se cada bloco e a primeira amostra do bloco seguinte, resultando em uma seqüência de tamanho  $N + 1$ . Aplica-se a rotação de

eixos sobre esta seqüência. A primeira e a última amostra resultante serão iguais a zero ( $x_{bl}[0] = x_{bl}[N+1] = 0$ ). A última posição, que pertence ao bloco seguinte, pode então ser descartada. Aplica-se a DST sobre a seqüência de tamanho  $N - 1$  para  $n = 1, \dots, N - 1$  e guarda-se a primeira amostra original do bloco  $x_{bl}[0]$  na primeira posição da seqüência de coeficientes da DST  $X^{sr}[0]$ .

Para se obter o sinal original, a DST é aplicada novamente sobre os coeficiente do bloco para  $n = 1, \dots, N - 1$  e a rotação inversa é aplicada sobre a seqüência resultante, utilizando os pontos  $X^{sr}[0]$  do bloco atual e do bloco seguinte.

O processo de interpolação de sinais utilizando a DSTr é exatamente igual ao da DCT, acrescentando-se os zeros no final da seqüência, de acordo com a equação:

$$X_i^{sr}[k] = \begin{cases} X^{sr}[k], & 0 \leq k \leq N - 1 \\ 0, & N \leq k \leq LN - 1 \end{cases} \quad (2.3.23)$$

lembrando-se que o procedimento de rotação de eixos foi realizado antes de se aplicar a DST. A seqüência interpolada é conseguida aplicando a transformada inversa sobre a seqüência de coeficientes e multiplicando o resultado por  $\sqrt{L}$  ( $L$  é o taxa de interpolação). Em seguida, a rotação de eixo é revertida, utilizando as extremidades armazenadas. A interpolação reproduzirá exatamente os pontos originais se uma quantidade exata de zeros  $Z_s = NL$ ,  $L$  inteiro, for adicionada. Caso contrário o formato da função será mantido, mas com uma certa escala, influenciando muito na relação Sinal-Ruído (SNR).

É possível notar-se que todos os algoritmos de interpolação por transformadas comprimem o espectro limitado em banda na freqüência normalizada  $\omega = \pi$ , o limite da freqüência de Nyquist. Porém, alguns sinais são filtrados antes da amostragem para garantir que não ocorra sobreposição espectral. Portanto, o conteúdo espectral de alta freqüência perdido, é considerado como sendo zero no processo de interpolação. Esse conteúdo espectral poderia ser aproximado por um valor diferente de zero, alcançando

maiores detalhes da função reconstruída. Para isso é possível utilizar a informação contida nas baixas frequências para tentar encontrar a informação das altas frequências utilizando-se um algoritmo de extrapolação de sinais.

No próximo capítulo serão apresentados um série de algoritmos de extrapolação de sinais, indicando suas características específicas e a classe de sinais que eles conseguem melhores aproximações. No geral os algoritmos de extrapolação trabalham sobre sinais limitados em banda, que é o caso da seqüência de coeficientes das transformadas aqui citadas.



## Capítulo 3

# Extrapolação

Como foi comentado no capítulo anterior, os procedimentos de amostragem de imagens são limitados e a informação armazenada muitas vezes não é suficiente para uma certa aplicação. Quando se busca conseguir informação espectral além da frequência disponível, diz-se que se deseja uma "super-resolução" do sinal que pode ser considerada como uma estimativa dos coeficientes da transformada. Uma variedade de métodos existe para se conseguir estimativas satisfatórias e geralmente tais métodos recaem sobre uma das seguintes categorias:

- a) Periodograma [1]: O segmento finito conhecido é multiplicado (modulado) por uma janela e a transformada de Fourier do produto é calculada.
- b) Filtragem Preditiva: Os coeficientes de um filtro digital linear de ordem fixa são determinados que levarão a uma predição ótima em um passo de uma versão amostrada do sinal de segmento finito. O filtro resultante é então usado para estimar o espectro do sinal. Esse procedimento inclui os métodos autoregressivos e máxima entropia equivalente.
- c) Extrapolação: Nesta abordagem, o segmento finito é estendido usando-se uma regra de extrapolação apropriada. Essa será a abordagem avaliada e utilizada neste

trabalho.

Extrapolação significa estender um sinal além do intervalo conhecido. Os algoritmos de extrapolação precisam levar em conta os seguintes casos:

- a) O segmento limitado  $z[n]$  de amostras conhecidas de um sinal discreto  $y[n]$  em um intervalo estão livres de ruído, ou seja:

$$z[n] = y[n], \quad -M \leq n \leq M \quad (3.0.1)$$

O problema consiste em estimar os valores de  $y[n]$  fora do intervalo  $[-M, M]$ ;

- b) As observações  $z[n]$  contêm ruído aditivo branco  $\eta[n]$  com média zero e variância  $\sigma_n^2$ , ou seja:

$$z[n] = y[n] + \eta[n], \quad -M \leq n \leq M \quad (3.0.2)$$

Encontrar uma estimativa para  $y[n]$  para  $n \in [-M, M]$  e extrapolar fora de  $[-M, M]$ ;

- c) As observações  $z[n]$  contêm, além do ruído branco, como acima, um ruído de interferência  $c[n]$  (colorido), ou seja:

$$z[n] = y[n] + c[n] + \eta[n], \quad -M \leq n \leq M \quad (3.0.3)$$

Encontrar uma estimativa para  $y[n]$  para  $n \in [-M, M]$  e extrapolar fora de  $[-M, M]$ ;

- d) Dadas as observações  $z[n]$  como em um dos casos acima, encontrar uma estimativa da densidade espectral de potência do sinal  $y[n]$ ;

Um sinal contínuo, determinístico, limitado em banda  $y(t)$ , dado em um intervalo finito  $[-T, T]$ , pode ser extrapolado sem erros fora do intervalo conhecido. Isso é possível porque o sinal é analítico e a expansão em série de Taylor:

$$y_a(T + \Delta) = y_a(T) + \Delta y'_a(T) + \Delta^2/2 y''_a(T) + \dots \quad (3.0.4)$$

pode ser utilizada para se determinar  $y_a(t)$  fora de  $[-T, T]$  já que todas as derivadas no intervalo  $[-T, T]$  podem ser calculadas (pelo menos em princípio). Na prática, o procedimento acima não é plausível porque os dados observados poderiam conter ruído (mesmo pequeno) e o cálculo das derivadas é um processo muito sensível a ruídos.

Para sinais discretos, a propriedade analítica desaparece devido à amostragem. Portanto uma estimativa extrapolada pode não coincidir com o sinal contínuo original. Na prática, outras restrições além da suposição de que o sinal é limitado em banda devem ser imposta sobre a estimativa para alcançar uma solução única. De fato, infinitas funções podem gerar o mesmo conjunto de amostras, gerando um número infinito de soluções para o problema de extrapolação. Portanto, a única saída é escolher uma das soluções que apresentará o menor erro em algum critério. A solução que apresentará o melhor resultado será aquela que utilize o maior número possível de informações sobre o sinal. Infelizmente, essas informações nem sempre estão disponíveis ou sequer podem ser inferidas a partir do segmento conhecido.

### 3.1 Projeções Ortogonais

Para se chegar a uma solução de extrapolação, deve-se restringir ao conjunto de sinais definidos no eixo do tempo de extensão infinita que possuem energia finita. O sinal  $y(t)$  é dito conter energia finita se a expressão integral:

$$\int_{-\infty}^{\infty} y(t)y^*(t)dt \quad (3.1.1)$$

é finita, onde  $y^*(t)$  denota complexo conjugado de  $y(t)$  e a integral é calculada no sentido de Lebesgue. Esse conjunto de sinais em conjunção com o produto interno padrão, definido por:

$$\langle y_1(t), y_2(t) \rangle = \int_{-\infty}^{\infty} y_1(t)y_2^*(t)dt \quad (3.1.2)$$

de fato constitui um espaço de Hilbert que tem sido designado classicamente de  $L_2$ .

Se um sinal  $y(t)$  está contido no espaço de Hilbert  $L_2$ , a transformada de Fourier associada existe e o sinal  $y(t)$  pode ser recuperado a partir de sua transformada de Fourier por meio da transformada inversa definida por:

$$y(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} Y(\omega) e^{j\omega t} d\omega \quad (3.1.3)$$

Grande parte da teoria de sinais depende da decomposição de um dado sinal nas componentes cujas transformadas de Fourier ocupam diferentes porções do espectro de frequência. Para demonstrar como essa decomposição é realizada, considera-se o conjunto  $\Omega$  de frequência limitada em banda, que possui um medida diferente de zero mas finita. Esse conjunto identifica tipicamente a componente espectral de frequência de  $y(t)$ , que é de interesse em uma dada aplicação. Com isso em mente, decompõe-se a integral da transformada inversa de Fourier da seguinte forma:

$$y(t) = \frac{1}{2\pi} \int_{\omega \in \Omega} Y(\omega) e^{j\omega t} d\omega + \frac{1}{2\pi} \int_{\omega \notin \Omega} Y(\omega) e^{j\omega t} d\omega = y_1(t) + y_2(t) \quad (3.1.4)$$

onde a primeira integral foi designada como sinal  $y_1(t)$  e a segunda como  $y_2(t)$ .

Utilizando essa decomposição de sinal, é evidente que a transformada de Fourier dos sinais componentes  $y_1(t)$  e  $y_2(t)$  são identicamente iguais a zero para todo  $\omega \in \Omega$  e  $\omega \notin \Omega$ , respectivamente. Isso implica que os sinais componentes são "ortogonais" no sentido de que  $\langle y_1(t), y_2(t) \rangle = 0$ . Para estabelecer essa propriedade, apresenta-se a função de transferência do filtro ideal limitado em banda, que corresponde ao conjunto  $\Omega$  definido por:

$$H(\omega) = \begin{cases} 1 & \text{para } \omega \in \Omega \\ 0 & \text{para } \omega \notin \Omega \end{cases} \quad (3.1.5)$$

Utilizando-se a transformada de Fourier da relação 3.1.4 e essa definição de  $H(\omega)$ , observa-se que:

$$Y(\omega) = Y_1(\omega) + Y_2(\omega) = H(\omega)Y(\omega) + [1 - H(\omega)]Y(\omega) \quad (3.1.6)$$

Obviamente, o produto das transformada  $Y_1(\omega)Y_2^*(\omega)$  é identicamente zero, devido à estrutura de  $H(\omega)$ . Esse fato em conjunção com a identidade de Parseval, que estabelece que:

$$\int_{-\infty}^{\infty} Y_1(\omega)Y_2^*(\omega)d\omega = 2\pi \int_{-\infty}^{\infty} y_1(t)y_2^*(t) \quad (3.1.7)$$

ou em representação equivalente de produto interno:

$$\langle Y_1, Y_2 \rangle = 2\pi \langle y_1, y_2 \rangle \quad (3.1.8)$$

estabelece a ortogonalidade dos sinais componentes  $y_1$  e  $y_2$ . Essa propriedade de ortogonalidade é de importância vital em muitos estudos de análise de sinais.

Ficou estabelecido assim que todo sinal contido no espaço de Hilbert  $L_2$  pode ser unicamente decomposto em uma soma de dois sinais, um cuja transformada de Fourier é idêntica a zero para  $\omega$  fora de um dado conjunto  $\Omega$  e outro cuja transformada de Fourier é idêntica a zero para  $\omega$  dentro do conjunto  $\Omega$ . Além do mais, os sinais componentes  $y_1$  e  $y_2$  resultantes desse processo são sempre ortogonais. Com isso em mente, é evidente que o espaço de Hilbert essencial  $L_2$  foi decomposto na soma direta específica:

$$L_2 = B_\Omega \oplus B_\Omega^\perp \quad (3.1.9)$$

onde  $B_\Omega^\perp$  denota o subespaço de  $L_2$  cujos elementos são ortogonais ao subespaço limitado em banda:

$$B_\Omega = \{y \in L_2 : Y(\omega) \equiv 0\} \quad \text{para } \omega \notin \Omega \quad (3.1.10)$$

Relembrando o problema de extrapolação: dada uma versão truncada no tempo do sinal  $y(t) \in B_\Omega$  especificado por  $y(t)$  para  $t \in \Lambda$  determinar o comportamento de  $y(t)$  fora do intervalo de observação  $\Lambda$ .

Nesta formulação, os conjuntos característicos  $\Lambda$  e  $\Omega$  são assumidos como tendo um medida finita mas diferente de zero. A solução para essa tarefa pode ser encontrada apresentando dois operadores de projeção.

Com a finalidade de dar ao problema de extrapolação uma estrutura algébrica, será benéfico considerar operadores de projeção ortogonal. O operador linear  $P$  é dito ser um operador de projeção no espaço  $X$  se ele for idempotente, ou seja,  $P^2x = Px, \forall x \in X$ . Além do mais, esse operador de projeção é considerado ortogonal, se sua imagem e seu espaço nulo, forem ortogonais (isto é,  $\langle x, y \rangle = 0 \forall x \in N(P)$  e  $y \in R(P)$ ). Com essas definições básicas, pode-se apresentar dois operadores de projeção que apresentam um papel central na investigação de extrapolação.

Operador de truncamento no tempo:  $T$

O operador fundamental de truncamento no tempo  $T$  é motivado pelo conjunto de observações no tempo  $\Lambda$ . Em particular, se esse operador for aplicado sobre um sinal de entrada  $y(t)$  produz um sinal de saída  $z(t)$  denotado por:

$$z = Ty \tag{3.1.11}$$

onde o valor de  $z$  no tempo  $t$  é especificado por:

$$z(t) = \begin{cases} y(t) & \text{para } t \in \Lambda \\ 0 & \text{para } t \notin \Lambda \end{cases} \tag{3.1.12}$$

Esse operador de truncamento no tempo é linear e sua imagem e espaço nulo são formalmente dados por:

$$R(T) = y \in L_2 : y(t) \equiv 0 \text{ para } t \notin \Lambda \tag{3.1.13}$$

e

$$N(T) = y \in L_2 : y(t) \equiv 0 \text{ para } t \in \Lambda \tag{3.1.14}$$

Examinando-se a relação 3.1.12, verifica-se que  $T^2y = Ty \quad \forall y \in L_2$ , que estabelece que  $T$  é um operador de projeção. Além do mais, visto que o produto interno entre elementos dos subespaços  $R(T)$  e  $N(T)$  é zero, esses subespaços são ortogonais. Tendo estabelecido

que o operador de truncamento no tempo  $T$  é um operador de projeção ortogonal e conseqüentemente, o espaço de sinais  $L_2$  pode ser decomposto pela soma direta:

$$L_2 = R(T) \oplus R(T)^\perp \quad (3.1.15)$$

onde  $R(T)^\perp = N(T)$  denota o subespaço ortogonal a  $R(T)$ . Essa decomposição em soma direta implica que qualquer sinal  $y \in L_2$  pode ser unicamente expresso como  $y = y_1 + y_2$  em que os componentes individuais  $y_1 \in R(T)$  e  $y_2 \in R(T)^\perp$ . Além do mais, esses componentes são gerados por operações de projeção ortogonal  $y_1 = Ty$  e  $y_2 = (I - T)y$ , respectivamente, onde  $I$  é o operador de identidade.

Operador de Banda Limitada: B

O segundo operador de projeção ortogonal a ser considerado é motivado pela decomposição da transformada de Fourier de um sinal genérico em  $L_2$ , conforme descrito pela relação 3.1.4. Em particular, os componentes individuais do sinal  $Y(\omega)$ , especificados por  $H(\Omega)Y(\omega)$  e  $[1 - H(\Omega)]Y(\omega)$ , estão contidos nos subespaços fechados  $B_\Omega$  e  $B_\Omega^\perp$ , respectivamente. No domínio do tempo, a componente  $y_1 \in B_\Omega$  é dado por:

$$y_1(t) = \int_{-\infty}^{\infty} h(t - \tau)y(\tau)d\tau \quad (3.1.16)$$

onde a resposta impulsiva  $h(t)$  é a transformada inversa da função de transferência do filtro ideal limitador de banda 3.1.5 especificado formalmente por:

$$h(t) = \frac{1}{2\pi} \int_{\omega \in \Omega} e^{j\omega t} d\omega \quad (3.1.17)$$

A relação 3.1.16 será expressa em notação de operador para prover uma estrutura algébrica ao problema de extrapolação. Com isso em mente, a correspondência natural é dada por:

$$y_1 = By \quad (3.1.18)$$

onde  $B$  é a operação de filtragem limitada em banda ideal especificada pela relação 3.1.16. Para se estabelecer que  $B$  é um operador de projeção, utiliza-se o fato de que  $H^2(\omega) = H(\omega) \forall \omega$  para concluir que  $B^2y = By \forall y \in L_2$ . Além do mais, a imagem e o espaço nulo desse operador de projeção, dados por:

$$R(B) = \{y \in L_2 : Y(\omega) \equiv 0 \text{ para } \omega \notin \Omega\} \quad (3.1.19)$$

$$N(B) = \{y \in L_2 : Y(\omega) \equiv 0 \text{ para } \omega \in \Omega\} \quad (3.1.20)$$

são ortogonais, como pode ser mostrado usando-se a identidade de Parseval. Portanto, o operador limitador de banda ideal  $B$  é um operador de projeção ortogonal cujo espaço de imagem é idêntico ao subespaço de banda limitada  $B_\Omega$ . Esse fato não é por coincidência, já que o operador  $B$  foi propositalmente definido para alcançar este objetivo. É interessante notar que esse operador limitador de banda ideal provê outra decomposição de soma direta do espaço do sinal dada por:

$$L_2 = R(B) \oplus R(B)^\perp \quad (3.1.21)$$

onde tem-se que  $R(B)^\perp = N(B)$ . Portanto, qualquer sinal  $y \in L_2$  pode ser unicamente decomposto como uma soma do sinal em  $R(B)$  e  $R(B)^\perp$ . Essa decomposição é conseguida pela operação básica  $y = By + [I - B]y$  na qual  $By \in R(B)$  e  $[I - B]y \in R(B)^\perp$ .

Considerando o problema de extrapolação onde o segmento conhecido é livre de ruído, considera-se que  $\mathbf{z}$  é um vetor de tamanho  $(2M + 1) \times 1$  de observações do vetor  $\mathbf{y}$  de tamanho infinito representando o sinal original, ou seja,  $\mathbf{z} = T\mathbf{y}$ . Se  $\mathbf{y}$  é um sinal limitado em banda, deve satisfazer a relação:

$$\mathbf{z} = T\mathbf{B}\mathbf{y} = \mathbf{A}\mathbf{y} \quad (3.1.22)$$

onde

$$\mathbf{A} \equiv T\mathbf{B} \quad (3.1.23)$$

O problema de extrapolação agora é encontrar uma estimativa do vetor infinito  $\mathbf{y}$ , dado o vetor finito de observações  $\mathbf{z}$ . Apesar do problema de extrapolação possuir uma solução única no caso contínuo, o mesmo não acontece no caso discreto, porque a matriz  $\mathbf{A}$  é retangular e possui posto no máximo  $(2M+1)$ . Além do mais, é um sistema indeterminado de equações simultâneas de modo que a solução de mínimos quadrados, apesar de existir, não é única. Portanto, o problema de extrapolação consiste em se escolher uma das infinitas soluções.

## 3.2 Energia Mínima

### 3.2.1 Redução da Energia do Erro

Papoulis e Gerchberg [24] propuseram um algoritmo iterativo onde a energia fora do segmento conhecido é zerada a cada iteração, reduzindo a energia do erro, já que o sinal é considerado ser limitado em banda e, portanto, somente energia do erro haverá fora da banda. Esse método é definido matematicamente por:

$$y_n = Bz_{n-1}, \quad z_0 = z = Ty \quad (3.2.1)$$

$$z_n = z_0 + [I - T]y_n \quad (3.2.2)$$

e tende ao sinal original na média quadrática quando  $n \rightarrow \infty$ , se o sinal for livre de ruído e a extensão do sinal for corretamente estimada. Se a extensão do sinal for superestimada, então haverá menor redução do erro. Se a extensão do erro for subestimada, o algoritmo falha.

Papoulis propôs uma implementação do algoritmo utilizando-se a FFT com os seguintes passos: aplicar a transformada de Fourier no sinal; limitar a banda de frequência do sinal, truncando os coeficientes; aplicar a transformada inversa sobre o espectro truncado; substituir o segmento do sinal pela parte conhecida; repetir os passos reduzindo iterativamente

o erro médio quadrático. O método é demonstrado na forma de blocos na Figura 3.1. A Figura 3.2 mostra a primeira iteração do método.

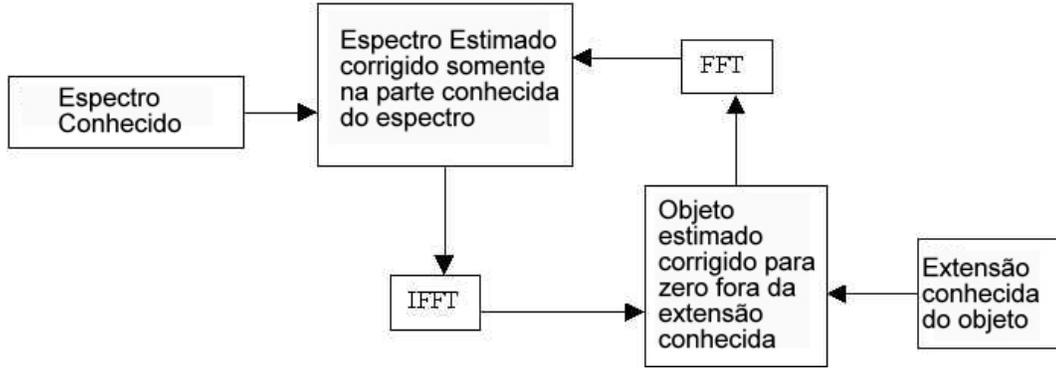


Figura 3.1: Esquema de blocos para o algoritmo de extrapolação por Redução da Energia do Erro.

Esse algoritmo pode ser identificado como sendo um caso específico de um algoritmo de gradiente associado com um problema de minimização de mínimos quadrados [25]. Esse algoritmo também é chamado de método de projeções ortogonais alternadas porque as iterações são projetadas alternadamente no espaço de funções limitadas em banda e truncadas no tempo.

### 3.2.2 Norma Mínima e Mínimos Quadrados

Para o caso discreto unidimensional, a seqüência conhecida a ser extrapolada pode ser definida como  $z[n] = y[n]; n \in \Delta$  denotando um vetor  $\mathbf{z}$  de tamanho  $(2M + 1) \times 1$  de observações do sinal limitado em banda  $y(t) \in B_\Omega$  no intervalo de observações  $\Delta = [-M, M]$ ; em geral,  $\Delta$  pode ser qualquer conjunto finito de índices em  $\mathbb{Z}$ .

Os operadores de projeção ortogonal são redefinidos nesse cenário. O operador de truncamento no espaço  $\mathbf{T} = \{t(n, m); n \in \Delta, m \in \mathbb{Z}\}$  é uma matriz de tamanho  $(2M +$

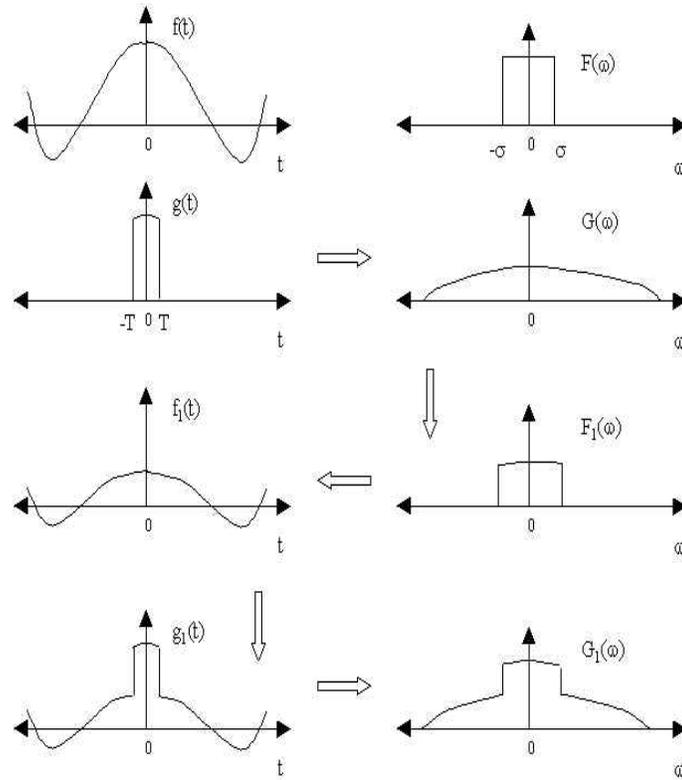


Figura 3.2: Primeira iteração do algoritmo.

1)  $\times \infty$  onde:

$$t(n, m) = \begin{cases} 1, & n = m \\ 0, & \text{caso contrário} \end{cases} \quad (3.2.3)$$

O operador  $\mathbf{T}$  trunca um vetor de tamanho  $\infty \times 1$  para um vetor de tamanho  $(2M + 1) \times 1$  simetricamente na origem. A transposta  $\mathbf{T}^T$  de tamanho  $\infty \times (2M + 1)$  estende um vetor de tamanho  $(2M + 1) \times 1$  para um vetor de tamanho  $\infty \times 1$  acrescentando zeros. Para qualquer operador matricial  $\mathbf{A}$  de tamanho  $\infty \times \infty$ ,  $\underline{\mathbf{A}}$  denota a matriz duplamente truncada  $\mathbf{TAT}^T$  de tamanho  $(2M + 1) \times (2M + 1)$ .

Similar a  $\mathbf{T}$  é definido o operador  $\mathbf{L}$  de tamanho  $\infty \times \infty$  onde  $\mathbf{L} = \{l(n - m; \sigma); n, m \in$

$\mathbb{Z}$ } e onde:

$$l(n - m; \sigma) = \frac{\sin \sigma(n - m)}{\pi(n - m)}, \quad 0 < \sigma < \pi \quad (3.2.4)$$

O operador  $\mathbf{L}$  é o operador matricial passa baixas ideal com frequência de corte  $\sigma$  rad. Ele pode ser interpretado como uma versão amostrada do operador Limitador de Frequência  $B$ , para o caso específico passa baixas.

Como já foi mostrado, a formulação popular do problema de extrapolação discreta consiste em estimar, a partir de  $\mathbf{z}$ , uma função  $\mathbf{y}$  que satisfaça a relação  $\mathbf{z} = \mathbf{T}\mathbf{y}$  tal que  $\mathbf{y} = \mathbf{L}\mathbf{y}$ , ou seja  $\mathbf{y}$  é limitada em banda. Esse problema se resume a solucionar  $\mathbf{z} = \mathbf{T}\mathbf{L}\mathbf{y}$ ,  $\mathbf{y} \in B_\Omega$ . Apesar dessa solução não especificar unicamente um  $\mathbf{y} \in B_\Omega$ , porque  $\mathbf{T}\mathbf{L}$  não é inversível, a inversa generalizada de Moore-Penrose, ou pseudoinversa, de  $\mathbf{T}\mathbf{L}$  sempre existe e leva à extrapolação de norma mínima, mínimos quadrados (*Minimum Norm Least Square* - MNLS) que é única. A extrapolação MNLS proposta por Jain [25] é chamada assim, porque também satisfaz o problema de norma mínima, mínimos quadrados:

$$\min \mathbf{y} \|\mathbf{z} - \mathbf{T}\mathbf{L}\mathbf{y}\|_\Delta^2 \text{ tal que } \|\mathbf{y}\|_{\mathbb{Z}}^2 \text{ é mínimo} \quad (3.2.5)$$

Portanto a estimativa do sinal  $\mathbf{y}$  é dada por:

$$\mathbf{y}^\dagger = \mathbf{E}^\dagger \mathbf{z} \quad (3.2.6)$$

onde

$$\mathbf{E}^\dagger = \mathbf{L}\mathbf{T}^T \underline{\mathbf{L}}^{-1} \quad (3.2.7)$$

é a matriz de tamanho  $\infty \times (2M + 1)$  pseudoinversa de  $\mathbf{T}\mathbf{L}$ .  $\mathbf{E}^\dagger$  sempre existirá porque  $\underline{\mathbf{L}}$  é positiva definida para qualquer  $M > 0$  [25]. Portanto a estimativa de  $y$  sempre existirá para qualquer conjunto finito de amostras. Esse algoritmo é equivalente a resolver o

sistema de equações:

$$\sum_{m \in \Delta} l[n - m; \sigma] x[m] = z[n], \quad n \in \Delta \quad (3.2.8)$$

para a seqüência  $x[n]$  de tamanho  $(2M + 1)$  e então utilizando:

$$y[n] = \sum_{m \in \Delta} l[n - m; \sigma] x[m] \quad (3.2.9)$$

para computar  $y[n]$  para todo  $n$ .

### 3.2.3 Mínimos Quadrados Ponderados

Como uma alternativa ao algoritmo MNLS, podemos considerar a restrição de Mínimos Quadrados Ponderados (*Weighted Least Squares - WLS*), proposto por Diethorn e Munson [26] que se baseia no cálculo dos coeficientes  $a(n)$  do espectro como uma estimativa do espectro  $Z(\omega)$  da seqüência, através do cálculo de:

$$Y(\omega) = P(\omega) \sum_{n \in \Delta} a(n) e^{-j\omega n} \quad (3.2.10)$$

que satisfaça o problema de Mínimos Quadrados Ponderados:

$$\min_{a(n)} \int_{-\pi}^{\pi} |Z(\omega) - Y(\omega)|^2 P(\omega)^{-1} d\omega \quad (3.2.11)$$

onde  $P(\omega)$  é uma função de peso especificada pelo usuário, não negativa em  $(-\pi, \pi)$ . A minimização de 3.2.11 mostra que o  $a(n)$  ótimo satisfaz:

$$z[n] = \sum_{m \in \Delta} a[m] p[n - m], \quad n \in \Delta \quad (3.2.12)$$

onde  $p[n]$  é a seqüência correspondente a  $P(\omega)$ . A transformada inversa de Fourier de 3.2.10 leva a:

$$y^p[n] = \sum_{m \in \Delta} a[m] p[n - m], \quad n \in \mathbb{Z} \quad (3.2.13)$$

a estimativa ótima de  $z[n]$ . Combinando-se 3.2.12 com 3.2.13 obtêm-se a forma mais compacta do algoritmo de extrapolação WLS:

$$\mathbf{y}^p = \mathbf{E}^p \mathbf{z} \quad (3.2.14)$$

onde

$$\mathbf{E}^p = \mathbf{P} \mathbf{T}^T \mathbf{P}^{-1} \quad (3.2.15)$$

e  $\mathbf{P} \triangleq p[n - m]$ .

A extrapolação por MNLS é portanto um caso especial da extrapolação por WLS, onde  $p[n] = l[n; \sigma]$ . Algumas normas heurísticas de procedimentos para a escolha da seqüência de peso  $p[n]$  podem ser encontradas em [27]. Nota-se que mais informação sobre o sinal pode ser utilizada, se ela estiver disponível, no algoritmo WLS do que no algoritmo MNLS, no caso, a distribuição da energia do sinal no domínio da freqüência.

### 3.2.4 Modificação Iterativa da Norma Mínima Ponderada

Os algoritmos que utilizam uma função de peso no domínio da freqüência necessitam do conhecimento da distribuição da energia e a banda a qual o sinal é limitado. Como em muitos casos essa informação não está disponível, Cabrera e Parks [28] propuseram um algoritmo onde uma função de peso é calculada iterativamente a partir das amostras dadas e de um conjunto de estimativas das iterações anteriores. A função de peso é estimada utilizando-se uma estimativa espectral não paramétrica  $\hat{P}_y(f)$ , correspondente a uma função de autocorrelação de duração finita  $\hat{r}_{xx}(n)$ .

O espectro é conseguido através de um periodograma modificado de modo que apenas uma forma ampla do espectro permanece das iterações anteriores e os detalhes são recuperados através da geração de extrapolação válida utilizando-se a restrição dos dados. O periodograma modificado envolve uma operação de janelamento no domínio do tempo. O

filtro na iteração  $\lambda$  é definido como:

$$h_\lambda[n] = p[n]\hat{y}_{\lambda-1}[n] \quad (3.2.16)$$

A cada iteração, a estimativa  $\hat{y}_\lambda[n]$  casa com o conjunto de observações dadas e portanto é uma extrapolação válida. A janela  $p[n]$  é escolhida de forma que inclua também algumas estimativas próximas ao conjunto de observações. O tamanho da janela determina a suavidade do espectro do periodograma modificado e também define o tamanho máximo da extrapolação resultante em todas as iterações.

A iteração pode ser iniciada criando o primeiro filtro  $H_1[k]$  igual a uma constante positiva para todo  $k = 0, 1, 2, \dots, N - 1$ . Nesse caso a primeira estimativa é um sinal constituído pelas observações dadas nas posições de  $\Delta$  e zero nas demais posições. Em seguida obtém-se uma estimativa de espectro de periodograma modificado, que utiliza apenas as observações dadas. Essa é a primeira função de peso não plana, que é então utilizada na segunda iteração. Percebe-se que os itens necessários para a utilização do algoritmo são: o tamanho  $N$  da DFT, a janela  $p[n]$  e o vetor de amostras  $\mathbf{z}$ . A computação realizada em cada iteração pode ser descrita no domínio do tempo através dos seguintes passos:

a) Computar a autocorrelação circular:

$$a_\lambda[n] = h_\lambda^*[-n] \otimes h_\lambda[n] \quad (3.2.17)$$

b) Construir a matriz  $\mathbf{G}$  com elementos:

$$\mathbf{G}_{r,c} = a_\lambda(\Delta(r) - \Delta(c)) \quad (3.2.18)$$

c) Computar o vetor de constantes:

$$\mathbf{x}_\lambda = \mathbf{G}^{-1}\mathbf{z} \quad (3.2.19)$$

d) Formar a estimativa utilizando as constantes e a autocorrelação:

$$\hat{y}_\lambda[n] = \sum_{i=1}^L x_\lambda[i] a_\lambda[n - \Delta(i)] \quad (3.2.20)$$

e) Definir o próximo filtro:

$$h_{\lambda+1}[n] = p[n] \hat{y}_\lambda[n] \quad (3.2.21)$$

Diferentes funções de pesos são obtidas a cada iteração devido ao acréscimo das novas amostras estimadas. O processo é repetido até que nenhuma mudança entre iterações ocorram, ou então um número de vezes especificado pelo usuário. Economias nos cálculos podem ser alcançados nas iterações intermediárias utilizando FFT para obter  $a_\lambda[n]$ . Além do mais, só é necessário avaliar  $\hat{y}_\lambda[n]$  para os valores onde a janela é diferente de zero. A matriz  $\mathbf{G}$  é Toeplitz e Hermiciana no caso especial onde as observações estão consecutivamente espaçadas e um algoritmo rápido pode ser utilizado para a sua inversão.

### 3.3 Dimensão Reduzida

Os algoritmos de Energia Mínima se caracterizam por ser uma reconstrução do sinal limitado em banda a partir de um segmento com a energia minimizada em algum senso, satisfazendo as restrições de largura de banda e amostras temporais.

Os métodos de extrapolação baseados em banda limitada têm se mostrado eficiente somente para uma extensão limitada do sinal além do segmento. Baseado em limites de erros propostos em [29], que dão o erro percentual máximo possível para a extrapolação de um ponto, concluiu-se que nenhum erro ocorre no intervalo correspondente ao segmento de observações (o que é evidente, visto que os algoritmos são projetados com esse objetivo). Porém, quanto mais afastado do segmento o ponto extrapolado estiver, maior será o erro. Após um intervalo de tamanho  $T_L$  o erro máximo é próximo a 100%. O valor de  $T_L$  é

dado por  $T_M/2\beta$  onde  $T_M$  é o tamanho do segmento de observações e  $\beta$  é a largura de banda normalizada do sinal.

Esse resultado corresponde à dimensão essencial dessa classe de sinais. Para sinais de largura de banda  $\beta$ , existem aproximadamente  $2\beta T_L$  sinais independentes no intervalo  $T_L$ . Além do mais, com  $2M + 1$  amostras e por volta de um sinal independente por amostra, o maior intervalo expandido por esses sinais é encontrado por  $2\beta T_L \approx 2M + 1$ .

O método de estimativa limitada em banda resulta em uma reconstrução do sinal desconhecido que é limitado em efetividade aproximadamente ao intervalo  $T_L$ , ou seja, apresenta uma aproximação aceitável somente nesse intervalo. No entanto, visto que o método limitado em banda resulta em uma reconstrução com energia mínima, o intervalo no qual uma boa extrapolação é esperada, é incluído na restrição de energia mínima.

Nos métodos de extrapolação com dimensões reduzidas, os sinais de interesse, além de serem considerados limitados em banda, também são considerados como sendo essencialmente limitados em tempo, ou seja, quase toda energia do sinal está concentrada em regiões conhecidas.

### 3.3.1 Estimação Limitada em Banda e Concentrada em Tempo

De forma a melhorar potencialmente a reconstrução no intervalo  $T_L$ , onde uma boa extrapolação é esperada, o método de estimação Limitada em Banda e Concentrada em Tempo (*Band-Limited Time Concentrated* - BLTC), proposto por Kolba e Parks [30], não inclui essa região na restrição de energia mínima, portanto a região definida em  $\Gamma$  a ser minimizada é simetricamente localizada ao redor da janela de observações  $\Delta$ . O tamanho de  $\Gamma$  é determinado igualando-se o número de observações ao produto  $N\beta$  tempo - largura de banda. Portanto, se o número de observações for  $p$  e o sinal for limitado em banda a uma banda passante de largura fracional  $\beta$ , o intervalo de concentração  $\Gamma$  terá  $N = p/\beta$

índices. Por exemplo, se  $\Delta = \{1, 2, 3\}$  e  $\beta = 1/5$ , o  $\Gamma$  usado será  $\{-5, -4, \dots, 9\}$ .

Assim como o operador  $\mathbf{T}$  seleciona as amostras de  $\mathbf{z}$  a partir dos índices contidos no conjunto  $\Delta$ , defini-se o operador  $\mathbf{W}$  que seleciona as amostras a partir dos índices do conjunto  $\Gamma$ . Portanto, se  $\Gamma = \{n_1, n_2, \dots, n_N\}$ , então:

$$\mathbf{W}\mathbf{y}(i) = \mathbf{y}(n_i), \quad i = 1, 2, \dots, N \quad (3.3.1)$$

e o operador adjunto  $\mathbf{W}^*$  de  $\mathbf{W}$  é definido por

$$\mathbf{W}^*\mathbf{z}(k) = \begin{cases} \mathbf{z}(i), & \text{se } k = n_i \\ 0, & \text{se } k \notin \Gamma \end{cases} \quad (3.3.2)$$

É evidente que quando  $\Gamma$  for contíguo, ou seja, completo no intervalo,  $\mathbf{W}^*\mathbf{W}$  é um operador de janelamento simples.

Da mesma forma, analogamente a  $\mathbf{L}$ , o operador passa-baixas ideal, define-se o operador  $\mathbf{B}$  que seleciona as frequências contidas no conjunto  $\Omega \in [-pi, pi]$ . Esse operador difere do operador  $\mathbf{L}$ , por não ser exclusivamente passa-baixas, podendo o conjunto  $\Omega$  ser, inclusive, descontínuo.

Retirando-se a restrição de índices contíguos do conjunto  $\Delta$  o operador  $\mathbf{T}$  pode ser redefinido de modo a selecionar as  $p$  observações indexadas pelo conjunto  $\Delta = \{t_1, \dots, t_p\}$  de forma que:

$$\mathbf{T}\mathbf{y}(i) = \mathbf{y}(t_i), \quad i = 1, 2, \dots, p \quad (3.3.3)$$

e o operador adjunto  $\mathbf{T}^*$  de  $\mathbf{T}$  é definido por

$$\mathbf{T}^*\mathbf{z}(k) = \begin{cases} \mathbf{z}(i), & \text{se } k = t_i \\ 0, & \text{se } k \notin \Delta \end{cases} \quad (3.3.4)$$

Portanto, dado  $\mathbf{z} - \mathbf{T}\mathbf{y}$ , a abordagem BLTC consiste em

$$\min \|\mathbf{y} - \mathbf{z}\|^2 \quad (3.3.5)$$

levando em conta que  $\mathbf{B}\mathbf{y} = \mathbf{y}$  e  $\mathbf{T}\mathbf{B}\mathbf{y} = \mathbf{z}$ .

Baseando-se na definição das Seqüências Esferoidais Estendidas Discretas (*Discrete Prolate Spheroidal Sequences* - DPSS) (Apêndice A) pode-se definir a matriz  $\Phi_T \triangleq [\mathbf{T}\phi_1, \dots, \mathbf{T}\phi_N]$  de tamanho  $p \times N$  e  $\mathbf{S} \triangleq \text{diag}[\lambda_1, \lambda_2, \dots, \lambda_N]$  a extrapolação BLTC pode ser construída através de um algoritmo de dois passos:

Passo 1) Resolver o sistema de equações lineares de tamanho  $p \times p$ :

$$\Phi_T(\mathbf{I} - \mathbf{S})^{-1}\mathbf{S}\Phi_T^*\mathbf{x} = \mathbf{z} \quad (3.3.6)$$

Passo 2) Obter as estimativas através de:

$$\mathbf{a} = (\mathbf{I} - \mathbf{S})^{-1}\mathbf{S}\Phi_T^*\mathbf{x} \quad (3.3.7)$$

e

$$\mathbf{y}_{BLTC} = \sum_{i=1}^N \mathbf{a}_i \phi_i \quad (3.3.8)$$

Para valores grandes de  $N$ , a escolha de  $\Gamma = N\beta$ , o produto tempo-largura de banda, é uma boa estimativa da dimensão essencial o que garante que o número de observações seja próximo ao número de graus de liberdade para sinais essencialmente limitados em tempo a  $\Gamma$ .

### 3.4 Concentração de Energia

Em muitas aplicações, é comum que o sinal, além de ser limitado em banda, também possua sua energia concentrada em uma certa região. O conhecimento dessa região de concentração de energia pode ser utilizado para se conseguir uma extrapolação melhor. Os algoritmos de dimensão reduzida desfrutam implicitamente dessa propriedade quando o sinal apresenta região de concentração em  $\Gamma$ , mesmo que essa informação não esteja disponível de antemão. Porém eles não tiram proveito explícito dessa informação quando ela está disponível.

A extrapolação com energia concentrada (*Energy Concentration - EC*), proposta por Potter e Arun [31], incorpora conhecimento antecipado da distribuição da energia do sinal tanto no domínio do tempo quanto no domínio da frequência, minimizando uma função diferente da energia total do sinal.

Deve-se lembrar que a energia do sinal no domínio do tempo é dado por:

$$\sum_{n \in \mathbb{Z}} |y|^2 \quad (3.4.1)$$

De modo a refletir informação adicional sobre a energia do sinal, são adotadas medidas ponderadas. No domínio do tempo a medida de energia pode ser modificada resultando em:

$$\sum_{n \in \mathbb{Z}} q_n |y|^2 \quad (3.4.2)$$

utilizando pesos  $q_n$  reais não negativos. Na prática, o conhecimento da distribuição de energia do sinal identifica as regiões do sinal com alta ou baixa energia. Os pesos  $q_n$  são escolhidos de modo a satisfazer  $q_n = 1$  para todo  $n$  exceto em um conjunto finito  $\Gamma$  de tamanho  $N$  de índices onde  $q_n$  pode ser qualquer número real não negativo. Esse conjunto é referido como sendo a região de concentração de energia.

Para refletir uma região de concentração de energia em  $\Gamma$ , pesos são escolhidos como sendo menores que a unidade:  $0 \leq q_n < 1$ ; inversamente a escolha de pesos  $q_n > 1$  em índices em  $\Gamma$  reflete o desejo de suprimir a energia em uma extrapolação nesses índices. Colocando os  $N$  pesos que diferem de 1 em uma matriz de tamanho  $N \times N$   $\mathbf{Q} \triangleq \text{diag}(1 - q_{n_1}, \dots, 1 - q_{n_N})$  onde  $\Gamma = \{n_1, \dots, n_N\}$ , vê-se que a medida de energia de 3.4.2 pode ser expressa por:

$$\sum_{n \in \Gamma} q_n |y[n]|^2 + \sum_{n \notin \Gamma} q_n |y[n]|^2 = \langle \mathbf{y}, (\mathbf{I} - \mathbf{W}^* \mathbf{Q} \mathbf{W}) \mathbf{y} \rangle_H. \quad (3.4.3)$$

A fim de refletir as informações conhecidas sobre a distribuição de energia do sinal no domínio da frequência, uma função de peso espectral  $P(e^{j\omega})$ , limitada, não negativa

e com valores reais no conjunto  $\Omega$  é escolhida. Por exemplo, a função de peso pode ser escolhida como sendo relativamente pequena na região  $[-\pi, \pi]$ , onde se sabe que o sinal observado possui alta energia e relativamente grande onde se sabe que a energia do sinal deve ser suprimida em uma extrapolação. O operador  $\mathbf{P}$  é definido como o corresponde a  $P(e^{j\omega})$ , levando a uma medida ponderada da energia no domínio da frequência dada por:

$$\langle \mathbf{y}, \mathbf{P}\mathbf{y} \rangle_H \triangleq \frac{1}{\pi} \int_{[-\pi, \pi]} |Y(e^{j\omega})|^2 P(e^{j\omega}) d\omega \quad (3.4.4)$$

As medidas de energia modificadas 3.4.2 e 3.4.5 são combinadas de modo a refletir as informações sobre a distribuição de energia do sinal tanto no domínio do tempo quanto no domínio da frequência. A medida de energia combinada fica definida por:

$$\|\mathbf{y}\|_{P,Q}^2 \triangleq \langle \mathbf{y}, [(\mathbf{I} - \mathbf{W}^* \mathbf{Q} \mathbf{W}) + \mathbf{P}]\mathbf{y} \rangle_H \quad (3.4.5)$$

Portanto a seleção de uma única extrapolação que não só se iguala às observações mas também reflete a informação disponível que descreve a distribuição da energia do sinal pode ser formulada como o seguinte problema de otimização: para um dado vetor de observações  $\mathbf{z}$  encontrar a seqüência limitada em banda  $\mathbf{y} \in B_\Omega$  que satisfaça  $\mathbf{T}\mathbf{y} = \mathbf{z}$  e que minimize  $\langle \mathbf{y}, [(\mathbf{I} - \mathbf{W}^* \mathbf{Q} \mathbf{W}) + \mathbf{P}]\mathbf{y} \rangle_H$ .

Uma solução não iterativa para o problema de extrapolação com energia concentrada e limitada em banda é derivado aplicando-se o teorema de projeção no espaço de Hilbert de sinais limitados em banda utilizando-se um produto interno modificado.

Seja  $\Omega \subset [-\pi, \pi]$  o conjunto que denota a banda passante,  $\Delta$  qualquer conjunto de  $p$  índices de observação,  $\Gamma$  qualquer conjunto ordenado de  $N$  índices de concentração de energia, e  $\mathbf{z}$  qualquer vetor com  $p$  valores de observações. Adicionalmente, seja  $\mathbf{Q}$  uma matriz diagonal de tamanho  $N \times N$  com  $Q_{i,i} \leq 1, Q_{i,i} \neq 0, i = 1, \dots, N$ , e seja  $P(e^{j\omega})$  real e não negativo em  $\Omega$ . Então existe um solução única,  $\hat{\mathbf{y}}$  dada por:

$$\hat{\mathbf{y}} = \mathbf{F}(\mathbf{W}^* \mathbf{u} + \mathbf{T}^* \mathbf{v}) \quad (3.4.6)$$

onde  $\mathbf{u}$  e  $\mathbf{v}$  são soluções únicas para o sistema não singular de  $p + N$  equações lineares:

$$\begin{bmatrix} \mathbf{TFT}^* & \mathbf{TFW}^* \\ \mathbf{QWFT}^* & \mathbf{QWFW}^* - \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \mathbf{u} \end{bmatrix} = \begin{bmatrix} \mathbf{z} \\ 0 \end{bmatrix} \quad (3.4.7)$$

e o operador  $\mathbf{F}$  é dado por  $(\mathbf{I} + \mathbf{P})^{-1}\mathbf{B}$ .

Sem perda de generalidade, a região de concentração pode ser considerada de modo que satisfaça  $\Gamma \cap \Delta = \emptyset$  como é evidente observando-se o sistema de equações em 3.4.7. Se  $\Gamma \cap \Delta \neq \emptyset$ , então 3.4.7 ainda seria não singular, mas pela expressão de definição de  $\mathbf{u}$ , os valores de  $\mathbf{v}$  que estivessem em índices de  $\Delta$  seriam obrigados a combinar com os pontos de dados de peso; portanto o sistema poderia ser reduzido em tamanho pelo número de índices em  $\Gamma \cap \Delta$ . Além do mais, o cálculo de 3.4.7 não necessita da inversão explícita de uma matriz de tamanho  $p + N \times p + N$ . Ao invés disto, o sistema de equações pode ser decomposto em dois sistemas menores: a matriz positiva definida  $N \times N$ ,  $(\mathbf{I} - \mathbf{QWFW}^*)$  e a matriz positiva definida  $p \times p$ ,  $\{\mathbf{TFT}^* + \mathbf{TFW}^*(\mathbf{I} - \mathbf{QWFW}^*)^{-1}\mathbf{QWFT}^*\}$ .

Essa solução pode ser vista como um algoritmo de dois passos para a obtenção única de uma extrapolação limitada em banda que é consistente com a seqüência dada e tem energia mínima medida pela norma ponderada  $\|\cdot\|_{P,Q}$ . No primeiro passo,  $\mathbf{u}$  e  $\mathbf{v}$  são obtidos resolvendo 3.4.7. As quatro matrizes que compõe o sistema, são finitas em tamanho e podem ser construídas a partir do conhecimento da banda passante  $\Omega$ , dos índices de observações  $\Delta$ , da região de concentração  $\Gamma$ , da matriz de pesos  $\mathbf{Q}$ , e da função de peso  $P(e^{j\omega})$ , dados. Com a região de observações  $\Delta = \{n_1, \dots, n_p\}$  e a região de concentração  $\Gamma = \{k_1, \dots, k_N\}$ , a entrada  $(i, j)$  na matriz  $\mathbf{TFW}^*$  é dada por:

$$\mathbf{f}(n_i - k_j) \quad n_i \in \Delta, \quad k_j \in \Gamma \quad (3.4.8)$$

onde  $\mathbf{f}$  é a transformada discreta inversa de Fourier de  $F(e^{j\omega}) = B(e^{j\omega}/\{1 + P(e^{j\omega})\})$ , e  $B(e^{j\omega})$  é a resposta em frequência do filtro passa-faixas ideal (que é um em  $\Omega$  e zero fora

de  $\Omega$ ). Os demais termos restantes em 3.4.7 são formados de forma análoga utilizando-se os índices de  $\Delta$  e  $\Gamma$ .

No segundo passo do algoritmo a extrapolação é conseguida em 3.4.6 filtrando-se uma seqüência de duração finita formada a partir de  $\mathbf{u}$  e  $\mathbf{v}$ . Portanto, a extrapolação concentrada em energia pode ser facilmente obtida por um algoritmo de dois passos similar ao algoritmo MNLS.

### 3.5 Dimensão Essencial

Um sinal não pode ser considerado limitado em banda e também limitado no tempo. No entanto, o sinal pode ser considerado essencialmente limitado no tempo a um conjunto de índices  $\Gamma$ . Dado um número real positivo  $\epsilon < 1$ , um sinal  $y$  é dito ser essencialmente limitado em tempo a um intervalo  $\Gamma$  em relação a  $\epsilon$  se menos do que uma fração de  $\epsilon$  de sua energia total estiver fora de  $\Gamma$ , ou seja,  $y$  é essencialmente limitado em tempo a  $\Gamma$  se  $\|\mathbf{W}\mathbf{y}\|^2 \geq (1 - \epsilon)\|\mathbf{y}\|^2$ . Pode-se definir então o conjunto  $G_\epsilon(\Omega, \Gamma)$  de sinais discretos, limitados em banda e de energia finita que são essencialmente limitados no tempo a  $\Gamma$  como:

$$G_\epsilon(\Omega, \Gamma) \triangleq \{\mathbf{y} \in B_\Omega : \|\mathbf{W}\mathbf{y}\|^2 \geq (1 - \epsilon)\|\mathbf{y}\|^2\} \quad (3.5.1)$$

Se  $N$  é o número de amostra em  $\Gamma$ , aparentemente, para um  $\epsilon$  pequeno, sinais que pertencem ao conjunto  $G_\epsilon(\Omega, \Gamma)$  possuem  $N$  graus de liberdades significantes. No entanto, a partir do teorema clássico de dimensionalidade, sabe-se que esses sinais, em alguns casos, possuem menos graus de liberdade, como é o caso de sinais superamostrados.

A abordagem do método de extrapolação com Dimensão Essencial, proposto por Dharanipragada e Arun [32], é aproximar  $G_\epsilon(\Omega, \Gamma)$  por um espaço  $M$ -dimensional ( $M < N$ ) e portanto reduzir o problema de extrapolação ao de encontrar  $M$  parâmetros que são linearmente relacionado às  $p$  observações. Se  $M$  não fosse maior do que  $p$ , as observações

especificariam unicamente o sinal. A tarefa, portanto, é selecionar a dimensão  $M$  e então, selecionar o melhor subespaço  $M$ -dimensional para aproximar a  $G_\epsilon(\Omega, \Gamma)$ . A escolha de  $M$  pode ser feita de forma racional a partir dos auto-valores da matriz  $\mathbf{WBW}^*$  de tamanho  $N \times N$ .

Dado um erro tolerável  $\epsilon' > 1 - \frac{1-\epsilon}{\lambda_1}$ , pode-se encontrar o maior inteiro  $M$  para o qual exista um subespaço  $M$ -dimensional  $S_M$  tal que para todo  $y \in G_\epsilon(\Omega, \Gamma)$ , uma aproximação  $\hat{y} \in S_M$  possa ser encontrado satisfazendo:

$$\frac{\|y - \hat{y}\|^2}{\|y\|^2} \quad (3.5.2)$$

Este  $M$  é o maior inteiro para o qual:

$$\lambda_M \geq \frac{(1 - \epsilon) - \lambda_1(1 - \epsilon')}{\epsilon'} \quad (3.5.3)$$

Para qualquer inteiro fixo  $M < N$ , entre todos os subespaços  $M$ -dimensionais  $S_M$ , o espaço  $S_{\phi, M} \triangleq \text{span}\{\phi_1, \phi_2, \dots, \phi_M\}$  se aproxima de  $G_\epsilon(\Omega, \Gamma)$  no senso em que produz o menor erro de extrapolação no pior caso.

Se para todo número real  $\epsilon' > 1 - \frac{1-\epsilon}{\lambda_1}$ , seja  $M(\epsilon', \epsilon, \Omega, \Gamma)$  o maior valor do inteiro como em 3.5.3, então para todo  $y \in G_\epsilon(\Omega, \Gamma)$ , existe um  $\hat{y} \in S_{\phi, M(\epsilon', \epsilon, \Omega, \Gamma)}$  de modo que:

$$\|y - \hat{y}\|^2 \leq \epsilon' \|y\|^2 \quad (3.5.4)$$

Portanto  $M(\epsilon', \epsilon, \Omega, \Gamma)$  parâmetros são suficientes para representar um sinal do conjunto  $G_\epsilon(\Omega, \Gamma)$  dentro de uma tolerância de erro  $\epsilon'$ .  $M(\epsilon', \epsilon, \Omega, \Gamma)$  é considerado como sendo a dimensão essencial de  $G_\epsilon(\Omega, \Gamma)$  e será denominado dimensão tempo-largura de banda.

O método de extrapolação com Dimensão Essencial propõe então que seja utilizado a menor dimensão  $M$  possível para a qual o erro no pior caso não seja maior do que uma

tolerância  $\epsilon'$  especificado. Se a extrapolação (que reside em  $S_{\phi, M}$ ) for  $\alpha_1\phi_1, \alpha_2\phi_2, \dots, \alpha_M\phi_M$ , então as observações  $\mathbf{z} = \mathbf{T}\mathbf{y}$  são aproximadas por  $\alpha_1\mathbf{T}\phi_1, \alpha_2\mathbf{T}\phi_2, \dots, \alpha_M\mathbf{T}\phi_M$ . Portanto tem-se o seguinte algoritmo:

Passo 0) Pré-computar a dimensão tempo-largura de banda  $M$

Passo 1) Solucionar  $A\alpha = \mathbf{z}$  para  $\alpha$ . Aqui,  $A$  é construído a partir das primeiras  $M$  seqüências DPSS, utilizando-se  $\mathbf{T}\phi_j$  como a  $j$ -ésima coluna para  $j = 1, 2, \dots, M$ , isto é:

$$A = \begin{bmatrix} \phi_1(t_1) & \phi_2(t_1) & \cdots & \phi_M(t_1) \\ \phi_1(t_2) & \phi_2(t_2) & \cdots & \phi_M(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_1(t_p) & \phi_2(t_p) & \cdots & \phi_M(t_p) \end{bmatrix} \quad (3.5.5)$$

e  $\mathbf{z}$  é o vetor de observações. Quando o sistema é indeterminado, uma solução de mínimos quadrados pode ser utilizada.

Passo 2) Utilizando  $\alpha$ , construir

$$\hat{y} = \sum_{i=1}^M \alpha_i \phi_i \quad (3.5.6)$$

O algoritmo incorpora informações adicionais como o conhecimento de  $\Gamma, \Omega$  e  $\epsilon$  na solução. Também explora a dimensão reduzida do conjunto de sinais limitados em banda e essencialmente limitados em tempo. Outra vantagem é que não impõe restrições nas posições dos índices de observação e concentração de energia ou na banda passante e portanto pode ser utilizado em configurações variadas.

Em muitas aplicações,  $\Gamma, \Omega$  e  $\epsilon$  são fixos, o que permite a computação da dimensão essencial e das seqüências DPS uma única vez. Além do mais, se o conjunto de índices de observações é conhecido com antecedência, a matriz de coeficientes  $A$  também pode ser construída uma única vez, visto que só depende das posições dos índices de observações e não dos valores das observações. O número de parâmetros, que é igual à dimensão essencial do conjunto  $G_\epsilon(\Omega, \Gamma)$ , não depende do número de observações. Portanto se o número de

observações for maior do que a dimensão essencial, pode-se conseguir diminuição de erro encaixando-se os mínimos quadrados dos dados ao menor espaço dimensional.

## Capítulo 4

# Aplicação da Extrapolação nos Coeficientes da DStr

O objetivo deste trabalho é propor um método que produza uma imagem digital com mais detalhes no processo de interpolação, através da recuperação das componentes de alta frequência da imagem, perdidos no processo de captura ou de decimação da imagem, buscando uma aproximação da imagem contínua original.

A abordagem aqui se restringe ao caso específico de imagens naturais (fotografias) monocromáticas (com 256 níveis de cinza) processada em blocos. Mas as aplicações do método não são restritas a esse problema, podendo a idéia ser aplicada a diversos tipos de imagens (médicas, sensoriamento remoto ou documentos digitalizados, por exemplo), porém a sua eficiência precisa ser estudada individualmente.

### 4.1 Considerações Sobre o Algoritmo de Extrapolação

No problema abordado neste trabalho, onde se busca conseguir uma imagem interpolada (ampliada) a partir de uma imagem dada, através de processamento em blocos, o que se procura fazer é extrapolar os coeficientes da transformada utilizada no processo de interpolação. Como a transformada é separável, então o processo pode ser aplicado somente

nas linhas ou nas colunas do bloco, independentemente.

Os algoritmos de extrapolação se baseiam em informações a respeito do sinal original  $y[n]$  para, a partir da seqüência limitada  $z[n], n \in \Delta$ , estimar  $y[n]$  fora de  $\Delta$ . No problema abordado neste trabalho o sinal original  $y[n]$  é totalmente desconhecido a menos da seqüência limitada  $z[n]$ . Portanto não existe informação real sobre o sinal, apenas considerações supostas, ou conhecimento geral a partir de observações de sinais de teste específicos.

Por exemplo, ao observar-se os espectros das seqüências formadas pelos coeficientes das transformadas de uma variedade de imagens, constata-se que esses coeficientes, quando interpolados, são de freqüência limitada em banda passa-baixas. Porém não é possível determinar, somente a partir de  $z[n]$ , a freqüência exata a qual o sinal  $y[n]$  é limitado em banda, para cada linha e coluna de cada bloco da imagem e uma estimativa incorreta pode levar a uma extrapolação com resultado pior do que a estimativa com valor zero (filtragem passa-baixas ideal). O mesmo ocorre com os demais parâmetros utilizados pelos algoritmos de extrapolação.

A grande dificuldade de se escolher os parâmetros, é que nesse problema há uma quantidade muito grande de sinais diferentes, e os parâmetros que podem ser bons para um determinado sinal, podem resultar em estimativas com erro muito acentuado. O ideal seria desenvolver um modelo para que esses parâmetros estivessem dentro de limites, determinados pelo conjunto de observações  $z[n]$  ou então baseados em estatísticas para conjuntos de imagens específicas (radar, médicas, paisagens ou faces). Devido ao tempo limitado deste trabalho, e à complexidade desse processo, não foi possível desenvolver tal modelo ou analisar dados estatísticos. Mais detalhes deverão ser incluídos em trabalhos futuros.

A escolha do algoritmo de extrapolação a ser utilizado neste trabalho, não se baseou na

eficácia do mesmo na recuperação dos coeficientes, mas sim na capacidade de se determinar os parâmetros necessários, principalmente porque a eficácia do algoritmo depende da boa estimativa desses parâmetros. Portanto decidiu-se adotar o critério da simplicidade de implementação e utilizar parâmetros empíricos, com o intuito de simplesmente demonstrar as melhorias possíveis e motivar o estudo nessa área.

O algoritmo de Redução da Energia do Erro de Papoullis e Gerchberg [24] é iterativo e só converge para o sinal original quando  $n$ , a iteração do algoritmo, tende para o infinito. Além disso, a eficácia da extrapolação depende da estimação correta da extensão do sinal original. O algoritmo também é muito sensível a ruídos no sinal, resultando em estimativas muito ruins, se as iterações não forem devidamente truncadas.

O algoritmo de Mínimos Quadrados Ponderado não é iterativo, sendo calculado em dois passos e leva a uma extrapolação boa quando a forma do espectro do sinal original é conhecido. Porém, a informação exata sobre a distribuição da energia do sinal original no domínio da frequência não está disponível no problema abordado e uma estimativa satisfatória é difícil de se conseguir.

O algoritmo de estimação Limitada em Banda e Concentrada no Tempo (BLTC) também é não iterativo e leva a uma extrapolação boa, pois não minimiza a energia do sinal onde a extrapolação é esperada ser boa. Esse algoritmo tira proveito implicitamente do fato da energia do sinal estar concentrada na região conhecida. Porém quando a energia não está concentrada na região conhecida, o desempenho do algoritmo não é tão otimizado. Uma das maiores dificuldades desse algoritmo, da forma como ele é proposto, é o uso de Sequências Esferoidais Estendidas Discretas (DPSS), que consomem muito processamento e são difíceis de serem calculadas. Apesar disso, é possível aproveitar o conceito de Dimensão Reduzida, para extrapolar somente uma quantidade de pontos que possuam erros menores.

O algoritmo de Concentração de Energia também é não iterativo e seu autor propõe diversas formas de otimizar o seu processamento [31], dividindo o sistema de equações em sistemas menores e pré-computando as matrizes antes de serem aplicadas a cada sinal, já que suas construções dependem somente de alguns parâmetros. Infelizmente, a capacidade do algoritmo alcançar boas extrapolações, depende do conhecimento da região onde a energia do sinal original, no domínio do tempo, está concentrada. Esse parâmetro, além de desconhecido no problema deste trabalho é difícil de ser estimado e a estimativa errada pode levar a uma extrapolação com erro muito grande. Se a região de concentração for considerada como a região onde se encontra o segmento conhecido, a solução recai sobre a solução de Norma Mínima e Mínimos Quadrados.

O algoritmo de Dimensão Essencial é o algoritmo que apresenta melhores resultados, dentre os desenvolvidos até o momento. Sua eficiência se deve à capacidade de empregar a maior quantidade de informação disponível sobre o sinal original em relação aos demais algoritmos. Pelo mesmo motivo dos algoritmos anteriores, sua aplicação no problema deste trabalho se torna impraticável, pois utiliza informação do conteúdo espectral, distribuição da energia no tempo e fração da energia no segmento desconhecido. Tais informações não estão disponíveis e se fossem assumidas como o caso clássico (passa-baixas ideal, distribuição contínua da energia dentro do segmento conhecido e nenhuma fração da energia no segmento desconhecido) o algoritmo se resumiria ao algoritmo MNLS, que é um caso específico deste.

Se o critério de capacidade de determinação dos parâmetros para os algoritmos for o único critério utilizado para se escolher o algoritmo de extrapolação, a escolha é sem dúvida o algoritmo de Modificação Iterativa da Norma Mínima, pois ele não necessita de informação adicional alguma sobre o sinal original. Ele assume somente que o sinal é limitado em banda, estima iterativamente a distribuição da energia do sinal no domínio da

freqüência baseado no espectro de potência calculado sobre os pontos extrapolados a cada iteração. Apresenta boa eficiência na aproximação dos pontos extrapolados aos pontos do sinal original. Porém esse método é iterativo, e apesar de convergir rápido a uma solução aceitável, ele não pode ser aplicado a um conjunto de vetores ao mesmo tempo. Como os dados neste trabalho são tratados como um conjunto de vetores coluna que compõe o bloco, a aplicação sobre cada coluna independentemente resultaria em um cálculo muito demorado para uma imagem de dimensões grandes (Ex.: maiores do que  $512 \times 512$ ).

O algoritmo de Norma Mínima e Mínimos Quadrados foi então escolhido, pois leva a uma solução única, possui implementação simples, é aplicável a um conjunto de vetores ao mesmo tempo utilizando multiplicação de matrizes e necessita como parâmetro somente a freqüência de corte do filtro passa-baixas a qual o sinal é considerado como sendo limitado. Esse parâmetro pode ser facilmente estimado a partir do segmento conhecido. Vale ressaltar que o algoritmo de Redução de Energia converge para a solução MNLS. Os demais algoritmos são formas mais genéricas do algoritmo MNLS, que só conseguem obter melhores estimativas se as informações adicionais necessárias estiverem disponíveis ou puderem ser estimadas com precisão.

## 4.2 Descrição do Método Proposto

O método proposto consiste nos seguintes passos: dada um imagem bidimensional monocromática (em escala de cinzas) de tamanho qualquer  $M \times N$ , a taxa de interpolação desejada  $tx$  (valor inteiro) e o tamanho  $bl$  do bloco:

- Dividir a imagem em blocos de tamanho  $bl + 1$  onde a última linha e a última coluna pertencem aos blocos seguintes conforme o processo de interpolação por DSTr-2D, explicado no Capítulo 2;

- Aplicar a rotação de eixos sobre as colunas, armazenando a primeira e a última linha para serem utilizadas no processo de rotação inversa;
- Excluir a primeira e a última linha que conterão valores nulos. Adicionar à matriz resultante linhas com valores nulos para fazer a interpolação dos coeficientes;
- Aplicar a DST-1D às colunas da matriz;
- Extrapolar os coeficientes da DST de cada coluna, recuperando as componentes de alta frequência;
- Subamostrar a matriz para obter a quantidade correta de coeficientes;
- Aplicar a rotação inversa, utilizando a primeira e última linha armazenadas anteriormente;

Tem-se até agora as colunas interpoladas. O processo pode ser facilmente completado, transpondo o bloco e aplicando-se os mesmos passos, transpondo novamente ao final e obtendo-se o bloco interpolado nas linhas e nas colunas, conforme ilustrado na Fig. 4.1. A Fig. 4.2 apresenta um diagrama de blocos correspondente aos procedimentos.

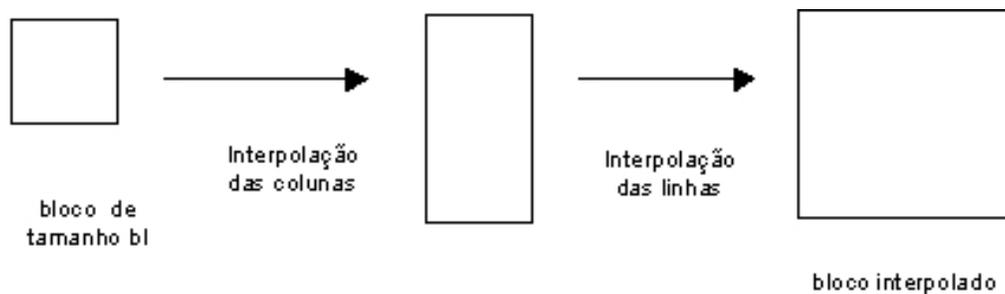


Figura 4.1: Ilustração da seqüência de interpolação de um bloco.

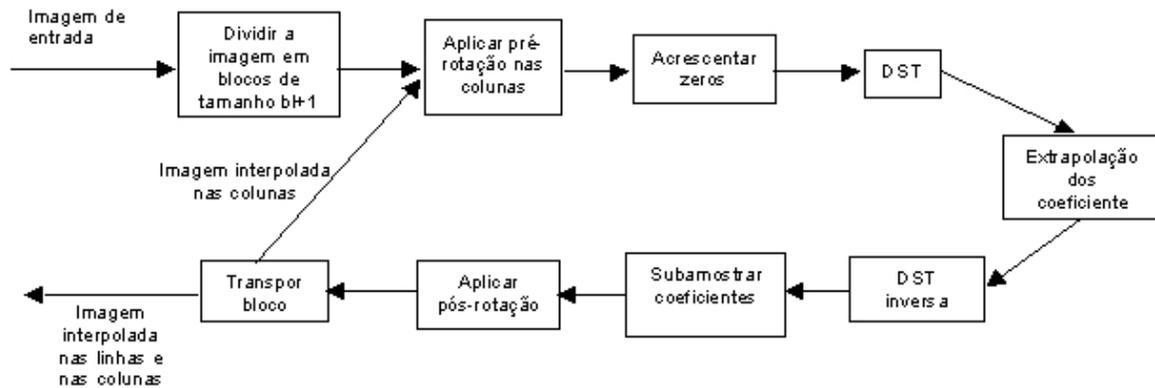


Figura 4.2: Diagrama de blocos representando o método proposto.

Devido às características separáveis dos algoritmos utilizados no método, e às vantagens implícitas da linguagem de programação utilizada, escolheu-se aplicar os passos primeiramente somente nas colunas e em seguida reaplicá-los na transposta do resultado. Desse modo simplifica-se a implementação, com um algoritmo único para as duas dimensões do sinal e obtêm-se o mesmo resultado.

O processamento em bloco é um requisito praticamente obrigatório para sinais de grandes dimensões, como é o caso de imagens digitais. Apesar de ser uma solução para as limitações físicas dos sistemas computacionais, é ao mesmo tempo a causadora de artifícios que degradam a qualidade da imagem. Apesar do fato de que blocos quadrados são adotados como padrão, o avanço nas tecnologias de componentes eletrônicos utilizados nos computadores atuais, permite que se utilize mais memória e processamento sem perdas de desempenho.

Na implementação do método proposto, foram utilizados blocos com a quantidade de linhas especificados pelo usuário em  $bl$ , porém a quantidade de colunas é igual à largura da imagem. Esse procedimento foi escolhido, porque a interpolação é realizada primeiro

nas colunas e depois nas linhas. Percebe-se que não há ganho de qualidade no sinal em relação ao método tradicional de separação em blocos (quadrados). Porém, devido à característica algébrica dos algoritmos, é possível obter um código mais otimizado e simples quando a linguagem utilizada é o ©Matlab e também um ganho em velocidade de processamento devido ao número reduzido de acessos à memória.

Um ponto importante a ser considerado é o fato de que imagens digitais não são infinitas e nem sempre suas dimensões possuem valores múltiplos do tamanho do bloco, podendo haver blocos com interrupções abruptas, causando artifícios danosos à qualidade da imagem, se as bordas não forem tratadas.

No capítulo sobre interpolação, nas seções sobre interpolações utilizando-se transformadas, foi evidenciado a vantagem da simetria par para suavizar a borda do sinal truncado. Baseando-se no mesmo princípio, utilizou-se a simetria par nas bordas da imagem para completar os pontos que faltavam para a formação do bloco.

Em sua Tese de Doutorado, Pelaes [3] demonstrou a superioridade da DSTr às demais transformadas citadas neste trabalho, no processo de interpolação. Portanto a rotação de eixos é aplicada às colunas do bloco em cada passo para minimizar o efeito de blocos causado pelo truncamento do sinal.

A DST é a transformada que apresenta menor correlação residual [18], conforme citado no Capítulo 2, portanto sua escolha parece ser a mais indicada. Nesse caso, serão seus coeficientes que serão extrapolados a fim de se tentar recuperar as componentes de alta frequência perdidas. Como os algoritmos de extrapolação trabalham com sinais limitados em banda e quanto mais estreita a banda do sinal, maior a eficiência do algoritmo, as colunas do bloco são concatenadas a uma seqüência de zeros (*zero pad*) após a rotação de eixos e antes de se aplicar a DST. Esse procedimento resulta em coeficientes da transformada interpolados e portanto com espectro de banda mais estreita. É interessante que

a quantidade de zeros seja um número inteiro múltiplo do tamanho do bloco conforme explicado na seção 2.1.5.

Devido ao fato de que informações sobre parâmetros do sinal original nem sempre estão disponíveis e à ausência de dados estatísticos, optou-se pelo algoritmo MNLS que necessita somente da frequência de corte do filtro passa-baixas, que é estimado de forma empírica e aplicado na imagem de forma global. Imagina-se que a estimação desse parâmetro para cada coluna do bloco poderia resultar em uma extrapolação mais precisa individualmente. Porém, não foi possível investigar tal hipótese devido à dificuldade de se conseguir tal estimativa a partir do segmento conhecido. Além do mais, se fosse possível calcular a frequência de corte para cada coluna, o algoritmo se tornaria muito lento, pois seria necessário construir e inverter uma matriz correspondente ao operador limitador de banda para cada coluna, resultando em muitas multiplicações de matrizes. Utilizando-se apenas uma matriz para cada bloco, é necessário somente uma multiplicação de matriz por bloco. Além disso é possível otimizar o algoritmo construindo-se a matriz do operador limitador de banda apenas uma vez se a frequência de corte escolhida for a mesma para todos os blocos da imagem.

A frequência utilizada no algoritmo de extrapolação é o inverso da taxa de interpolação dos coeficientes da DSTr, ou seja  $f_c = tx_{coef}^{-1}$  ( $f_c$  é a frequência de corte e  $tx_{coef}$  é um número inteiro correspondente à taxa de interpolação dos coeficientes). Esse valor foi escolhido baseado no fato de que quanto maior a taxa de amostragem, tanto menor a frequência máxima do sinal amostrado, na mesma proporção. Em outras palavras, a teoria de amostragem, demonstra que existe uma relação recíproca entre o domínio do tempo e o domínio da frequência, portanto ao se fazer uma expansão no domínio do tempo, acontece uma compressão no domínio das frequências e vice-versa. Logo, ao considerar-se os coeficientes como um sinal no domínio do tempo, sua interpolação (expansão) levará

a uma compressão no domínio da frequência. A eficiência da utilização desse valor na aplicação do algoritmo extrapolação foi testada aplicando-se vários valores inteiros de taxa de interpolação e todos os valores de frequência de corte com precisão de quatro casas decimais e comparando-se o resultado do método proposto com a imagem original da seguinte forma:

- Dada uma imagem digital qualquer de tamanho  $M \times N$ , obter uma imagem reduzida pela metade do tamanho, isto é,  $M/2 \times N/2$ ;
- Interpolar a imagem com o método proposto, utilizando-se todos os valores de frequências  $0 > f_c > 1$ , com precisão de quatro casas decimais, para um valor inteiro de taxa de interpolação dos coeficientes;
- Comparar a imagem interpolada com a imagem original utilizando-se a relação sinal-ruído (SNR);
- Obter a frequência de corte que apresenta o maior valor SNR;

Os valores de frequência de corte que obtiveram maiores valores para uma dada taxa de interpolação foram valores muito próximos ao inverso da taxa de interpolação  $tx_{coef}^{-1}$ , demonstrando que essa estimativa pode ser uma boa escolha para o algoritmo.

No capítulo seguinte é feita a comparação entre o método proposto e o método de interpolação utilizando-se DSTr com extrapolações para demonstrar as melhoras alcançadas. Os resultados são mostrados e comentados.

# Capítulo 5

## Resultados Obtidos

### 5.1 Comparação

Com o objetivo de comprovar a idéia de que é possível melhorar a interpolação de uma imagem digital, estimando-se as componentes de alta frequência a partir das componentes de baixa frequência da transformada, compara-se o resultado da interpolação utilizando-se o método proposto com o método de interpolação utilizando DSTr. Essa comparação foi escolhida, porque Pelaes em [3], já comprovou que a interpolação por DSTr é superior às demais formas de interpolação utilizando-se transformadas. Além do mais, apresenta as mesmas características e os mesmos artifícios e benefícios do método proposto, visto que o último é uma extensão do primeiro.

Para se fazer a comparação entre os dois métodos, escolheu-se a imagem "Lena" de tamanho  $512 \times 512$  mostrada na Fig. 5.1. Fez-se a decimação dessa imagem obtendo-se uma imagem com a metade do tamanho ( $256 \times 256$ ) e aplicou-se os dois métodos de interpolação à imagem reduzida, com blocos de mesmo tamanho. Calculou-se então o SNR e o PSNR entre as imagens obtidas e a imagem original. Os resultados para diferentes tamanhos de blocos são mostrados na TABELA 5.1.

O mesmo procedimento foi aplicado em outras imagens com características diferen-



Figura 5.1: Imagem original "Lena"  $512 \times 512$ .

tes. As imagens utilizadas foram "Peppers" (Fig. 5.2), "Baboo" (Fig. 5.3), "Rabbit" (Fig. 5.4), e "Figuras Geométricas" (Fig. 5.5). Essas imagens apresentam diferentes características apesar de serem todas imagens naturais (fotografias) com exceção de "Figuras Geométricas" que é sintética. A imagem "Peppers" apresenta poucas áreas com textura, mas uma grande quantidade de bordas. A imagem "Baboo" apresenta muitas áreas com textura. A imagem "Rabbit" apresenta poucas áreas com textura e muito poucas bordas, com pixels com mesmo valor por quase toda imagem. A imagem "Figuras Geométricas" não apresenta nenhuma textura, somente bordas.

Na TABELA 5.2 estão os valores de SNR e PSNR para blocos de tamanhos 8 e 16, para a imagem "Peppers". Da mesma forma estão na TABELA 5.3, os valores para a imagem "Baboo", na TABELA 5.4 os valores para a imagem "Rabbit" e na TABELA 5.5 os valores

TABELA 5.1: SNR ENTRE AS IMAGENS OBTIDAS E A IMAGEM ORIGINAL. IMAGEM "LENA".

Tam. bloco	8		16	
	SNR	PSNR	SNR	PSNR
DSTr	28.4394	34.0433	28.3088	33.9128
DSTrEx	28.4746	34.0738	28.3315	33.9326

para a imagem "Figuras Geométricas".

Tabela 5.2: Valores de SNR e PSNR entre a imagem original e a restauração com os métodos. Imagem "Peppers".

Tam. bloco	8		16	
	SNR	PSNR	SNR	PSNR
DSTr	25.7127	31.4794	25.6074	31.3741
DSTrEx	25.7278	31.4945	25.6194	31.3862

TABELA 5.3: VALORES DE SNR E PSNR ENTRE A IMAGEM ORIGINAL E A RESTAURAÇÃO COM OS MÉTODOS. IMAGEM "BABOO".

Tam. bloco	8		16	
	SNR	PSNR	SNR	PSNR
DSTr	16.1481	21.6164	16.0153	21.4835
DSTrEx	16.1638	21.6321	16.0282	21.4965

Os resultados obtidos a partir das comparações mostram que é possível melhorar a interpolação, recuperando-se as componentes de alta frequência. Como a maioria da informação relevante ao olho humano, assim como a energia do sinal, fica concentrada nas componentes de baixa frequência, são necessários muitos coeficientes corretos da transformada para se conseguir um maior ganho em qualidade. É por esse motivo que os resultados mostram uma diferença tão pequena entre os dois métodos.



Figura 5.2: Imagem original "Peppers".

Para comprovar este fato, fez-se um teste aplicando-se a transformada sobre a imagem original e substituindo-se progressivamente os coeficientes de frequência mais alta de cada bloco, por zero até a metade do tamanho do bloco, simulando o processo inverso de extrapolação. A imagem reconstruída foi comparada com a imagem original utilizando-se SNR. Os gráficos mostram os valores obtidos para cada quantidade de linhas de coeficientes de cada bloco, sendo metade nas linhas e a outra metade nas colunas. Para esse teste foram utilizadas as imagens "Lena", "Peppers", "Baboo" e os resultados para cada imagem são mostrados nos gráficos das Figs. 5.6, 5.7 e 5.8 respectivamente. Os blocos utilizados foram de tamanho 16, simulando uma interpolação onde um bloco de tamanho 8 é interpolado para um bloco de tamanho 16, acrescentado coeficientes.

É possível notar que com poucos coeficientes além do seguimento conhecido, o ganho

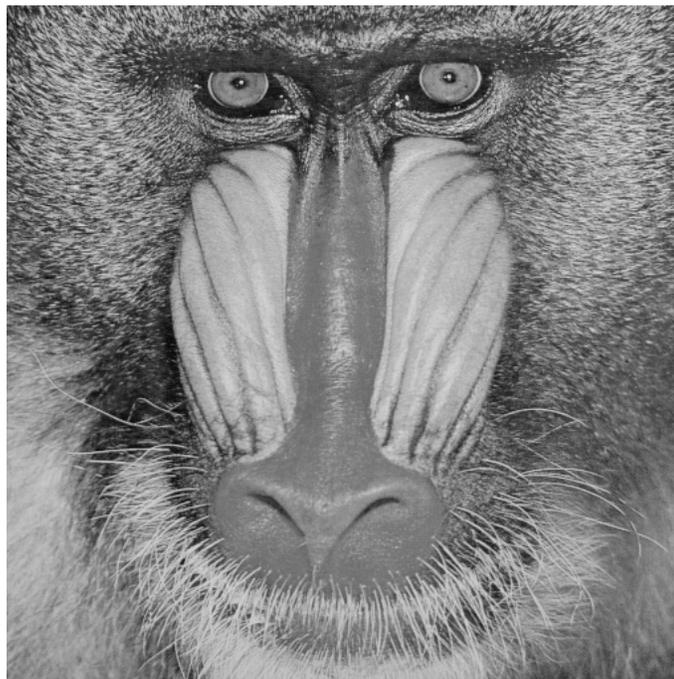


Figura 5.3: Imagem original "Baboo".

em qualidade é muito pequeno, mesmo tendo os coeficientes corretos. Somente quando muitos coeficientes corretos são acrescentados, o ganho é considerável.

Imagina-se que seja possível melhorar esses resultados encontrando-se os parâmetros necessários aos algoritmos de extrapolação que possuem melhor desempenho e aplicá-los individualmente a cada coluna de cada bloco. Porém, mesmo que isso seja possível, é preciso levar em consideração que tanto a imagem original quanto a imagem reduzida possuem ruído, ao qual os algoritmos de extrapolação são sensíveis. Além do mais, a imagem reduzida pode ser conseguida de diversos modos e mesmo quando é filtrada para evitar sobreposição espectral (*aliasing*), muitos parâmetros podem ser utilizados (como a ordem do filtro, por exemplo) resultando em coeficientes de baixa frequência com valores diferentes dos coeficientes de baixa frequência da imagem original para a DSTr. Ou seja, os



Figura 5.4: Imagem original "Rabbit".

TABELA 5.4: VALORES DE SNR E PSNR ENTRE A IMAGEM ORIGINAL E A RESTAURAÇÃO COM OS MÉTODOS. IMAGEM "RABBIT".

Tam. bloco	8		16	
	SNR	PSNR	SNR	PSNR
DSTr	40.0533	41.1046	40.1120	41.1633
DSTrEx	40.0672	41.1185	40.1276	41.1789

coeficientes da transformada da imagem reduzida podem não coincidir com os coeficientes da imagem original que estão nas mesmas posições.

Quando os coeficientes da transformada da imagem reduzida são iguais aos primeiros coeficientes da transformada da imagem original, a diferença entre os métodos de interpolação sem extrapolação e com extrapolação fica muito mais evidente, onde o método com extrapolação dos coeficientes resulta em um ganho de qualidade maior.

Este fato pode ser verificado, aplicando a transformada sobre a imagem original e substituindo-se os últimos coeficientes por zero, como seria feito no método de interpolação sem extrapolação e substituindo-se os últimos coeficientes por extrapolações, como seria

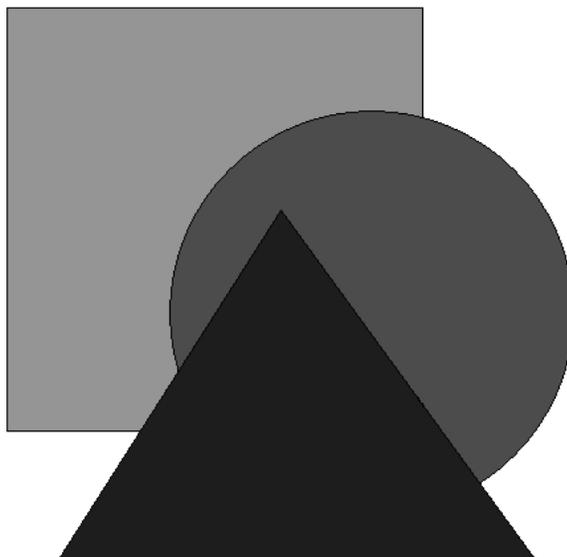


Figura 5.5: Imagem original "Figuras Geométricas".

feito no caso de interpolação com extrapolação. Comparando-se as imagens recuperadas com a imagem original, observa-se que o segundo caso apresenta melhor resultado. Como teste específico, foi utilizada a imagem "Lena"  $512 \times 512$  e os valores SNR e PSNR são mostrados na TABELA 5.6. Também foram utilizadas as imagens "Peppers" (Fig. 5.2), "Baboo" (Fig. 5.3), "Rabbit" (Fig. 5.4), e "Figuras Geométricas" (Fig. 5.5) e os valores obtidos são mostrados nas TABELAS 5.7, 5.8, 5.9 e 5.10, respectivamente.

Com esses resultados nota-se que, a diferença entre os métodos se torna bastante significativa quando a extrapolação é aplicada sobre os primeiros coeficientes da transformada da imagem original. Em outras palavras, se for possível obter os mesmos coeficientes, o método proposto apresentará uma melhora de qualidade muito maior na interpolação da imagem.

TABELA 5.5: VALORES DE SNR E PSNR ENTRE A IMAGEM ORIGINAL E A RESTAURAÇÃO COM OS MÉTODOS. IMAGEM "FIGURAS GEOMÉTRICAS".

Tam. bloco	8		16	
	SNR	PSNR	SNR	PSNR
DSTr	21.7059	24.6318	21.7004	24.6264
DSTrEx	21.7210	24.6470	21.7066	24.6326

TABELA 5.6: VALORES DE SNR E PSNR ENTRE A IMAGEM ORIGINAL E AS RESTAURAÇÕES UTILIZANDO SUBSTITUIÇÃO POR ZERO E POR EXTRAPOLAÇÃO. IMAGEM "LENA".

Tam. bloco	8		16	
	SNR	PSNR	SNR	PSNR
DSTr	30.2423	35.8462	30.7357	36.3396
DSTrEx	33.8590	39.4630	32.6733	38.2772

TABELA 5.7: VALORES DE SNR E PSNR ENTRE A IMAGEM ORIGINAL E AS RESTAURAÇÕES UTILIZANDO SUBSTITUIÇÃO POR ZERO E POR EXTRAPOLAÇÃO. IMAGEM "PEPPERS".

Tam. bloco	8		16	
	SNR	PSNR	SNR	PSNR
DSTr	27.8210	33.5877	28.2054	33.9722
DSTrEx	30.4866	36.2533	29.4994	35.2661

TABELA 5.8: VALORES DE SNR E PSNR ENTRE A IMAGEM ORIGINAL E AS RESTAURAÇÕES UTILIZANDO SUBSTITUIÇÃO POR ZERO E POR EXTRAPOLAÇÃO. IMAGEM "BABOO".

Tam. bloco	8		16	
	SNR	PSNR	SNR	PSNR
DSTr	18.6379	24.1061	18.8608	24.3291
DSTrEx	21.4843	26.9525	20.4171	25.8853

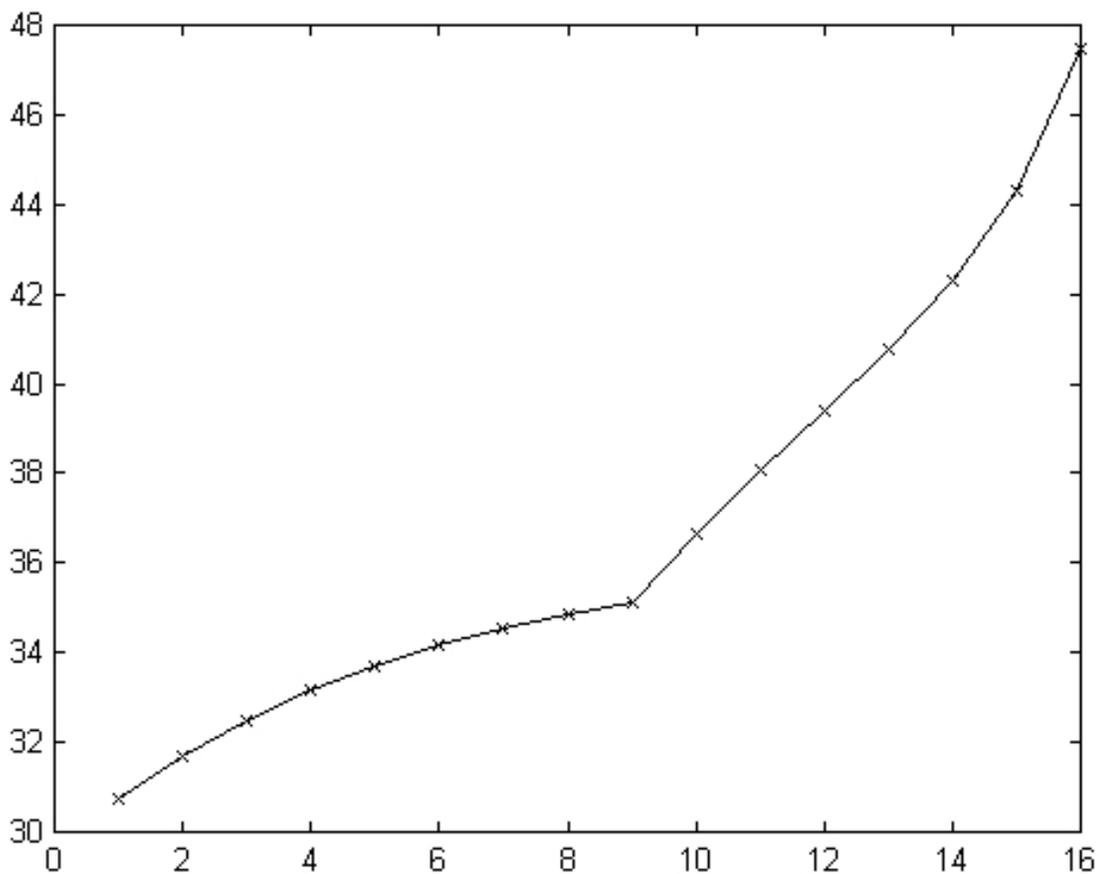


Figura 5.6: Gráfico SNR × quantidade de coeficientes. Imagem "Lena".

TABELA 5.9: VALORES DE SNR E PSNR ENTRE A IMAGEM ORIGINAL E AS RESTAURAÇÕES UTILIZANDO SUBSTITUIÇÃO POR ZERO E POR EXTRAPOLAÇÃO. IMAGEM "RABBIT".

Tam. bloco	8		16	
	SNR	PSNR	SNR	PSNR
DSTr	41.2267	42.2780	42.1704	43.2217
DSTrEx	45.8748	46.9261	44.2534	45.3047

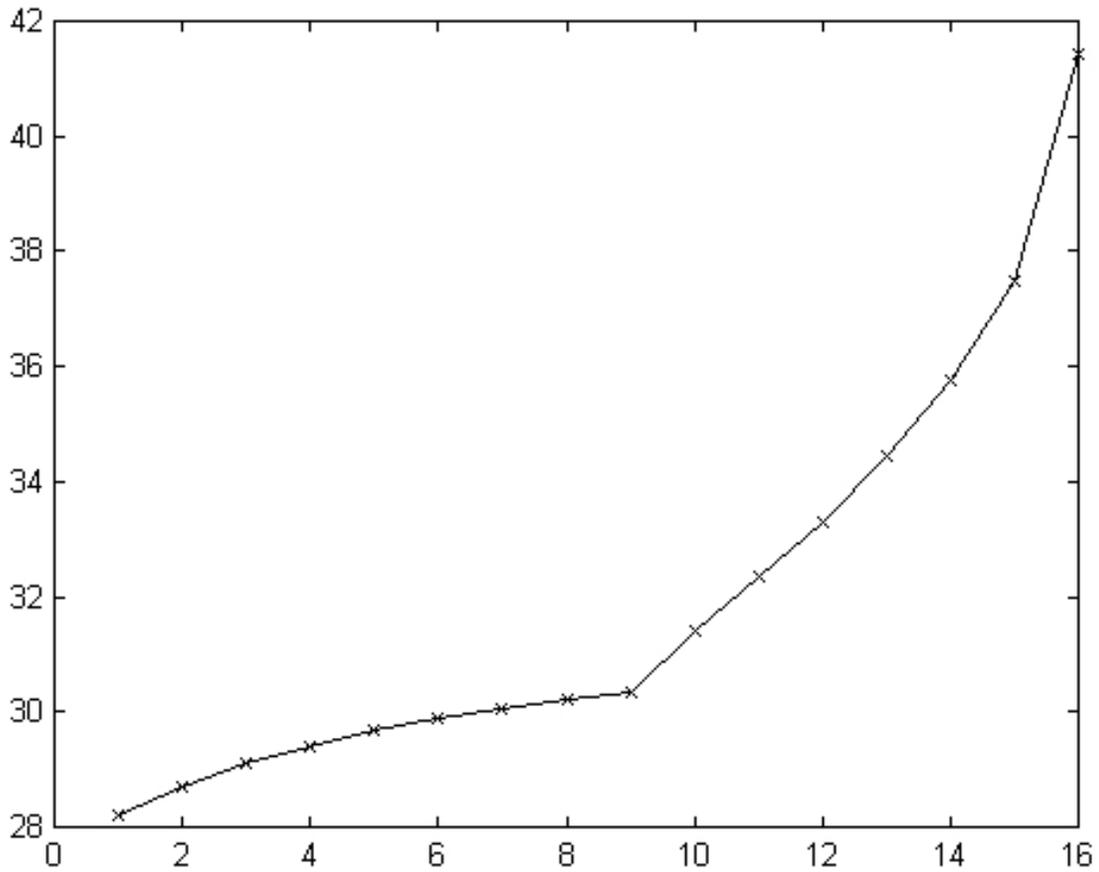


Figura 5.7: Gráfico SNR  $\times$  quantidade de coeficientes. Imagem "Peppers".

TABELA 5.10: VALORES DE SNR E PSNR ENTRE A IMAGEM ORIGINAL E AS RESTAURAÇÕES UTILIZANDO SUBSTITUIÇÃO POR ZERO E POR EXTRAPOLAÇÃO. IMAGEM "FIGURAS GEOMÉTRICAS".

Tam. bloco	8		16	
	SNR	PSNR	SNR	PSNR
DSTr	23.9301	26.8561	24.2605	27.1865
DSTrEx	26.0000	28.9260	25.3053	28.2313

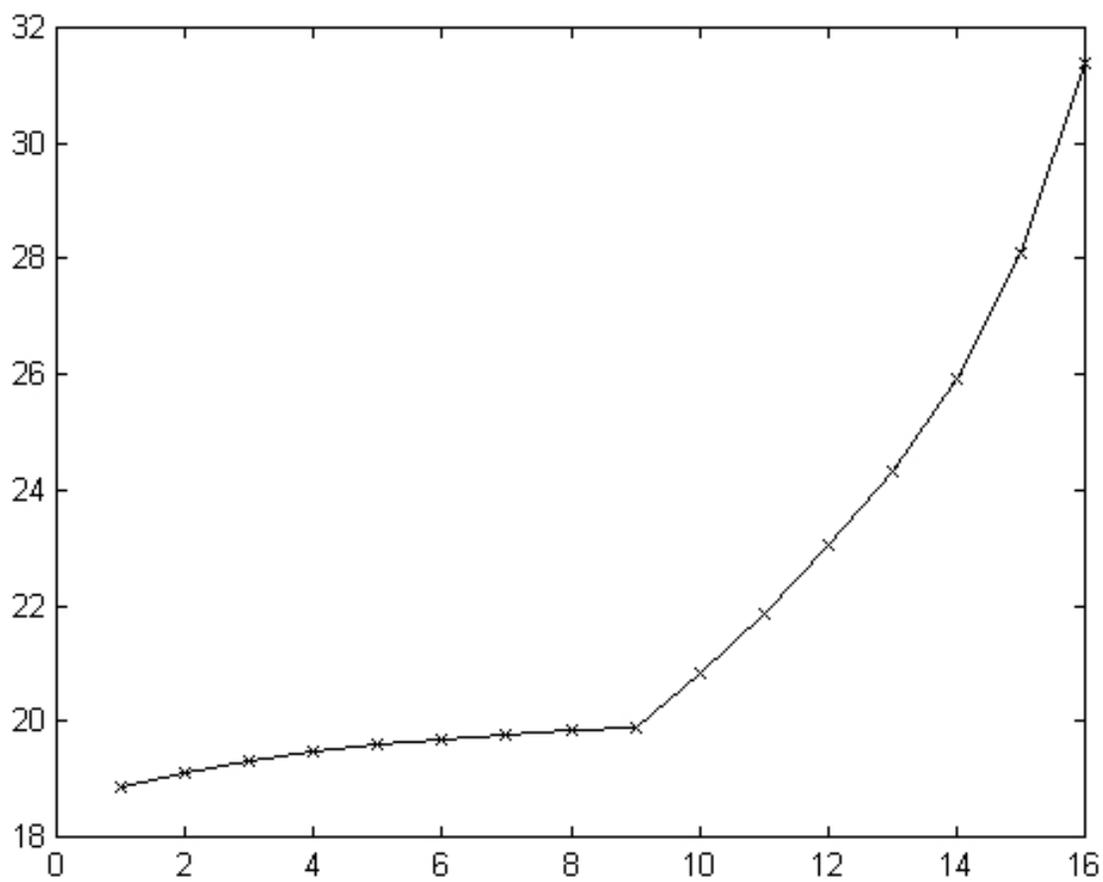


Figura 5.8: Gráfico SNR  $\times$  quantidade de coeficientes. Imagem "Baboo".



# Capítulo 6

## Conclusão

### 6.1 Considerações e Conclusão

Este trabalho apresentou um método de interpolação de imagens, baseado na transformada DSTr, aplicando extrapolação sobre seus coeficientes, com o propósito de tentar recuperar as componentes de alta frequência perdidas no processo de amostragem ou decimação, buscando-se alcançar super-resolução. Para que o problema fosse completamente entendido, apresentou-se um breve resumo da teoria de amostragem de sinais, enfatizando-se as características e os efeitos do processo de amostragem e mudança na taxa de amostragem que corresponde a interpolação e decimação, quando somente o sinal digital esta disponível. Também foi apresentado uma pesquisa ampla sobre interpolação de extrapolação. Apesar de não ser totalmente completa, a pesquisa mostra os principais métodos e as características intrínsecas a eles. Os métodos de interpolação de sinais no domínio da transformada nos dá uma noção bastante clara da perda de informação no processo e indica onde é possível buscar uma melhora. Já os métodos de extrapolação de sinais limitados em banda, mostra que é possível estimar a informação perdida a partir dos dados existentes, se alguma informação adicional sobre o comportamento do sinal estiver disponível.

O método proposto é uma extensão do método de interpolação por DSTr. A comparação entre os métodos mostra que é possível alcançar algum ganho em qualidade quando a estimativa dos coeficientes adicionados é uma extrapolação dos coeficientes existentes em comparação com a aplicação de valores nulos. No entanto o ganho é muito insignificante em comparação com a complexidade computacional.

A aplicação dos algoritmos de extrapolação é dificultada pela falta de informação sobre o sinal original e a dificuldade de se estimar essas informações. No caso de imagens, que são sinais bidimensionais, processadas em blocos, cada linha e coluna do bloco necessitaria de parâmetros individuais para sua extrapolação com maior precisão. Mesmo se fosse possível obter esses parâmetros, o cálculo para a imagem inteira se tornaria muito lento.

O ganho pequeno de qualidade pode ser explicado pela presença das seguintes características do problema:

- A imagem muito provavelmente possui sobreposição espectral, o que faz com que os coeficientes obtidos com a transformação da imagem decimada sejam diferentes dos coeficientes originais. Logo a extrapolação leva a um resultado errado;
- Mesmo que a imagem tenha sido filtrada antes de ser amostrada ou decimada, os filtros realizados na prática não são completamente perfeitos e são suscetíveis aos erros de processamento digital, deixando ocorrer sobreposição espectral, mesmo que pequeno;
- A imagem possui ruído de diferentes tipos, inclusive ruídos com energia fora da banda da imagem. Este ruído não é levado em consideração pelo algoritmo de extrapolação, que é bastante sensível a erros no sinal. O algoritmo de extrapolação utiliza inversão de matriz, que é um processo que pode ser instável;

Para se conseguir um resultado onde o ganho em qualidade seja mais significativo é

necessário ter mais informação sobre o sinal original, de preferência em cada bloco individualmente, para que um algoritmo de extrapolação mais eficiente e com parâmetros mais precisos possa ser utilizado. Possivelmente essa informação adicional pode ser conseguida com alguma relação ou modelo sobre as informações do segmento de coeficientes disponíveis, ou então analisando-se estatisticamente um conjunto de imagens. Esse estudo estatístico poderia ser realizado sobre um conjunto de imagens com características específicas.

Com base no estudo realizado neste trabalho, pode-se verificar que quando se deseja uma interpolação com baixo nível de detalhes, a extrapolação dos coeficientes não é interessante, pois o ganho em qualidade na imagem é imperceptível. No entanto, também pode-se verificar que é possível alcançar super-resolução de imagens através deste método. Apesar de não se ter conseguido um resultado totalmente satisfatório, foi possível levantar uma grande quantidade de dificuldades que necessitam ser contornadas, apontando para uma nova linha de pesquisa a ser seguida.

## 6.2 Trabalhos Futuros

Com base nos conceitos mostrados na idéia básica do trabalho podem ser sugeridos uma série de trabalhos futuros que consistem tanto na extensão ou aperfeiçoamento do método proposto, quanto no estudo da aplicação do método proposto em outros ambientes, conforme apontado a seguir.

### **Extensão e aperfeiçoamento:**

- Procurar um modelo que consiga relacionar os coeficientes do bloco disponível aos primeiros coeficientes originais do bloco, como se o processo de decimação tivesse sido feito, excluindo-se os últimos coeficientes;

- Procurar um modelo para se determinar, a partir do segmento conhecido, os parâmetros necessários aos algoritmos de extrapolação que apresentem melhores resultados;
- Determinar dados estatísticos, calculados sobre um número grande de imagens com características bem determinadas e específicas e estudar a aplicação dos valores obtidos no método ou utilizá-los como limites para o modelo a ser determinado como citado acima;
- Explorar informações sobre as características da representação do sinal digital utilizado (por exemplo, N pontos inteiros entre 0 e 255) e o sinal resultante da transformação utilizada;
- Levar em consideração o ruído contido no sinal a ser extrapolado, a sobreposição espectral ou que a filtragem não foi perfeita;
- Especializar um modelo onde o processo de aquisição da imagem, assim como seus parâmetros, sejam conhecidos e possam ser utilizados na determinação dos parâmetros do algoritmo de extrapolação ou na reformulação do método;
- Aplicar métodos de estabilização numérica na inversão da matriz utilizada no processo de extrapolação;
- Estudar a utilização de outras transformadas, incluindo DCT ou Wavelet;
- O modelo de extrapolação é baseado na utilização da Transformada de Fourier. Desenvolver e estudar um modelo baseado em outras transformadas;

**Aplicação em outros ambientes:**

- Imagens coloridas: Pesquisar correlações entre as componentes de cores RGB, HSV, etc. Explorar as limitações psico-visuais e estabelecer um forma de aplicação do método (ou um aperfeiçoamento);
- Imagens codificadas: Utilizar os erros de quantização para determinar limites para os parâmetros do algoritmo de extrapolação e utilizá-lo para determinar os coeficientes desconhecidos ou para corrigir com uma nova estimativa os valores dos coeficientes quantizados com menos bits, extrapolando-se os coeficientes quantizados com mais bits;
- A idéia de extrapolação poderia ser utilizada no padrão JPEG tanto para realizar a interpolação diretamente no domínio codificado como para restaurar a qualidade de imagens codificadas a baixas taxas. Por extensão, também é possível aplicar a idéia no padrão MPEG. O estudo da aplicação em um padrão é interessante no sentido de que vários parâmetros já estão bem definidos e possuem valores fixos;

### 6.3 Publicações

- 1) Yuzo Iano, Evaldo G. Pelaes, Patrick Biesdorf. Image Coding Using Decimation and Interpolation by Discrete Sine Transform with Axis Rotation. *IEICE* - Submitted.
- 2) Yuzo Iano, Patrick Biesdorf. Image Interpolation by Extrapolation of DSTr Coefficients. *IEICE* - Submitted.

**OBS:** Os arquivos de programas desenvolvidos encontram-se em CD-ROM anexo.



# Referências Bibliográficas

- [1] A. V. Oppenheim and R. W. Schaffer, *Discrete-Time Signal Processing*. Englewood Cliffs, New Jersey, EUA: Prentice-Hall, 1989.
- [2] A. K. Jain, *Fundamentals of Digital Image Processing*. Englewood Cliffs, New Jersey, EUA: Prentice-Hall, 1987.
- [3] E. G. Pelaes, *Transformada Seno Discreta com Rotação de Eixos Bidimensional (DSTr-2D): Aplicações na Codificação e Interpolação de Imagens para Redução de Efeito de Blocos*. PhD thesis, Universidade Estadual de Campinas, Campinas, SP, 1998.
- [4] M. L. Liou, “Spline fit made fast,” *IEEE Transactions on Computers*, vol. C-25, pp. 522–527, May 1976.
- [5] A. Aldroubi, M. Unser, and M. Eden, “B-spline signal processing: Part i - theory,” *IEEE Transactions on Signal Processing*, vol. 41, pp. 821–832, Feb. 1993.
- [6] A. Aldroubi, M. Unser, and M. Eden, “B-spline Signal Processing: Part II - Efficient Design and Applications,” *IEEE Transactions on Signal Processing*, vol. 41, pp. 834–848, Feb. 1993.
- [7] T. Blu and M. Unser, “Approximation error for quasi-interpolators and (multi) wavelet expansions,” *Applied and Computational Harmonic Analysis*, vol. 6, pp. 219–251, Mar. 1999.

- [8] S. K. Park and R. A. Schowengerdt, "Image sampling, reconstruction, and the effect of sample-scene phasing," *Applied Optics*, vol. 21, pp. 3142–3151, Sep. 1982.
- [9] G. Strang and G. Fix, "A Fourier analysis of the finite element variational method," *Constructive Aspect of Functional Analysis*, pp. 796–830, 1971.
- [10] A. Schaum, "Theory and design of local interpolators," *CVGIP: Graphical Models and Image Processing*, vol. 55, pp. 464–481, Nov. 1993.
- [11] J. L. Ostuni, A. K. S. Santha, V. S. Mattay, D. R. Weinberger, R. L. Levin, and J. A. Frank, "Analysis of interpolation effects in the reslicing of functional MR images," *Journal of Computer Assisted Tomography*, vol. 21, pp. 803–810, 1997.
- [12] R. W. Parrot, M. R. Styzt, P. Amburn, and D. Robinson, "Towards statistically optimal interpolation for 3-d medical imaging," *IEEE Engineering in Medicine and Biology*, vol. 12, pp. 49–59, Oct. 1993.
- [13] A. Aldroubi, , and M. Unser, "Sampling procedures in function spaces and asymptotic equivalence with shannon's sampling theorem," *Numerical Function Analysis and Optimization*, vol. 15, pp. 1–21, 1993.
- [14] A. Aldroubi, M. Unser, , and M. Eden, "Cardinal spline filters: Stability and convergence to the ideal sinc interpolator," *Signal Processing*, vol. 28, pp. 127–138, 1992.
- [15] T. Blu, P. Thévenaz, and M. Unser, "Minimum suport interpolators with optimum approximation properties," *IEEE International Conference on Image Processing*, Oct. 1998.
- [16] R. G. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP, pp. 1153–1160, 1981.
- [17] J. S. Lim, *Two-Dimensional Signal and Image Processing*. Englewood Cliffs, New Jersey, EUA: Prentice-Hall, 1990.

- [18] K. R. Rao and P. Yip, *Discrete Cosine Transform: Algorithms, Advantages, Applications*. Boston, EUA: Academic Press Inc., 1990.
- [19] D. Fraser, “Interpolation by the FFT revised - an experimental investigation,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 17, pp. 665–675, May 1989.
- [20] A. M. L. Araújo, A. K. Jr., and V. G. Macedo, “Interpolação usando FFT associada ao processo de rotação,” *Anais do XVII CNMAC*, vol. 2, pp. 628–642, Sep. 1994.
- [21] A. M. L. Araújo and A. M. F. Jr., “Interpolação pela transformada discreta do seno com pré e pós rotação,” *XIV Simpósio Brasileiro de Telecomunicações*, Sep. 1996.
- [22] A. M. L. Araújo, E. G. Pelaes, and Y. Iano, “Interpolação de imagens por transformadas com eliminação de extrapolações e reduzido efeito de bloqueamento,” *Anais do XV Simpósio Brasileiro de Telecomunicações*, Sep. 1997.
- [23] M. Hamidi and J. Pearl, “Comparison of cosine and Fourier transforms of Markov-I signals,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 24, pp. 428–429, Oct. 1976.
- [24] A. Papoulis, “A new algorithm in spectral analysis and band-limited extrapolation,” *IEEE Transactions on Circuits and Systems*, vol. CAS-22, pp. 735–742, Sep. 1975.
- [25] A. K. Jain and S. Ranganath, “Extrapolation algorithms for discrete signals with application in spectral estimation,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-29, pp. 830–844, Aug. 1981.
- [26] E. J. Diethorn and D. C. Munson, “A linear, time-varying system framework for noniterative discrete-time band-limited signal extrapolation,” *IEEE Trans. on Signal Processing*, vol. 39, pp. 55–68, Jan. 1991.
- [27] J. L. C. Sanz and T. S. Huang, “Discrete and continuous band-limited signal extrapolation,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-31, pp. 1276–1285, Oct. 1983.

- [28] S. D. Cabrera and T. W. Parks, "Extrapolation and spectral estimation with iterative weighted norm modification," *IEEE Trans. on Signal Processing*, vol. 39, pp. 842–851, Apr. 1991.
- [29] D. P. Kolba and T. W. Parks, "Extrapolation and spectral estimation for band-limited signals," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing.*, pp. 372–374, 1978.
- [30] D. P. Kolba and T. W. Parks, "Optimal estimation for band-limited signals including time domain," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP, pp. 113–122, Feb. 1983.
- [31] L. C. Potter and K. S. Arun, "Energy concentration in band-limited extrapolation," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 1027–1041, Jul. 1989.
- [32] S. Dharanipragada and K. S. Arun, "Bandlimited extrapolation using time-bandwidth dimension," *IEEE Trans. on Signal Processing*, vol. 45, pp. 2951–2966, Dec. 1997.

# Apêndice A

## Seqüências Esferoidais Estendidas

### A.1 Funções Esferoidais Estendidas

As Funções Esferoidais Estendidas (*Prolate Spheroidal Functions*) são definidas como as autofunções da equação:

$$\int_{-T}^T \phi_k(\tau) \frac{\sin \sigma(t - \tau)}{\pi(t - \tau)} d\tau = \lambda_k \phi_k(t) \quad (\text{A.1.1})$$

Propriedades:

a)  $\phi_k(t)$  são limitadas em banda em  $\sigma$ , isto é:

$$\phi_k(t) * \frac{\sin \sigma t}{\pi t} = \phi_k(t) \quad (\text{A.1.2})$$

b)  $\phi_k(t)$  são ortonormais em  $(-\infty, \infty)$  e ortogonais em  $(-T, T)$ , isto é:

$$\int_{-\infty}^{\infty} \phi_k(t) \phi_l(t) dt = \begin{cases} 1 & k = l \\ 0 & k \neq l \end{cases} \quad (\text{A.1.3})$$

$$\int_{-T}^T \phi_k(t) \phi_l(t) dt = \begin{cases} \lambda_k & k = l \\ 0 & k \neq l \end{cases} \quad (\text{A.1.4})$$

c) Os auto valores  $\lambda_k$  são tais que  $1 > \lambda_0 > \lambda_1 > \dots > \lambda_k > \dots > 0$  e

$$\lim_{k \rightarrow \infty} \lambda_k = 0 \quad (\text{A.1.5})$$

d) Qualquer função  $f(t)$  limitada em banda em  $\sigma$  pode ser expressa como:

$$\begin{aligned} f(t) &= \sum_{k=0}^{\infty} a_k \phi_k(t) \\ a_k &= \int_{-\infty}^{\infty} f(t) \phi_k(t) dt \end{aligned} \quad (\text{A.1.6})$$

e

$$\|f(t)\|^2 = \sum_{k=0}^{\infty} a_k^2 \quad (\text{A.1.7})$$

e) Qualquer função  $g(t)$  com

$$\|g(t)\|_T^2 = \int_{-T}^T |g(t)|^2 dt < \infty \quad (\text{A.1.8})$$

pode ser expressa no intervalo  $(-T, T)$  como:

$$\begin{aligned} g(t) &= \sum_{k=0}^{\infty} b_k \phi_k(t) \quad |t| < T \\ b_k &= \frac{1}{\lambda_k} \int_{-T}^T g(t) \phi_k(t) dt \end{aligned} \quad (\text{A.1.9})$$

e

$$\|g(t)\|_T^2 = \sum_{k=0}^{\infty} \lambda_k b_k^2 \quad (\text{A.1.10})$$

## A.2 Seqüências Esferoidais Estendidas Discretas

As seqüências esferoidais estendidas discretas (*Discrete Prolate Spheroidal Sequences* - DPSS) são os autovetores  $\phi_k[m]$  da equação:

$$\sum_{n=-M}^M \frac{\sin \sigma(m-n)}{\pi(m-n)} \phi_k[n] = \lambda_k \phi_k[m] \quad k = 0, \dots, 2M \quad (\text{A.2.1})$$

Propriedades:

a)  $\phi_k[m]$  são limitadas em banda a  $\sigma$ , isto é:

$$\phi_k[m] * \frac{\sin \sigma m}{\pi m} = \phi_k(m) \quad (\text{A.2.2})$$

b)  $\phi_k[m]$  são duplamente ortogonais, isto é:

$$\sum_{-\infty}^{\infty} \phi_k[m] \phi_l[m] = \begin{cases} 1 & k = l \\ 0 & k \neq l \end{cases} \quad (\text{A.2.3})$$

$$\sum_{-M}^M \phi_k[m] \phi_l[m] = \begin{cases} \lambda_k & k = l \\ 0 & k \neq l \end{cases} \quad (\text{A.2.4})$$

c) Os autovalores  $\lambda_k$  são tais que  $1 > \lambda_0 > \lambda_1 > \dots > \lambda_{2M} > 0$

d) Existem seqüências  $\phi_k[m]$ ,  $k = 2M + 1, \dots, 0$  limitadas em banda a  $\sigma$ , tais que:

$$\sum_{m=-\infty}^{\infty} \phi_k[m] \phi_l[m] = \begin{cases} 1 & k = l \\ 0 & k \neq l \end{cases} \quad k, l = 0, 1, 2, \dots \quad (\text{A.2.5})$$

e qualquer seqüência  $f[m]$  limitada em banda a  $\sigma$  pode ser expressa como:

$$\begin{aligned} f[m] &= \sum_{k=0}^{\infty} a_k \phi_k[m] \\ a_k &= \sum_{m=-\infty}^{\infty} f[m] \phi_k[m] \end{aligned} \quad (\text{A.2.6})$$

e

$$\|f[m]\|^2 = \sum_{k=0}^{\infty} a_k^2 \quad (\text{A.2.7})$$

e) Qualquer vetor  $(g[-M], \dots, g[M])$  pode ser expresso como:

$$\begin{aligned} g[m] &= \sum_{k=0}^{2M} b_k \phi_k[m] \quad |m| \leq M \\ b_k &= \frac{1}{\lambda_k} \sum_{m=-M}^M g[m] \phi_k[m] \quad k = 0, \dots, 2M \end{aligned} \quad (\text{A.2.8})$$

e

$$\|g[m]\|_{-M, M}^2 = \sum_{k=0}^{2M} \lambda_k b_k^2 \quad (\text{A.2.9})$$

fica óbvio que se  $g[m] = f[m]$  para  $|m| \leq M$ ,  $b_k = a_k$  para  $k = 0, 1, \dots, 2M$ .

### A.3 Seq. Esferoidais Estendidas Discretas Periódicas

As seqüências esferoidais estendidas discretas periódicas (*Periodic Discrete Prolate Spheroidal Sequences* - P-DPSS) são os autovetores  $\phi_i[m]$  da equação:

$$\sum_{n=-M}^M \frac{\sin\left(\frac{\pi}{N}(2K+1)(m-n)\right)}{N \sin\left(\frac{\pi}{N}(m-n)\right)} \phi_i[n] = \lambda_i \phi_i[m] \quad i = 0, \dots, 2M \quad (\text{A.3.1})$$

onde  $2K+1 < N, 2M+1 < N$ .

Propriedades:

a) Os autovalores  $\lambda_i$  são tais que  $1 \geq \lambda_0 \geq \lambda_1 \geq \dots \geq \lambda_{2M} > 0$  se  $M \leq K$  ou  $1 \geq \lambda_0 \geq \lambda_1 \geq \dots \geq \lambda_{2K} > \lambda_{2K+1} = \dots = \lambda_{2M} = 0$  se  $M > K$

b)  $\phi_i[m]$  são duplamente ortogonais, isto é:

$$\sum_{m=l_0}^{l_0+N-1} \phi_i[m] \phi_j[m] = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases} \quad (\text{A.3.2})$$

$$\sum_{m=-M}^M \phi_i[m] \phi_j[m] = \begin{cases} \lambda_i & i = j \\ 0 & i \neq j \end{cases} \quad (\text{A.3.3})$$

onde  $i, j = 0, \dots, \min(2K, 2M)$ .

c)  $\phi_i[m], i = 0, \dots, \min(2K, 2M)$ , são limitadas em banda a  $K$ , isto é:

$$\sum_{n=l_0}^{l_0+N-1} \frac{\sin\left(\frac{\pi}{N}(2K+1)(m-n)\right)}{N \sin\left(\frac{\pi}{N}(m-n)\right)} \phi_i[n] = \phi_i[m] \quad (\text{A.3.4})$$

d) Quando  $M \leq K$ , existem seqüências  $\phi_i[m], i = 2M+1, \dots, 2K$  limitadas em banda a  $K$  e periódicas em  $N$ , tais que:

$$\sum_{m=l_0}^{l_0+N-1} \phi_i[m] \phi_j[m] = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases} \quad (\text{A.3.5})$$

e qualquer seqüência  $f[m]$  limitada em banda a  $K$  e periódica em  $N$  pode ser expressa como:

$$\begin{aligned} f[m] &= \sum_{i=0}^{2K} a_i \phi_i[m] \\ a_i &= \sum_{m=l_0}^{l_0+N-1} f[m] \phi_i[m] \end{aligned} \quad (\text{A.3.6})$$

e

$$\|f[m]\|_{l_0, l_0+N-1}^2 = \sum_{i=0}^{2K} a_i^2 \quad (\text{A.3.7})$$

e) Quando  $M > K$ , qualquer seqüência  $f[m]$  limitada em banda a  $K$  e periódica em  $N$  pode ser expressa como:

$$\begin{aligned} f[m] &= \sum_{i=0}^{2K} a_i \phi_i[m] \\ a_i &= \sum_{m=l_0}^{l_0+N-1} f[m] \phi_i[m] \end{aligned} \quad (\text{A.3.8})$$

e

$$\|f[m]\|_{l_0, l_0+N-1}^2 = \sum_{i=0}^{2K} a_i^2 \quad (\text{A.3.9})$$

f) Quando  $M \leq K$ , qualquer vetor  $(g[-M], \dots, g[M])$  pode ser expresso como:

$$\begin{aligned} g[m] &= \sum_{i=0}^{2M} b_i \phi_i[m] \quad |m| \leq M \\ b_i &= \frac{1}{\lambda_i} \sum_{m=-M}^M g[m] \phi_i[m] \end{aligned} \quad (\text{A.3.10})$$

e

$$\|g[m]\|_{-M, M}^2 = \sum_{i=0}^{2M} \lambda_i b_i^2 \quad (\text{A.3.11})$$

g) Quando  $M > K$ , qualquer vetor  $(g[-M], \dots, g[M])$  pode ser expresso como:

$$g[m] = \sum_{i=0}^{2K} b_i \phi_i[m] + \tilde{g}[m] \quad |m| \leq M$$

$$\sum_{m=-M}^M \tilde{g}[m] \phi_i[m] = 0 \quad i = 0, \dots, 2K \quad (\text{A.3.12})$$

$$b_i = \frac{1}{\lambda_i} \sum_{m=-M}^M g[m] \phi_i[m] \quad (\text{A.3.13})$$

e

$$\|g[m]\|_{-M,M}^2 = \sum_{i=0}^{2K} \lambda_i b_i^2 + \|\tilde{g}[m]\|_{-M,M}^2 \quad (\text{A.3.14})$$

## A.4 Expansão de Karhunen Loève

Considere um processo estocástico  $\mathbf{y}(t)$  com autocorrelação

$$R(t_1, t_2) = E\{\mathbf{y}(t_1) \mathbf{y}^*(t_2)\}. \quad (\text{A.4.1})$$

Com

$$\Gamma(u, v) = E\{\mathbf{Y}(u) \mathbf{Y}^*(v)\} \quad (\text{A.4.2})$$

a autocorrelação da transformada de Fourier  $\mathbf{Y}(\omega)$  de  $\mathbf{y}(t)$ , é possível mostrar que

$$\Gamma(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} R(t_1, t_2) \exp(-j(ut_1 - vt_2)) dt_1 dt_2 \quad (\text{A.4.3})$$

Com  $\Gamma(u, v)$  dado acima, forma-se a equação

$$\int_{-\sigma}^{\sigma} \Gamma(u, v) \psi(v) dv = \gamma \psi(u) \quad (\text{A.4.4})$$

e denota-se por  $\psi_k(u)$  e  $\gamma_k$  suas autofunções e autovalores respectivamente. Pode ser mostrado que na expansão:

$$\mathbf{Y}(\omega) = \sum_0^{\infty} \mathbf{z}_k \psi_k(\omega), \quad |\omega| \leq \sigma \quad (\text{A.4.5})$$

as variáveis aleatórias  $z_k$  são ortogonais e

$$E\{z_i z_k^*\} = \begin{cases} \gamma_k, & i = k \\ 0, & i \neq k \end{cases} \quad (\text{A.4.6})$$

Ruído branco não estacionário: Diz-se que o processo  $\mathbf{y}(t)$  é um ruído branco se:

$$R(t_1, t_2) = q(t_1)\delta(t_1 - t_2) \quad (\text{A.4.7})$$

onde  $q(t) \geq 0$  é sua intensidade média. Com  $Q(\omega)$  a transformada de Fourier de  $q(t)$ , segue de A.4.3 que:

$$\Gamma(u, v) = Q(u - v). \quad (\text{A.4.8})$$

Se  $\mathbf{y}(t) = \mathbf{n}(t)p_T(t)$  e  $\mathbf{n}(t)$  é um ruído branco estacionário com autocorrelação  $S\delta(t_1 - t_2)$ , então:

$$R(t_1, t_2) = Sp_T(t_1)\delta(t_1 - t_2). \quad (\text{A.4.9})$$

Neste caso,

$$q(t) = Sp_T(t) \quad Q(\omega) = 2S \sin T\omega/\omega. \quad (\text{A.4.10})$$

Da equação acima e de A.4.8 segue que a equação integral A.4.4 toma a forma:

$$2S \int_{-\sigma}^{\sigma} \frac{\sin T(u - v)}{u - v} \psi(v) dv = \gamma\psi(u). \quad (\text{A.4.11})$$

Como observa-se de A.1.1, suas autofunções  $\psi(u)$  se igualam às funções esferoidais estendidas com escala apropriada.