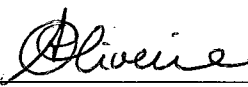


APLICAÇÕES DE B-WAVELETS, SEMI-ORTOGONAIS SOBRE  
INTERVALOS, EM MODELAGEM GEOMÉTRICA, FILTRAGEM E  
RECONSTRUÇÃO DE IMAGENS EM MULTIRESOLUÇÃO

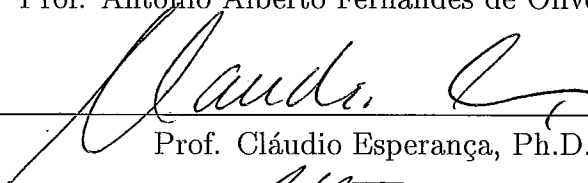
José Geraldo Franco Méxas

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS  
PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA  
UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS  
REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE  
DOUTOR EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E  
COMPUTAÇÃO.

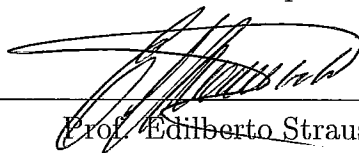
Aprovada por:



Prof. Antônio Alberto Fernandes de Oliveira, D.Sc.



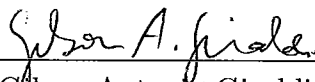
Prof. Cláudio Esperança, Ph.D.



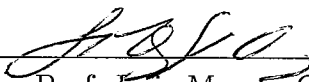
Prof. Edilberto Strauss, Ph.D.



Prof. Aura Conci, D.Sc.



Prof. Gilson Antonio Giraldi, D.Sc.



Prof. Luiz Marcos Garcia Gonçalves, D.Sc.

RIO DE JANEIRO, RJ - BRASIL  
AGOSTO DE 2001

MÉXAS, JOSÉ GERALDO FRANCO

Aplicações de B-Wavelets, semi-ortogonais sobre intervalos, em modelagem geométrica, filtragem e reconstrução de imagens em multiresolução [Rio de Janeiro] 2001

X, 91 pp., 29.7 cm, (COPPE/UFRJ, D.Sc., Engenharia de Sistemas e Computação, 2001)

Tese – Universidade Federal do Rio de Janeiro, COPPE

1 – Processamento, Reconstrução de Imagens, Multiresolução

2 – B-Splines, B-Wavelets, Modelagem Geométrica, Transformada de Radon

I. COPPE/UFRJ II. Título (série)

## Dedicatória

À meus pais Geraldo e Carmen, à minha esposa Mirian e à meus filhos  
Vivian e Rodrigo.

*”Quem quiser trabalhar na Grande Obra, deve visitar sua alma, penetrar no  
maís fundo de seu ser e aí executar um trabalho oculto e misterioso. Como  
o grão que deve ser sepultado no seio da terra, assim também quem houve o  
chamado de Deus deve, retificando-se, obter a sublime transformação e  
fazer do carvão um diamante e fazer do chumbo um puro ouro. E terá  
então encontrado a Pedra Filosofal, que ocultava em si próprio.”*

*Robert Amadou*

## Agradecimentos

Aos professores Aura Conci, Ronaldo Marinho Persiano, Antônio Alberto Fernandes de Oliveira e Cláudio Esperança agradeço a atenção e a iluminação que proporcionaram a tomada da consciência da importância da computação gráfica na visualização de estruturas matemáticas. Aos colegas do LCG, em especial ao Gilson, Paulo Sérgio, Apolinário e a Sra. Gercina Armando da Silva meu muito obrigado pelo apoio e incentivo.

*”A única iniciação que recomendo, a busca mais ardorosa de minha alma, é aquela pela qual podemos penetrar o coração de Deus e induzir este coração divino a penetrar o nosso. Assim se tornará perfeito o matrimônio indissolúvel que fará de nós o irmão, o cônjuge do nosso Divino Salvador ”*

*Louis-Claude de Saint-Martin*

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

APLICAÇÕES DE B-WAVELETS, SEMI-ORTOGONAIS SOBRE  
INTERVALOS, EM MODELAGEM GEOMÉTRICA, FILTRAGEM E  
RECONSTRUÇÃO DE IMAGENS EM MULTIRESOLUÇÃO

José Geraldo Franco Méxas

Agosto /2001

Orientadores: Antônio Alberto Fernandes de Oliveira  
Cláudio Esperança

Programa: Engenharia de Sistemas e Computação

Este trabalho é sobre a aplicação de Wavelets geradas por B-Splines, semi-ortogonais e definidas sobre intervalos limitados, em modelagem geométrica variacional na minimização da energia de curvatura de splines, filtragem tensorial de imagens em multi-resolução e na reconstrução em multiresolução de imagens em associação com a Transformada Algébrica de Radon.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

APPLICATIONS OF B-WAVELETS, SEMIORTHOGONAL ON  
INTERVALS, IN GEOMETRIC MODELING, FILTRATION AND  
IMAGES RECONSTRUCTION IN MULTIREOLUTION

José Geraldo Franco Méxas

August /2001

Advisors: Antônio Alberto Fernandes de Oliveira  
Cláudio Esperança

Department: Computing and Systems Engineering

This work is about the Wavelets application, generated by B-spline, semi-orthogonal and defined on limited intervals, in variational geometric modeling in minimization of the splines curvature energy, tensor filtration of images in multiresolution and in the reconstruction in multiresolution of images in association with Algebraic Radon Transform.

# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>1</b>
<b>2</b>	<b>B-Splines sobre intervalos limitados</b>	<b>5</b>
2.1	Funções B-Splines . . . . .	6
2.2	Spline na forma B-Spline . . . . .	9
2.3	Refinamento de nós . . . . .	12
2.4	Refinamento diádico e a equação associada . . . . .	13
<b>3</b>	<b>B-Wavelets semi-ortogonais sobre intervalos limitados</b>	<b>17</b>
3.1	B-Wavelets . . . . .	17
3.2	Mudança de bases entre resoluções distintas . . . . .	19
3.3	Transformada Wavelet Discreta . . . . .	21
3.4	A transformada B-Wavelet, semi-ortogonal definida sobre intervalo limitado . . . . .	22
<b>4</b>	<b>Aplicação das B-Wavelets em modelagem geométrica variacional</b>	<b>29</b>
4.1	Apresentação do problema de modelagem geométrica variacional	30
4.2	Resolvendo o problema de otimização na base das B-Wavelets	32
4.3	Aplicação . . . . .	35
<b>5</b>	<b>Aplicação das B-Wavelets tensoriais em filtragem de imagens em multiresolução</b>	<b>45</b>
5.1	Matriz associada ao operador linear na base das wavelets . . .	45
5.1.1	Representação matricial de um operador linear em níveis distintos de resolução . . . . .	46
5.1.2	Representação tensorial de um operador linear em níveis distintos de resolução . . . . .	48
5.2	Transformada discreta B-Wavelet bidimensional e tensorial . .	51
5.3	Aplicação . . . . .	54

<b>6</b>	<b>Aplicação das B-Wavelets em reconstrução de imagens</b>	<b>64</b>
6.1	Transformada Discreta de Fourier . . . . .	65
6.2	Transformada de Radon . . . . .	66
6.2.1	Transformada contínua . . . . .	66
6.2.2	Transformada discreta . . . . .	67
6.2.3	Transformada algébrica . . . . .	68
6.3	Aplicação . . . . .	71
6.3.1	A Transformada algébrica de Radon com B-Wavelets (B-RADONLET) . . . . .	71
<b>7</b>	<b>Conclusão</b>	<b>73</b>
<b>A</b>	<b>Condicionamento de matrizes e o algoritmo do gradiente conjugado</b>	<b>74</b>
A.1	Normas de vetores e matrizes . . . . .	75
A.1.1	Propriedades . . . . .	76
A.1.2	Exemplos . . . . .	76
A.2	Condicionamento de matrizes . . . . .	76
A.3	Algoritmo do Gradiente Conjugado . . . . .	79
A.3.1	Definições . . . . .	79
A.3.2	O algoritmo do gradiente conjugado . . . . .	80
A.3.3	O algoritmo do gradiente conjugado e o método dos mínimos quadrados . . . . .	83



# Lista de Figuras

1.1	Pierre Etienne Bézier - (01/09/1910 - 25/11/1999). . . . .	2
2.1	B-Splines de grau 0, 1, 2 e 3 no nível três de resolução . . . . .	8
2.2	Curva de Bézier do segundo grau, renderizada no latex2. . . . .	9
2.3	Desenho de Pierre E. Bézier feito com a sua curva. . . . .	10
2.4	O algoritmo de inserção de nós de W. Boem . . . . .	16
3.1	Todas as B-Splines (8) e B-Wavelets (8) de grau 0, (HAAR), no nível três de resolução, sobre [0,1] . . . . .	22
3.2	Todas as B-Splines (9) e B-Wavelets (8) de grau 1, (Linear), no nível três de resolução, sobre [0,1] . . . . .	23
3.3	Todas as B-Splines (10) e B-Wavelets (8) de grau 2, (Quádricas), no nível três de resolução, sobre [0,1] . . . . .	23
3.4	Todas as B-Splines (11) e B-Wavelets (8) de grau 3, (Cúbicas), no nível três de resolução, sobre [0,1] . . . . .	24
3.5	Transformada B-Wavelet sobre as coordenadas dos 35 pontos de controle da letra r . . . . .	25
3.6	A letra r nos níveis 5 e 4 de resolução . . . . .	26
3.7	A letra r nos níveis 3 e 2 de resolução . . . . .	26
3.8	A letra r nos níveis 1 e 0 de resolução . . . . .	27
3.9	Reconstrução da letra r . . . . .	28
4.1	Base com $16+1=17$ B-Splines cúbicas sobre o intervalo [0,16] . . . . .	37
4.2	Filtro de análise. Convolução com decimação. . . . .	38
4.3	Filtro de síntese. Interpolação com convolução. . . . .	39
4.4	Nível 6 de resolução com B-Splines (curva superior) e o nível 0 com B-Wavelets (curva inferior). Iterações 0, 4 e 16. . . . .	40
4.5	Nível 6 de resolução com B-Splines (curva superior) e o nível 0 com B-Wavelets (curva inferior). Iterações 64, 256 e 1024 . . . . .	40
4.6	Posição inicial da curva . . . . .	41
4.7	Interpolação dos três pontos, na base B-Spline cúbica no nível 6, com 464 iterações . . . . .	42

4.8	Interpolação dos três pontos, na base com B-Wavelets cúbicas no nível 3, com 35 iterações . . . . .	43
4.9	Interpolação dos três pontos, na base com B-Wavelets cúbicas no nível 0, com 25 iterações . . . . .	44
5.1	Imagem original de Lena com 256 linhas e 128 colunas . . . . .	55
5.2	Decomposição LowLow da Imagem da Lena, usando o produto tensorial dos filtros de Haar. . . . .	56
5.3	Decomposicao HighLow da Imagem da Lena usando o produto tensorial dos filtros de Haar. . . . .	57
5.4	Decomposicao LowHigh da Imagem da Lena usando o produto tensorial dos filtros de Haar. . . . .	58
5.5	Decomposicao HighHigh da Imagem da Lena usando o produto tensorial dos filtros de Haar. . . . .	59
5.6	A transformada wavelet direta, bidimensional e tensorial . . . . .	60
5.7	Filtragem da imagem de Lena com Haar, do nível (7,8) para o nível (5,6) . . . . .	61
5.8	Filtragem da imagem de Radon com os filtros Linear e de Haar, do nível (8,8) para o nível (7,7) . . . . .	62
6.1	Transformada de Radon contínua . . . . .	67
6.2	Transformada de Radon discreta . . . . .	68
6.3	Transformada inversa algébrica de Radon usando o algoritmo do gradiente conjugado dos mínimos quadrados . . . . .	70
6.4	Transformada inversa generalizada algébrica de Radon, com wavelets, usando o algoritmo GCMQ. . . . .	72
A.1	O algoritmo do gradiente conjugado . . . . .	80
A.2	O algoritmo do gradiente conjugado em n iterações . . . . .	81

# Capítulo 1

## Introdução

Os trabalhos em desenho mecânico e controle numérico na indústria automobilística, de Paul de Casteljaou,[1], na Citroen e de Pierre Etienne Bézier, imagem 1.1 e referências [2, 3, 4], na Renault, levaram a criação das “curvas de Bézier”.

Estas curvas são caracterizadas numericamente pelas coordenadas de um conjunto de pontos de controle e que por um processo de subdivisão iterativo, denominado “Algoritmo de Casteljaou”, possibilita a criação da curva no nível desejado de aproximação.

Posteriormente a descrição analítica das curvas polinomiais de Bézier foram encontradas nos estudos de Bernstein,[5], onde verificou-se que as curvas de Bézier podem ser descritas como combinação linear da base de Bernstein no espaço dos polinômios, sendo os pontos de controle associados aos coeficientes desta combinação.

Outro problema importante com aplicação em modelagem geométrica e que ocorre na teoria da elasticidade, é a modelagem matemática de uma barra flexível submetida a esforço, cuja forma, deduzida pelas correspondentes equações [6, 7], é de uma curva, formada por polinômios cúbicos por partes, denominada spline cúbica.

Estas curvas foram introduzidas em computação gráfica por Riesenfeld [8] cujo estudo matemático foi feito por Carl de Boor [9, 10], com a introdução de uma “base de B(Basic)-Splines”, onde os pontos de controle eram associados aos coeficientes da base, em relação a qual a spline era descrita como combinação linear.

A modelagem analítica se estende naturalmente através do produto tensorial de funções polinomiais definidas por pedaços para a descrição de superfícies que modelam chapas flexíveis [11].

Em modelagem geométrica ocorre, muitas vezes, o problema da representação de deformações abruptas na geometria local de curvas e superfícies.

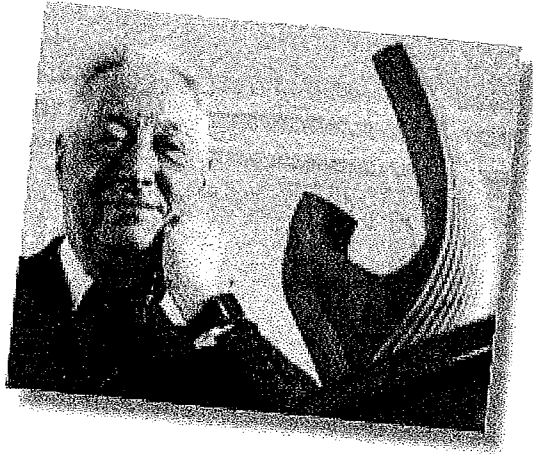


Figura 1.1: Pierre Etienne Bézier - (01/09/1910 - 25/11/1999).

Nestes casos a descrição dos detalhes da região de interesse pode ser feita com a introdução das splines no formato de “pequenas ondas” denominadas “ondelettes” ou “wavelets”, [12, 13], que se adaptam a qualquer ponto da superfície por “translação” e com relação ao nível de resolução dos detalhes da geometria, por mudança de “escala”.

A formulação de modelos com wavelets leva a descrição de uma curva ou superfície em vários níveis de resolução onde as informações globais (frequências baixas) são descritas pela base das B-Splines e as informações dos detalhes entre diferentes níveis de resolução são descritas pelas B-Wavelets. Neste caso as mudanças de bases entre os vários níveis se traduz na filtragem por filtros passa-baixa associados as B-Splines e passa-alta associados as B-Wavelets.

As splines e wavelets foram usadas com sucesso no estudo das imagens, onde podemos citar os trabalhos de Michael Unser et al, [14, 15, 16].

No início do século XX, em 1917, o austríaco Johann Radon descreveu uma transformação, [17, 18] em que uma função definida em uma região limitada no plano era caracterizada pela sua integral de linha ao longo de várias direções sobre esta região. Este trabalho de Radon foi introduzido no estudo experimental da aquisição das imagens médicas em tomografia computadorizada, com as pesquisas de Allan. M. Cormack [19] e G. N. Hounsfield [20] (ganhadores do prémio nobel em 1973). Nestes experimentos uma região plana de interesse, era varrida por um feixe de raios X, cuja atenuação em cada direção pré fixada, era dada por uma integral de linha. Este problema era inicialmente resolvido algebricamente e posteriormente foi identificado

com a Transformada Inversa de Radon.

Existe uma relação muito próxima entre a Transformada de Radon e as wavelets quando se considera a iluminação ou visualização da região de interesse por pulsos de raio X, formado por pequenas ondas ou wavelets [21]. Neste caso, do ponto de vista das B-splines, o problema de interesse consiste em reconstruir a superfície que representa a imagem conhecendo a sua Transformada de Radon.

Com relação as wavelets, as *contribuições* nesta tese estão na formulação matricial das B-Wavelets semi-ortogonais sobre intervalos limitados, construídas a partir das B-Splines, nos seguintes contextos:

I- Em modelagem geométrica variacional, relativa a interpolação de pontos do plano por uma curva, onde a representação da curva na base das B-Wavelets, é descrita a partir de filtros *matriciais* de B-Splines *adaptados nos bordos do intervalo limitado*.

II- Na apresentação de uma definição *mais geral* da Transformada Wavelet Bidimensional e Tensorial com a introdução do produto tensorial de duas B-Wavelets de *graus distintos* e do *correspondente conceito generalizado de nível de resolução, designado pelo par  $(j, i)$* .

III- Na filtragem de imagens, *em multiresolução  $(j, i)$* , onde  $j$  é o nível de resolução da largura e  $i$  é o nível de resolução da altura da imagem, com filtros de wavelets, construídos com B-Splines de *graus diferentes ao longo das linhas e colunas da imagem*. A escolha das B-Splines segundo o grau, pode ser feita, por exemplo, em função da distribuição dos pixels da imagem, em termos de continuidade, ao longo das linhas e colunas.

IV- Na apresentação de uma nova transformada algébrica, a Transformada Algébrica B-Radonlet (Radon+B-Wavelet), definida a partir da formulação matricial da Transformada Algébrica de Radon unificada com a Transformada Matricial das B-Wavelets, no nível  $(j, i)$  de resolução, onde  $i$  é o nível de resolução do senograma e  $j$  é o nível de resolução da imagem reconstruída.

V- Na aplicação da B-Radonlet Algébrica na reconstrução de imagens médicas em multiresolução, onde podemos considerar, *uma B-Wavelet de melhor grau que se adapte, por exemplo, em termos de continuidade, ao senograma unidimensional de nível  $i$  de resolução e outra B-Wavelet de grau distinto, que melhor se adapte a imagem unidimensional reconstruída, no nível  $j$  de resolução*.

No que se segue, com o intuito de introduzir os *conceitos e a notação* usados no desenvolvimento do trabalho, faremos no capítulo 2 a apresentação das B-Splines sobre intervalos limitados adaptadas nos bordos, e em seguida, no capítulo 3, a apresentação das B-Wavelets semi-ortogonais sobre intervalos limitados.

As *contribuições* citadas anteriormente serão apresentadas da seguinte

forma: o item I será descrito no capítulo 4, os itens II e III serão apresentados no capítulo 5 e os itens IV e V comentados no capítulo 6.

No apêndice são revisados os conceitos de condicionamento de matrizes e o algoritmo do gradiente conjugado para a solução de sistemas lineares, incluindo o caso de sistemas não inversíveis, cuja solução aproximada é obtida no contexto da inversa generalizada.

## Capítulo 2

# B-Splines sobre intervalos limitados

Faremos neste capítulo a revisão de splines e B-Splines, considerando as curvas de Bézier como caso particular, tal como as referências [22, 10]. Usaremos a notação matricial como estabelecida em [23] e de resultados tratados em detalhes em [24, 25]. O resultado importante é a construção de uma base de B-Splines mais refinada (mudança de escala) a partir do algoritmo de Inserção de Nós de W. Boehm [26].

Observemos primeiramente que no intervalo  $[0, n + 1]$  a B-Spline de grau  $n$  tem suporte de comprimento  $n + 1$  e assim sofreria um corte em sua “forma”, na medida que a mesma fosse transladada para a esquerda ou para a direita do intervalo. Para levar em consideração esta influência dos bordos  $0$  e  $n + 1$  na forma da B-Spline, seria necessário estender  $[0, n + 1]$  tanto a esquerda como a direita, por  $n$  unidades e posteriormente “comprimir” esta extensão aos respectivos pontos  $0$  e  $n + 1$ . Em outros termos isto equivale a afirmação de que os extremos do intervalos são considerados nós de multiplicidade igual a  $n + 1$ .

Desta forma, consideraremos B-Splines de grau  $n$ , em intervalos limitados por nós de multiplicidade  $n + 1$ , e assim o algoritmo de Inserção de Nós de W. Boehm, vai nos fornecer a matriz de mudança de base para a representação na escala mais fina, com a devida adaptação nos bordos do intervalo.

A construção será feita no intervalo  $[0, 1]$ , através do refinamento dos Polinômios de Bernstein, mas é válida em qualquer intervalo limitado  $[a, b]$ .

## 2.1 Funções B-Splines

**Definição 2.1.1** Consideremos uma sequência de nós não decrescentes, definidos no intervalo  $[t_0, t_m]$  :

$$(t_i) := t_0 \leq \dots t_i \leq \dots \leq t_m$$

Definimos a  $i$ -ésima B-Spline  $N_i^n$  de grau  $n$  ou ordem  $n+1$ , em relação aos nós  $(t_i)$  pela fórmula de recorrência:

$$N_i^n(t) := \lambda_i^n N_i^{n-1}(t) + (1 - \lambda_{i+1}^n) N_{i+1}^{n-1}(t)$$

onde

$$N_i^0(t) := H_i(t) := \begin{cases} 1 & \text{se } t_i \leq t \leq t_{i+1} \\ 0 & \text{se } t < t_i \text{ ou } t > t_{i+1} \end{cases}, \quad \text{sendo } N_i^0(t) := 0, \quad \text{se } t_i = t$$

$$\lambda_i^n := \frac{t - t_i}{t_{i+n} - t_i}, \quad \text{sendo } \lambda_i^n := 0 \quad \text{se } t_{i+n} - t_i = 0$$

**Definição 2.1.2** Se  $t_i = t_{i+1} = \dots = t_{i+r-1}$ , ou seja se  $r$  nós sucessivos coincidem, então dizemos que  $t_i$  tem multiplicidade  $r$ . Se um dos nós não coincide com outro nó, nos dizemos que ele é simples ou tem multiplicidade 1.

**Exemplo 2.1.1** Consideremos os Polinômios de Bernstein de grau 1, no intervalo  $[0, 1]$ , com relação aos nós, de multiplicidade 2:  $t_0 = t_1 = 0 \leq t_1 = t_2 = 1$

$$\text{Se } i = 0 \implies N_0^1(t) := \lambda_0^1 N_0^0(t) + (1 - \lambda_1^0) N_1^0(t)$$

$$N_0^1(t) = \lambda_0^1 \times 0 + (1 - t) \times 1$$

$$N_0^1(t) = 1 - t$$

Por outro lado,

$$\text{Se } i = 1 \implies N_1^1(t) := \lambda_1^1 N_1^0(t) + (1 - \lambda_2^1) N_2^0(t)$$

$$N_1^1(t) = t \times 1 + (1 - t) \times 0$$

$$N_1^1(t) = t$$

**Exemplo 2.1.2** Consideremos os Polinômios de Bernstein de grau 2, no intervalo  $[0, 1]$ , com relação aos nós, de multiplicidade 3:  $t_0 = t_1 = t_2 = 0 \leq t_1 = t_2 = t_3 = 1$ .

$$\text{Se } i = 0 \implies N_0^2(t) := \lambda_0^2 N_0^1(t) + (1 - \lambda_1^2) N_1^1(t) = \lambda_0^2 \times 0 + (1 - t) N_1^1(t)$$



onde,

$$N_1^1(t) = \lambda_1^1 N_1^0(t) + (1 - \lambda_2^1) N_2^0(t) = \lambda_1^1 \times 0 + (1 - t) \times 1 = 1 - t$$

logo,

$$N_0^2(t) := (1 - t) \times (1 - t) = (1 - t)^2$$

$$\text{Se } i = 1 \implies N_1^2(t) := \lambda_1^2 N_1^1(t) + (1 - \lambda_2^2) N_2^1(t) = t \times N_1^1(t) + (1 - t) N_2^1(t)$$

onde,

$$N_1^1(t) = \lambda_1^1 N_1^0(t) + (1 - \lambda_2^1) N_2^0(t) = \lambda_1^1 \times 0 + (1 - t) \times 1 = 1 - t$$

e,

$$N_2^1(t) = \lambda_2^1 N_2^0(t) + (1 - \lambda_3^1) N_3^0(t) = t \times 1 + (1 - \lambda_3^1) \times 0 = t$$

logo,

$$N_1^2(t) := t \times (1 - t) + (1 - t) \times t = 2t(1 - t)$$

$$\text{Se } i = 2 \implies N_2^2(t) := \lambda_2^2 N_2^1(t) + (1 - \lambda_3^2) N_3^1(t) = t \times N_2^1(t) + 0 \times N_3^1(t)$$

onde,

$$N_2^1(t) = \lambda_2^1 N_2^0(t) + (1 - \lambda_3^1) N_3^0(t) = t \times 1 + (1 - \lambda_3^1) \times 0 = t$$

logo,

$$N_2^2(t) := t \times t = t^2$$

Pelos exemplos acima, podemos em geral, denotando o  $n$ -ésimo Polinômio de Bernstein de grau  $n$  por  $B_i^n := N_i^n$ , onde  $N_i^n$  é a  $i$ -ésima B-Spline de grau  $n$ , definida no intervalo  $[0, 1]$  de nós  $t_0 = 0$  e  $t_1 = 1$  com multiplicidades  $n+1$ , ter a relação de recorrência,

$$B_0^0 = 1$$

$$B_i^n = (1 - t) B_i^{n-1} + t B_{i-1}^{n-1}$$

Donde, temos a fórmula:

$$B_i^n = \frac{n!}{(n-i)!i!} t^i (1-t)^{n-i}$$

Que pelo desenvolvimento binomial, demonstra que os polinômios de Bernstein produzem a partição da unidade:

$$(t + (1 - t))^n = \sum_{i=0}^n B_i^n(t) = 1$$

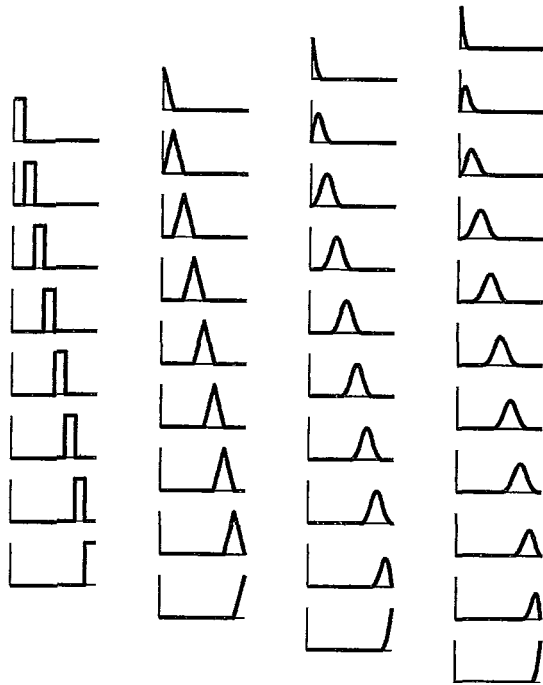


Figura 2.1: B-Splines de grau 0, 1, 2 e 3 no nível três de resolução

Desta forma, temos os polinômios de Bernstein de graus 0, 1, 2, e 3, representando respectivamente as B-Splines de Haar, linear, quadrática e cúbica, no intervalo  $[0, 1]$ , dados por:

$$\begin{array}{cccc}
 1 & & & \\
 1-t & t & & \\
 (1-t)^2 & 2t(1-t) & t^2 & \\
 (1-t)^3 & 3t(1-t)^2 & 3t^2(1-t) & t^3
 \end{array}$$

Na figura 2.1 temos os polinômios de Bernstein de grau, 0, 1, 2 e 3 no nível 3 de resolução.

S. Bernstein em 1912, [5], usou o desenvolvimento em sua base,

$$x_n(t) = \sum_{i=0}^n c\left(\frac{i}{n}\right) B_i^n(t)$$

para considerar uma  $n$ -ésima aproximação polinomial de uma curva contínua  $c(\cdot)$  em  $[0, 1]$  e demonstrar assim o Teorema de Aproximação de Weirstrass,

$$\lim_{n \rightarrow \infty} x_n(t) = c(t)$$

Por outro lado, se considerarmos uma curva polinomial  $c(\cdot)$  definida em  $[0, 1]$  de grau  $n$ , então temos, usando o desenvolvimento de  $p(t) = t$ , a representação paramétrica da curva  $c(\cdot)$  no  $R^2$ , na base de Bernstein dada por

$$(c(t), t) = \left( \sum_{i=0}^n c\left(\frac{i}{n}\right) B_i^n(t), \sum_{i=0}^n \frac{i}{n} B_i^n(t) \right) = \sum_{i=0}^n \left( c\left(\frac{i}{n}\right), \frac{i}{n} \right) B_i^n(t)$$

onde os coeficientes  $(c(\frac{i}{n}), \frac{i}{n})$  são exatamente os “pontos de controle de Bézier” da curva polinomial  $c(\cdot)$ . No sistema PostScript, as letras são representadas pelos seus pontos de controle e posteriormente, na impressão, renderizadas até o nível de aproximação desejado usando, por exemplo, o algoritmo iterativo de Calsteljau.

Na figura 2.2 abaixo temos uma curva quádrica de Bézier renderizada no latex2 pelo comando `\qbezier`

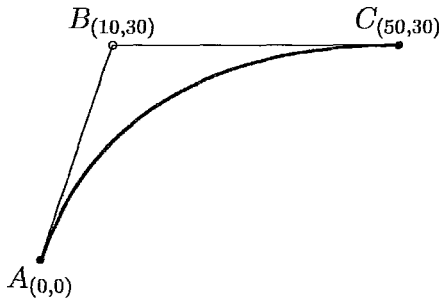


Figura 2.2: Curva de Bézier do segundo grau, renderizada no latex2.

e a figura 2.3 apresenta um desenho de Pierre E. Bézier, usando a sua curva.

## 2.2 Spline na forma B-Spline

**Definição 2.2.1** *Denominamos o espaço vetorial das Splines de grau  $n$ ,  $S^n(t_0 \dots t_m)$  em relação aos nós*

$$(t_i) := t_0 \leq \dots t_i \leq \dots \leq t_m$$

*o espaço dos polinômios  $p(\cdot)$  definidos por partes em  $[t_0, t_1]$ , tais que:*

*$p(\cdot)$  restrito ao intervalo  $[t_i, t_{i+1}]$  é um polinômio de grau  $n$ , e*

*$p(\cdot)$  possui derivada de ordem  $n - 1$  em  $t_i$*

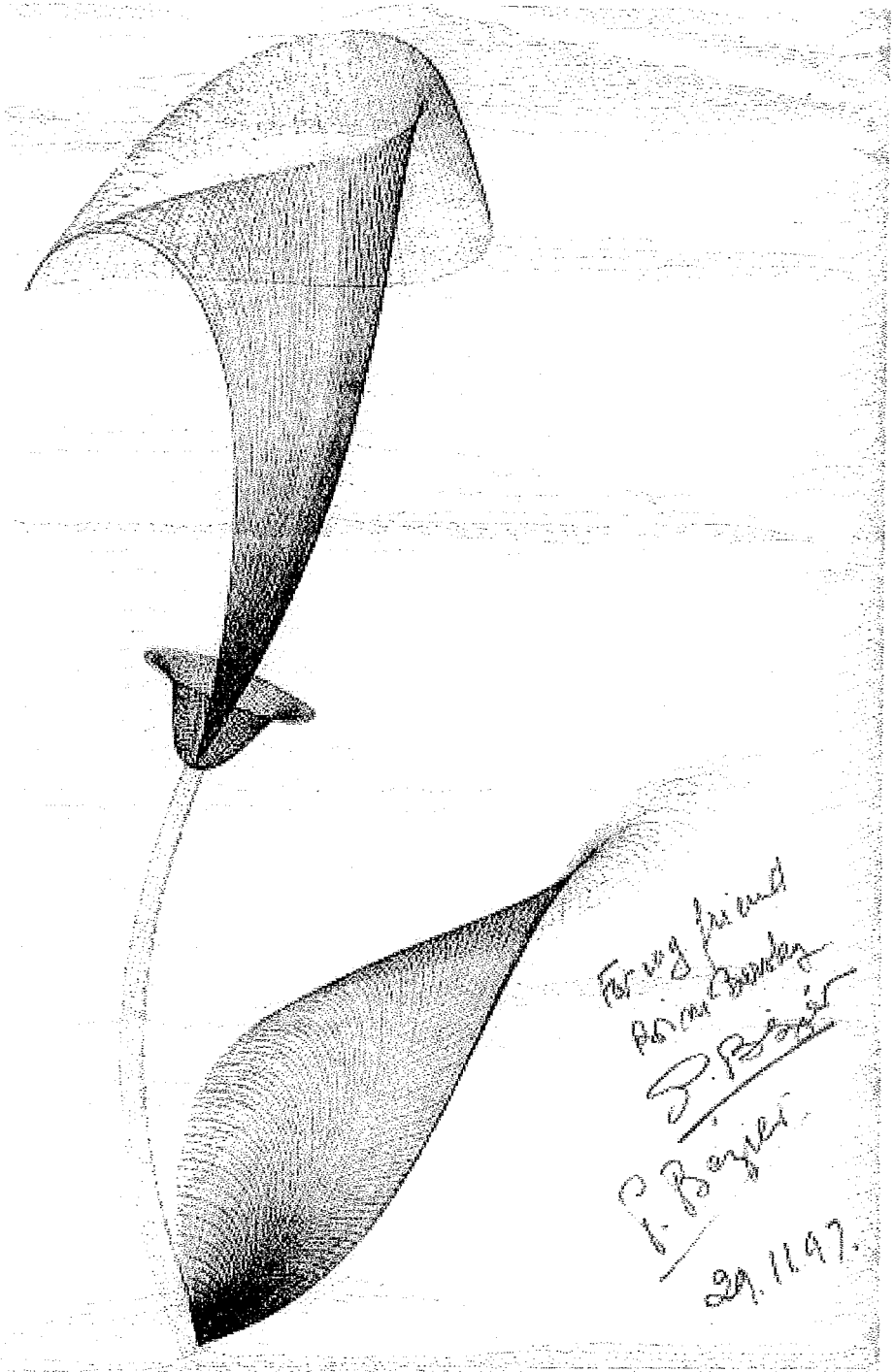


Figura 2.3: Desenho de Pierre E. Bézier feito com a sua curva.

**Definição 2.2.2** Denominamos uma Spline  $S(\cdot)$  na forma B(Basic)-Spline em relação a sequência de nós

$$(t_i) := t_0 \leq \dots t_i \leq \dots \leq t_m$$

quando a mesma está escrita como combinação linear da base gerada pelas  $N_i^n(\cdot)$ :

$$S(t) = \sum_{i=0}^{m-(n+1)} s_i N_i^n(t) = \sum_{i=j-n}^j s_i N_i^n(t)$$

**Definição 2.2.3** Os pontos do plano  $(t, s)$  de coordenadas  $P_i = (\xi_i, s_i)$  onde

$$\xi_i = \frac{\xi_{i+1} \cdots \xi_{i+n}}{n}$$

é denominada, a abscissa de Greville de  $s_i$ , são ditos pontos de controle, ou pontos de Boor, da spline  $S(\cdot)$  da definição anterior.

**Proposição 2.2.1 ( Algoritmo de Boor )** Podemos avaliar a B-Spline  $S(\cdot)$  pelas coordenadas  $s_i$  na representação B-Spline,

$$S(t) = \sum_{i=0}^{m-(n+1)} s_i N_i^n(t)$$

considerando a recursão em cima dos coeficientes  $s_i$ ,

$$s_i^0 = s_i \quad \text{para } i = j - n, \dots, j$$

$$s_i^k = \frac{t - t_i}{t_{i+n-k+1} - t_i} s_i^{k-1} + \frac{t_{i+n-k+1} - t}{t_{i+n-k+1} - t_{i-1}} s_{i-1}^{k-1} \quad \text{para } i = j + k - n, \dots, j - 1, j$$

então  $S(t) = s_i^n(t)$

*Demonstração:* Notemos que, se  $t_j \leq t \leq t_{j+1}$ ,

$$\begin{aligned} S(t) &= \sum_{i=0}^{m-(n+1)} s_i N_i^n(t) = \sum_{i=j-n}^j s_i N_i^n(t) = \\ &= \dots + s_{i-1}^0 N_{i-1}^n + s_i^0 N_i^n + \dots = \\ &= \dots s_{i-1}^0 (\lambda_i^n N_{i-1}^{n-1} + (1 - \lambda_i^n) N_i^{n-1}) + s_i^0 (\lambda_i^n N_i^{n-1} + (1 - \lambda_{i+1}^n) N_{i+1}^{n-1}) \dots = \\ &= s_{i-1}^0 \lambda^n N_{i-1}^{n-1} + (s_{i-1}^0 (1 - \lambda_i^n) + s_i^0 \lambda_i^n) N_i^{n-1} + s_i^0 (1 - \lambda_i^n) N_i^{n-1} \dots = \end{aligned}$$

$$S(t) == \sum_{i=j-(n-1)}^j s_i^1 N_i^{n-1}(t)$$

Daí para a primeira iteração  $k=1$ , temos:

$$s_i^1 = s_{i-1}^0(1 - \lambda_i^n) + s_i^0 \lambda_i^n$$

ou, como

$$\lambda_i^n := \frac{t - t_i}{t_{i+n} - t_i}$$

temos,

$$s_i^1 = \frac{t - t_i}{t_{i+n} - t_i} s_i^{k-1} + \frac{t_{i+n} - t}{t_{i+n} - t_i} s_{i-1}^{k-1}$$

**Exemplo 2.2.1** Para calcular, uma B-Spline cúbica  $S(t) = \sum_{i=j-n}^j s_i^0 N_i^3(t)$  em  $t_j \leq t \leq t_{j+1}$ , temos a seguinte sequência de avaliações:

$$s_{j-3}^0 \quad s_{j-2}^0 \quad s_{j-1}^0 \quad s_j^0$$

primeira iteração,

$$s_{j-2}^1 \quad s_{j-1}^1 \quad s_j^1$$

segunda iteração,

$$s_{j-1}^2 \quad s_j^2$$

terceira iteração,

$$s_j^3$$

## 2.3 Refinamento de nós

**Definição 2.3.1** Uma sequência não decrescente de nós  $(\tau_i) := \tau_0 \leq \dots \leq \tau_i \dots \leq \tau_l$  é um refinamento da sequência  $(t_i) := t_0 \leq \dots \leq t_i \leq \dots \leq t_m$  onde  $l > m$  se para todo  $0 \leq i \leq n$  existe  $j(i)$  tal que  $t_i = \tau_{i(j)}$ .

**Definição 2.3.2** Dizemos que o sub-espaço vetorial  $S^n(\tau_0 \dots \tau_l)$  é um refinamento do espaço vetorial  $S^n(t_0 \dots t_m)$ .

**Proposição 2.3.1 (Algoritmo de inserção de nós de W. Boehm)** Se  $\widehat{N}^n$  é uma nova B-Spline de grau  $n$  associada ao novo nó  $\hat{t}$  e a B-Spline é dada por,

$$S(t) = \sum_{i=j-n}^j s_i N_i^n(t)$$

então, renumerando os nós, pela inclusão de  $\hat{t}$ , temos:

$$N_i^n(t) = \frac{\hat{t} - t_i}{t_{i+n} - t_i} \widehat{N}_i^n(t) + \frac{t_{i+1+n} - \hat{t}}{t_{i+1+n} - t_{i+1}} \widehat{N}_{i+1}^n(t)$$

*Demonstração:* A B-Spline  $N_i^n(t)$  é definida pelos coeficientes  $s_j = \delta_{ij} = \begin{cases} 1 & \text{se } j = i \\ 0 & \text{se } j \neq i \end{cases}$ . Com a inserção do novo nó  $\hat{t}$  temos, os dois novos coeficientes  $\hat{s}_i$  e  $\hat{s}_{i+1}$  em relação a nova base mais refinada, dada por  $\widehat{N}_i^n$ . Usando o Algoritmo de Boor temos:

$$\hat{s}_i = \lambda_i^n s_i^0 + (1 - \lambda_i^n) s_{i-1}^0$$

$$\hat{s}_i = \frac{\hat{t} - t_i}{t_{i+n} - t_i} s_i^0 + \frac{t_{i+n} - \hat{t}}{t_{i+n} - t_i} s_{i-1}^0$$

donde

$$\hat{s}_i = \frac{\hat{t} - t_i}{t_{i+n} - t_i} \times 1 + \frac{t_{i+n} - \hat{t}}{t_{i+n} - t_i} \times 0 = \frac{\hat{t} - t_i}{t_{i+n} - t_i}$$

por outro lado,

$$\hat{s}_{i+1} = \lambda_{i+1}^n s_{i+1}^0 + (1 - \lambda_{i+1}^n) s_i^0$$

$$\hat{s}_{i+1} = \frac{\hat{t} - t_{i+1}}{t_{i+1+n} - t_{i+1}} s_{i+1}^0 + \frac{t_{i+1+n} - \hat{t}}{t_{i+1+n} - t_i} s_i^0$$

donde

$$\hat{s}_{i+1} = \frac{\hat{t} - t_{i+1}}{t_{i+1+n} - t_{i+1}} \times 0 + \frac{t_{i+1+n} - \hat{t}}{t_{i+1+n} - t_i} \times 1 = \frac{t_{i+1+n} - \hat{t}}{t_{i+1+n} - t_i}$$

Logo,

$$N_i^n(t) = \hat{s}_i \widehat{N}_i^n(t) + \hat{s}_{i+1} \widehat{N}_{i+1}^n(t)$$

donde,

$$N_i^n(t) = \frac{\hat{t} - t_i}{t_{i+n} - t_i} \widehat{N}_i^n(t) + \frac{t_{i+1+n} - \hat{t}}{t_{i+1+n} - t_{i+1}} \widehat{N}_{i+1}^n(t)$$

## 2.4 Refinamento diádico e a equação associada

**Definição 2.4.1** Denominamos refinamento diádico ou simplesmente refinamento de resolução  $j$  do intervalo  $[0, 1]$  a sequência de nós  $(t_i)$  abaixo, onde 0 e 1 são tomados com multiplicidade  $n+1$ ,

$$(t_0, \dots, t_{2^j+2n}) = \left( \overbrace{0, \dots, 0}^{n+1}, \frac{1}{2^j}, \frac{2}{2^j}, \dots, \frac{2^j - 2}{2^j}, \frac{2^j - 1}{2^j}, \overbrace{1, \dots, 1}^{n+1} \right)$$

**Definição 2.4.2** Denominamos espaço vetorial de resolução  $j$  das B-Splines de grau  $n$ , definidas em  $[0, 1]$ , que interpolam os pontos extremos, denotado por  $V^{j,n}$  ou simplesmente  $V^j$ , o espaço vetorial gerado pelas B-Splines de grau  $n$   $\phi_i^j(t) = N^n(2^j t - i)$ , de resolução  $j$ , associadas ao refinamento  $(t_i^j)$  diádico de  $[0, 1]$ , onde  $\phi_i^0(t)$  são os polinômios de Bernstein, dimensão de  $V^{j,n}$  é  $2^j + n$  e  $V^j \subset V^{j+1}$ .

**Definição 2.4.3** Denominamos equação de refinamento para as resoluções  $j$  e  $j-1$ , a equação representando a combinação linear da base  $\{\phi_i^{j-1}(\cdot)\}$  em relação a base  $\{\phi_i^j(\cdot)\}$ , dada por,

$$\phi_r^{j-1} = \sum_i \phi_i^j h[i]$$

Ou usando a notação matricial,

$$\underbrace{\left( \phi_0^{j-1} \dots \phi_{v_{j-1}-1}^{j-1} \right)}_{\Phi^{j-1}} = \underbrace{\left( \phi_0^j \dots \phi_{v_j-1}^j \right)}_{\Phi^j} \underbrace{\begin{pmatrix} \dots & & & & \\ & h[0] & & & \\ & & h[1] & & \\ & & & \dots & \end{pmatrix}}_{H^j}$$

ou,

$$\Phi^{j-1} = \Phi^j H^j$$

onde  $H^j$  representa um filtro passa-baixa.

**Exemplo 2.4.1** Equações de refinamento para splines cúbicas definidas em  $[0, 1]$ , que interpolam os pontos extremos.

Consideremos no nível 0 de resolução, o espaço vetorial  $V^{0,3}$  gerado pelos polinômios de Bernstein de grau 3, representado pelo vetor,

$$\Phi^0(t) = \left( \underbrace{(1-t)^3}_{\Phi_0^0(t)}, \underbrace{3t(1-t)^2}_{\Phi_1^0(t)}, \underbrace{3t^2(1-t)}_{\Phi_2^0(t)}, \underbrace{t^3}_{\Phi_3^0(t)} \right)$$

onde os nós são dados pelo vetor,

$$T^0 = (0, 0, 0, 0, 1, 1, 1, 1)$$



e as abscissas de Greville pelo vetor,

$$G^0 = (0, \frac{1}{3}, \frac{2}{3}, 1)$$

no nível 1 de resolução temos o espaço vetorial  $V^{1,3}$  gerado pelas splines cúbicas refinadas dadas pelo vetor,

$$\Phi^1(t) = (\phi_0^1(t), \phi_1^1(t), \phi_2^1(t), \phi_3^1(t))$$

onde os nós são dados pelo vetor,

$$T^1 = (0, 0, 0, 0, \frac{1}{2}, 1, 1, 1, 1)$$

e as abscissas de Greville pelo vetor,

$$G^1 = (0, \frac{1}{6}, \frac{3}{6}, \frac{5}{6}, 1)$$

Para determinar  $H^1$  na equação de refinamento  $\Phi^0 = \Phi^1 H^1$  notemos que escrevendo  $\Phi^0$  em relação a  $\Phi^0$  temos matricialmente,

$$\Phi^0 = \Phi^0 I$$

onde  $I$  é a matriz identidade, e temos assim os coeficientes de  $\Phi_i^0$ , no nível zero, calculados nas abscissas  $G^j = (g_i^j)$  de Greville correspondente,

$$\begin{array}{l} g_0^0 = 0 \longrightarrow \\ g_1^0 = \frac{1}{3} \longrightarrow \\ g_2^0 = \frac{2}{3} \longrightarrow \\ g_3^0 = 1 \longrightarrow \end{array} \begin{array}{c} \phi_0^0 \downarrow \\ \phi_1^0 \downarrow \\ \phi_2^0 \downarrow \\ \phi_3^0 \downarrow \end{array} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{array}{l} \longleftarrow \phi_0^0 \\ \longleftarrow \phi_1^0 \\ \longleftarrow \phi_2^0 \\ \longleftarrow \phi_3^0 \end{array}$$

Usando agora o Algoritmo de Inserção de Nós de W. Boehm, teremos  $H^1$  dado por,

$$\begin{array}{l} g_0^1 = 0 \longrightarrow \\ g_1^1 = \frac{1}{6} \longrightarrow \\ g_2^1 = \frac{3}{6} \longrightarrow \\ g_3^1 = \frac{5}{6} \longrightarrow \\ g_4^1 = 1 \longrightarrow \end{array} \begin{array}{c} \phi_0^0 \downarrow \\ \phi_1^0 \downarrow \\ \phi_2^0 \downarrow \\ \phi_3^0 \downarrow \end{array} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0.5 & 0.5 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 \\ 0 & 0 & 0.5 & 0.5 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{array}{l} \longleftarrow \phi_0^1 \\ \longleftarrow \phi_1^1 \\ \longleftarrow \phi_2^1 \\ \longleftarrow \phi_3^1 \\ \longleftarrow \phi_4^1 \end{array}$$

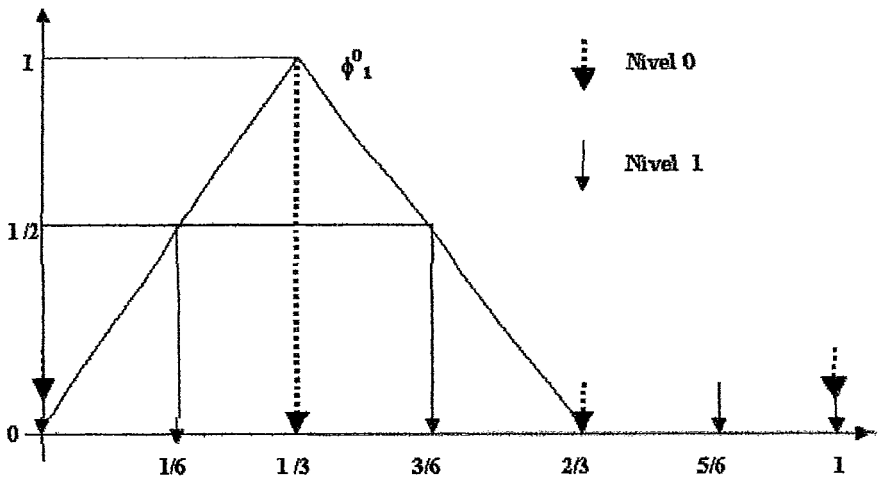


Figura 2.4: O algoritmo de inserção de nós de W. Boem

# Capítulo 3

## B-Wavelets semi-ortogonais sobre intervalos limitados

Através do conceito de multi-resolução apresentaremos as B-Wavelets sobre intervalos limitados, adaptadas aos bordos, que possuem a propriedade de serem ortogonais as B-Splines do mesmo nível de resolução tal como apresentadas por CHUI [27]. As equações de refinamento para as wavelets entre níveis distintos de resolução fornecerão a matriz, adaptada aos bordos, que representa um filtro passa-alta. Os resultados e notações aqui apresentados baseiam-se nas referências [23, 28, 29].

### 3.1 B-Wavelets

Consideremos  $V^j$ , o espaço vetorial de resolução  $j$  das B-Splines de grau  $n$  definidas em  $[0, 1]$ , que interpolam os pontos extremos, e tomemos a decomposição,

$$V^j = V^{j-1} \oplus W^{j-1}$$

onde  $W^{j-1}$  é o espaço vetorial das B-Splines que representam a diferença ou "detalhes" de  $V^{j-1}$  em relação a  $V^j$ .

**Definição 3.1.1** *Denominamos espaço vetorial de resolução  $j$  das B-Wavelets de grau  $n$ , definidas em  $[0, 1]$ , que interpolam os pontos extremos, denotado por  $W^{j,n}$  ou simplesmente  $W^j$ , o espaço vetorial gerado pelas B-Splines de grau  $n$   $\psi_i^j(t) = \psi^0(2^j t - i)$ , de resolução  $j$ , associadas ao refinamento  $(t_i^j)$  diádico de  $[0, 1]$ , onde*

$$\psi^0(t) = \begin{cases} 1 & \text{se } 0 \leq t < 1/2 \\ -1 & \text{se } 1/2 \leq t \leq 1 \end{cases}$$

são as *B-Wavelets* de resolução 0 e a dimensão de  $W^{j,n}$  é  $2^j$ .

**Definição 3.1.2** Denominamos componentes de  $f \in V^j$  em relação a base  $\Phi^j$  e denotadas por  $[f]_{\Phi^j}$  as coordenadas de  $f$  em relação a  $\Phi^j$ . Assim, se

$$f = C^j \Phi^j = \sum c_i^j \phi_i^j = \sum_i c_i^{j-1} \phi_i^{j-1} + \sum_i d_i^{j-1} \psi_i^{j-1} = C^{j-1} \Phi^{j-1} + C^{j-1} \Psi^{j-1}$$

então, as componentes de  $f$  nos níveis  $j$  e  $j-1$  de resolução, são dadas respectivamente por,

$$[f]_{\Phi^j} = C^j$$

e

$$[f]_{(\Phi^{j-1} \Psi^{j-1})} = \begin{pmatrix} C^{j-1} \\ D^{j-1} \end{pmatrix}$$

onde  $D^{j-1}$  representam os detalhes da resolução do nível  $j-1$  em relação ao nível  $j$ .

**Definição 3.1.3** Denominamos equação de refinamento para as *B-Wavelets* em relação as resoluções  $j$  e  $j-1$ , a equação representando a combinação linear da base  $\{\psi_i^{j-1}(\cdot)\}$  em relação a base  $\{\phi_i^j(\cdot)\}$ , dada por,

$$\psi_r^{j-1} = \sum_i \phi_i^j g[i]$$

Ou usando a notação matricial,

$$\underbrace{(\psi_0^{j-1} \dots \psi_{w_{j-1}-1}^{j-1})}_{\Psi^j} = \underbrace{(\phi_0^j \dots \phi_{v_j-1}^j)}_{\Phi^j} \underbrace{\begin{pmatrix} \dots & & & & \\ & g[0] & & & \\ & g[1] & & & \\ & & \dots & & \end{pmatrix}}_{G^j}$$

ou,

$$\Psi^{j-1} = \Phi^j G^j$$

onde  $G^j$  representa um filtro passa-alta.

## 3.2 Mudança de bases entre resoluções distintas

**Definição 3.2.1** Denominamos matriz mudança de bases entre as resoluções  $j - 1$  e  $j$  a matriz  $(H^j \ G^j)$ , formada pela justaposição dos filtros de síntese passa-baixa  $H^j$  e passa-alta  $G^j$ , que verifica a equação,

$$(\Phi^{j-1} \ \Psi^{j-1}) = \Phi^j (H^j \ G^j) \quad (3.1)$$

ou ainda,

$$\begin{aligned} \Phi^{j-1} &= \Phi^j H^j \\ \Psi^{j-1} &= \Phi^j G^j \end{aligned}$$

**Definição 3.2.2** Denominamos matriz mudança de bases duais, entre as resoluções  $j - 1$  e  $j$ , a matriz  $(\widetilde{H}^j \ \widetilde{G}^j)$ , formada pela justaposição dos filtros de análise passa-baixa  $\widetilde{H}^j$  e passa-alta  $\widetilde{G}^j$ , que verifica a equação,

$$(\widetilde{\Phi}^{j-1} \ \widetilde{\Psi}^{j-1}) = \widetilde{\Phi}^j (\widetilde{H}^j \ \widetilde{G}^j) \quad (3.2)$$

ou ainda,

$$\begin{aligned} \widetilde{\Phi}^{j-1} &= \widetilde{\Phi}^j \widetilde{H}^j \\ \widetilde{\Psi}^{j-1} &= \widetilde{\Phi}^j \widetilde{G}^j \end{aligned}$$

onde,

$$(\widetilde{H}^j \ \widetilde{G}^j)^T = (H^j \ G^j)^{-1} \quad (3.3)$$

### Proposição 3.2.1

$$\Phi^j = (\Phi^{j-1} \ \Psi^{j-1}) \begin{pmatrix} (\widetilde{H}^j)^T \\ (\widetilde{G}^j)^T \end{pmatrix}$$

ou ainda

$$\Phi^j = \Phi^{j-1} (\widetilde{H}^j)^T + \Psi^{j-1} (\widetilde{G}^j)^T$$

*Demonstração:* de 3.1 e 3.3, temos

$$\Phi^j = (\Phi^{j-1} \ \Psi^{j-1}) (H^j \ G^j)^{-1} = (\Phi^{j-1} \ \Psi^{j-1}) (\widetilde{H}^j \ \widetilde{G}^j)^T$$

### Proposição 3.2.2

$$\widetilde{\Phi}^j = (\widetilde{\Phi}^{j-1} \ \widetilde{\Psi}^{j-1}) \begin{pmatrix} (H^j)^T \\ (G^j)^T \end{pmatrix}$$

ou ainda

$$\tilde{\Phi}^j = \tilde{\Phi}^{j-1} (H^j)^T + \tilde{\Psi}^{j-1} (G^j)^T$$

*Demonstração:* de 3.2 e 3.3, temos

$$\tilde{\Phi}^j = (\tilde{\Phi}^{j-1} \tilde{\Psi}^{j-1}) (\tilde{H}^j \tilde{G}^j)^{-1} = (\tilde{\Phi}^{j-1} \tilde{\Psi}^{j-1}) (H^j G^j)^T$$

**Proposição 3.2.3** *Considere,*

$$f = \sum c_i^j \phi_i^j = \sum_i c_i^{j-1} \phi_i^{j-1} + \sum_i d_i^{j-1} \psi_i^{j-1}$$

ou matricialmente,

$$f = \underbrace{(\phi_0^j \dots \phi_{\dim V^{j-1}}^j)}_{\Phi^j} \underbrace{\begin{pmatrix} c_0^j \\ \vdots \\ c_{\dim V^{j-1}}^j \end{pmatrix}}_{C^j} = \Phi^j C^j$$

Daí, temos a mudança de coordenadas entre as resoluções  $j-1$  e  $j$ ,

$$C^j \mapsto \begin{pmatrix} C^{j-1} \\ D^{j-1} \end{pmatrix} = \begin{pmatrix} \tilde{H}^{jT} C^j \\ \tilde{G}^{jT} C^j \end{pmatrix} \quad (3.4)$$

e,

$$\begin{pmatrix} C^{j-1} \\ D^{j-1} \end{pmatrix} \mapsto C^j = H^j C^{j-1} + G^j D^{j-1}$$

*Demonstração:* Notemos que,

$$f = \Phi^j C^j = \Phi^{j-1} C^{j-1} + \Psi^{j-1} D^{j-1} = (\Phi^{j-1} \Psi^{j-1}) \begin{pmatrix} C^{j-1} \\ D^{j-1} \end{pmatrix} \quad (3.5)$$

daí,

$$\begin{pmatrix} C^{j-1} \\ D^{j-1} \end{pmatrix} = (\Phi^{j-1} \Psi^{j-1})^{-1} \Phi^j C^j$$

Por 3.1

$$\begin{pmatrix} C^{j-1} \\ D^{j-1} \end{pmatrix} = (\Phi^j (H^j G^j))^{-1} \Phi^j C^j = (H^j G^j)^{-1} (\Phi^j)^{-1} \Phi^j C^j = (H^j G^j)^{-1} C^j$$

e por 3.3,

$$\begin{pmatrix} C^{j-1} \\ D^{j-1} \end{pmatrix} = (\tilde{H}^j \tilde{G}^j)^T C^j = \begin{pmatrix} (\tilde{H}^j)^T \\ (\tilde{G}^j)^T \end{pmatrix} C^j = \begin{pmatrix} (\tilde{H}^j)^T C^j \\ (\tilde{G}^j)^T C^j \end{pmatrix}$$

Por outro lado, por 3.1,

$$f = \Phi^j C^j == (\Phi^{j-1} \Psi^{j-1}) \begin{pmatrix} C^{j-1} \\ D^{j-1} \end{pmatrix} = \Phi^j (H^j \ G^j) \begin{pmatrix} C^{j-1} \\ D^{j-1} \end{pmatrix}$$

logo,

$$C^j = (H^j \ G^j) \begin{pmatrix} C^{j-1} \\ D^{j-1} \end{pmatrix} = H^j C^{j-1} + G^j D^{j-1}$$

**Proposição 3.2.4** *As componentes wavelets de  $f \in V^j$  em relação aos níveis  $j$  e  $j - 1$  de resolução satisfazem a igualdade,*

$$[f]_{(\Phi^{j-1} \Psi^{j-1})} = \begin{pmatrix} (\widetilde{H}^j)^T \\ (\widetilde{G}^j)^T \end{pmatrix} [f]_{\Phi^j}$$

*Demonstração: consequência imediata de 3.4.*

### 3.3 Transformada Wavelet Discreta

**Definição 3.3.1** *Da proposição 3.2.3 definimos a transformada wavelet discreta que denotaremos por  $FWT(\cdot)$  (Forward Wavelet Transform)*

$$C^j \mapsto FWT(C) = \begin{pmatrix} C^{j-1} \\ D^{j-1} \end{pmatrix} = \begin{pmatrix} (\widetilde{H}^j)^T C^j \\ (\widetilde{G}^j)^T C^j \end{pmatrix}$$

onde  $\widetilde{H}^j$  e  $\widetilde{G}^j$  representam os filtros de análise passa-baixa e passa-alta respectivamente., no nível  $j$  de resolução.

**Definição 3.3.2** *Da proposição 3.2.3 definimos a transformada wavelet discreta inversa que denotaremos por  $IWT(\cdot)$  (Inverse Wavelet Transform),*

$$\begin{pmatrix} C^{j-1} \\ D^{j-1} \end{pmatrix} \mapsto IWT \begin{pmatrix} C^{j-1} \\ D^{j-1} \end{pmatrix} = C^j = H^j C^{j-1} + G^j D^{j-1}$$

onde  $H^j$  e  $G^j$  representam os filtros de síntese passa-baixa e passa-alta respectivamente.

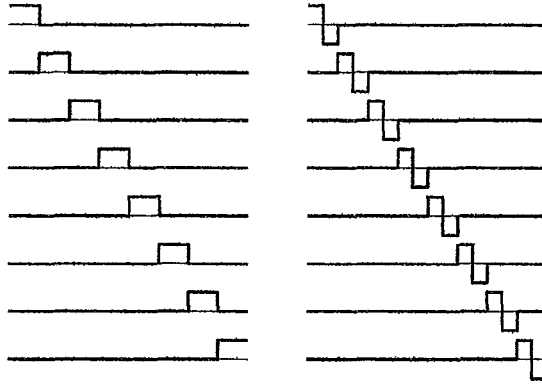


Figura 3.1: Todas as B-Splines (8) e B-Wavelets (8) de grau 0, (HAAR), no nível três de resolução, sobre [0,1]

### 3.4 A transformada B-Wavelet, semi-ortogonal definida sobre intervalo limitado

**Proposição 3.4.1** *Considere  $(\Phi^j \ \Psi^j)$  de tal forma que, para todo  $j \geq 0$ , as wavelets em  $\Psi^j$  sejam ortogonais as funções em  $\Phi^j$ , com relação ao produto interno dos mínimos quadrados dado por  $\langle X|Y \rangle = X^T Y$  então  $\Psi^j$  satisfaz as condições de semi-ortogonalidade,*

$$(H^j)^T \langle \Phi^j | \Phi^j \rangle G^j = 0 \text{ para todo } j \geq 0$$

*Demonstração:* por 3.1 temos,

$$\begin{aligned} \langle \Phi^{j-1} | \Psi^{j-1} \rangle &= (\Phi^{j-1})^T \Psi^{j-1} = (\Phi^j H^j)^T (\Phi^j G^j) = \\ &= (H^j)^T \left( (\Phi^j)^T \Phi^j \right) G^j = (H^j)^T \langle \Phi^j | \Phi^j \rangle G^j = 0 \end{aligned}$$

**Definição 3.4.1** *Denominamos Transformada B-Wavelet semi-ortogonal ou simplesmente Transformada Wavelet, quando os filtros de síntese  $(H^j \ G^j)$  satisfazem as condições da proposição 3.4.1,*

$$(H^j)^T \langle \Phi^j | \Phi^j \rangle G^j = 0 \text{ para todo } j \geq 0$$

Nas figuras 3.1, 3.2, 3.3 e 3.4 podemos ver as funções escalas B-Splines e as respectivas B-Wavelets semi-ortogonais, definidas sobre intervalo, de graus 0, 1, 2 e 3 respectivamente.



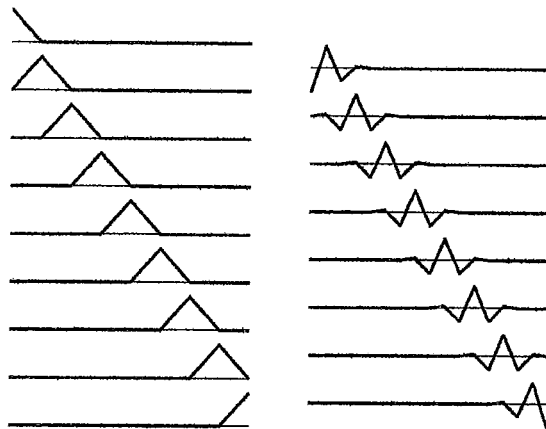


Figura 3.2: Todas as B-Splines (9) e B-Wavelets (8) de grau 1, (Linear), no nível três de resolução, sobre  $[0,1]$

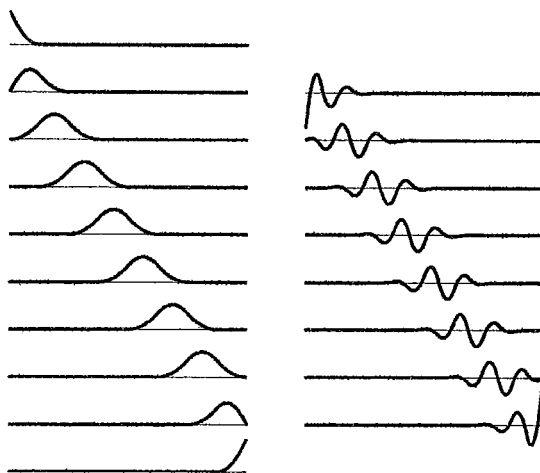


Figura 3.3: Todas as B-Splines (10) e B-Wavelets (8) de grau 2, (Quádricas), no nível três de resolução, sobre  $[0,1]$

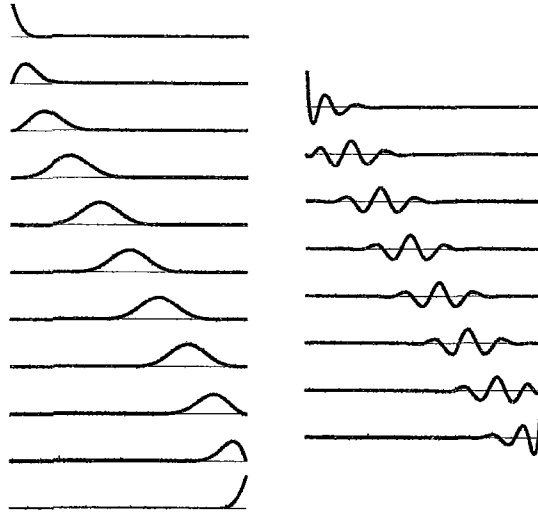


Figura 3.4: Todas as B-Splines (11) e B-Wavelets (8) de grau 3, (Cúbicas), no nível três de resolução, sobre  $[0,1]$

**Exemplo 3.4.1** Consideremos a Transformada B-Wavelet semi-ortogonal, definida no intervalo  $[0, 1]$ , considerando as B-Splines cúbicas definidas em  $[0, 1]$ . Pelo exemplo 2.4.1, no nível 1 de resolução temos o filtro de síntese passa-baixo dado por,

$$H^1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0.5 & 0.5 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 \\ 0 & 0 & 0.5 & 0.5 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

A condição de semi-ortogonalidade impõe o cálculo do filtro passa-alta de síntese  $G^1$ , de tal forma que:

$$(H^1)^T \langle \Phi^1 | \Phi^1 \rangle G^1 = 0$$

onde  $\Phi^1(\cdot)$  são os polinômios de Bernstein cúbicos reescalados para o nível 1 de resolução. Daí, calculando temos,

$$G^1 = (1 \quad -2 \quad 3 \quad -2 \quad 1)^T$$

Os filtros de análise são calculados por 3.3,

$$(\widetilde{H}^1 \widetilde{G}^1)^T = (H^1 G^1)^{-1}$$

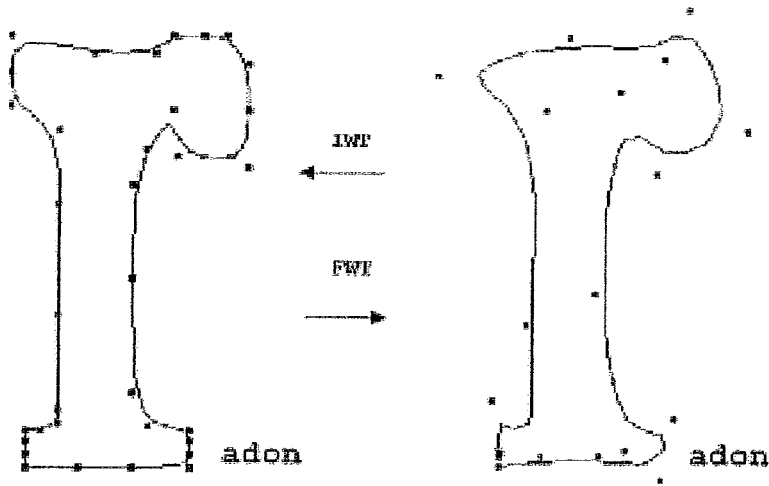


Figura 3.5: Transformada B-Wavelet sobre as coordenadas dos 35 pontos de controle da letra r

Logo, calculando a inversa, temos,

$$\begin{aligned}
 (\widetilde{H}^1)^T &= \begin{pmatrix} 0.9375 & 0.1250 & -0.1250 & 0.1250 & -0.0625 \\ -0.6875 & 1.3750 & 0.6250 & -0.6250 & 0.3125 \\ 0.3125 & -0.6250 & 0.6250 & 1.3750 & -0.6875 \\ -0.0625 & 0.1250 & -0.1250 & 0.1250 & 0.9375 \end{pmatrix} \\
 (\widetilde{G}^1)^T &= (0.0625 \quad -0.1250 \quad 0.1250 \quad -0.1250 \quad 0.0625)
 \end{aligned}$$

Na figura 3.5 podemos ver o resultado da Transformada B-Wavelet cúbica, atuando sobre os pontos de controle  $P_i = (x_i, y_i)$  onde  $1 \leq i \leq 35$ , que modelam geométricamente a letra "r" adon, através de B-Splines cúbicas.

Nas figuras 3.6 ,3.7 e 3.8 temos a decomposição da letra "r" do nível 6 de resolução com  $2^6 + 3 = 35$  pontos de controles até ao nível 0 de resolução com  $2^0 + 3 = 4$  pontos de controle e na figura 3.9 temos a reconstrução, através da Transformada Wavelet Inversa (IWT), da letra "r", do nível 0 até o nível 6 de resolução.

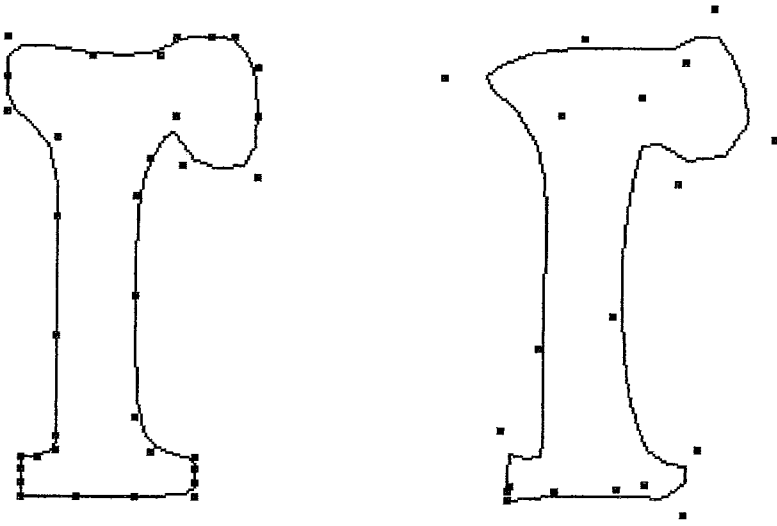


Figura 3.6: A letra r nos níveis 5 e 4 de resolução

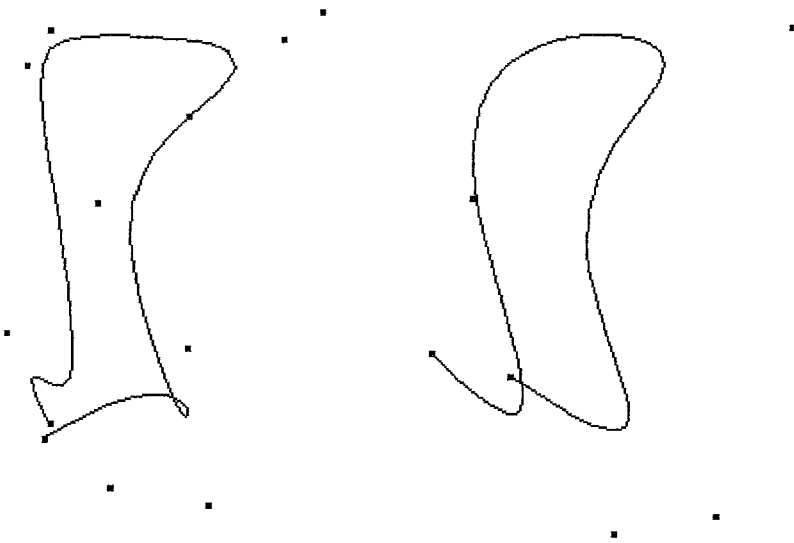


Figura 3.7: A letra r nos níveis 3 e 2 de resolução

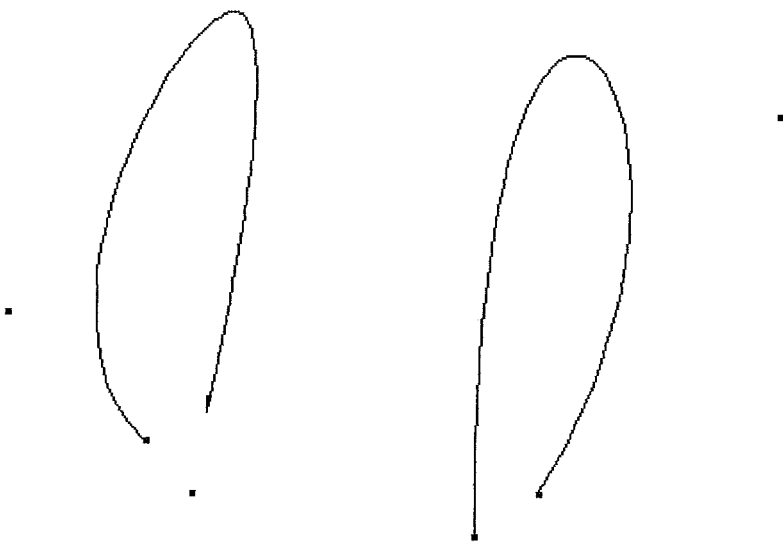


Figura 3.8: A letra r nos níveis 1 e 0 de resolução

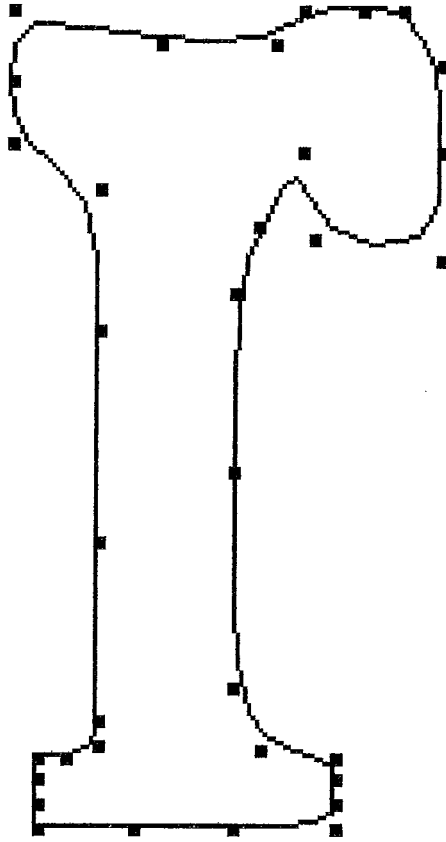


Figura 3.9: Reconstrução da letra r

## Capítulo 4

# Aplicação das B-Wavelets em modelagem geométrica variacional

A aplicação que aqui será tratada tem por base ajustar uma curva a pontos do plano de tal forma que a energia de curvatura (bend) interna seja mínima. Este problema é análogo ao da deformação de uma barra flexível, sujeita a vínculos, sem considerar a energia de estiramento.

No contexto dos elementos finitos procura-se uma solução formada por combinação linear de splines cúbicas associadas a uma malha linear construída ao longo do intervalo de definição da barra, que minimiza o funcional de energia. O problema aqui será resolvido de maneira semelhante onde procuraremos uma solução spline cúbica descrita como combinação linear de B-Splines associadas aos pontos (partículas) do refinamento do intervalo, semelhante aos métodos sem malha ( meshless) [30] como o MRKP (Multiresolution Reproducing Kernel Particle Method) [31] .

O sistema de equações lineares formado pelos coeficientes será resolvido pelo algoritmo iterativo do gradiente conjugado. Veremos que ao introduzir as B-Wavelets junto com as B-Splines o sistema será mais bem condicionado e desta forma a solução será obtida com um menor número de iterações. Isto equivale a representar a curva em vários níveis de resolução, de tal forma que o reduzido número dos coeficientes em relação as B-Splines (filtro passa-baixa), que representam a curva globalmente em determinado nível de resolução, se ajustam mais rapidamente aos vínculos, enquanto que os coeficiente em relação as B-Wavelets (filtro passa-alta), que representam os detalhes, procuram ajustar a curva localmente.

Resolveremos um problema análogo ao da minimização da energia de curvatura da curva sujeita a 3 vínculos como apresentado em [32]. No entanto

em vez de usar B-Wavelets cúbicas bi-ortogonais não adaptadas aos bordos nossa contribuição se baseia no uso das B-Wavelets cúbicas *semi-ortogonais adaptadas aos bordos* com os respectivos *filtros matricias*.

Outras abordagens relativas a problema análogo de mimimização de funcionais de energia se encontram em [33, 34, 35, 36].

## 4.1 Apresentação do problema de modelagem geométrica variacional

Consideremos o problema de encontrar uma curva  $\gamma(t) \in R$ , onde  $t \in [0, 1]$ , que satisfaça as condições,

$$\text{minimize a energia de deformação } \int_0^1 |\gamma(t)''|^2 dt$$

$$\text{sujeita aos vínculos } \begin{cases} \gamma(0) = 3 \\ \gamma(1/2) = 18 \\ \gamma(1) = 12 \end{cases}$$

Vamos resolver pelo método de Ritz, utilizando-se de um polinômio do quarto grau, tal como apresentado em [23],

$$\gamma(t) = x_1 + x_2 t + x_3 t^2 + x_4 t^3 + x_5 t^4 = U(t) x$$

onde  $U(t) = [1 \ t \ t^2 \ t^3 \ t^4]$  e  $X = [x_1 \ x_2 \ x_3 \ x_4 \ x_5]^T$

Daf,

$$\begin{aligned} \int_0^1 |\gamma''|^2 dt &= \int_0^1 (U''(t) X) (U''(t) X) dt = \\ &= \int_0^1 (U''(t) X) (U''(t) X) dt = \int_0^1 (X^T (U''(t))^T) (U''(t) X) dt = \\ &= \int_0^1 X^T ((U''(t))^T U''(t)) X dt = \\ &= \int_0^1 X^T ((U''(t))^T U''(t)) X dt = X^T \left( \int_0^1 (U''(t))^T U''(t) dt \right) X = \frac{1}{2} X^T H X \end{aligned}$$

onde, a matriz H, denominada Hessiana, é dada por,

$$H = 2 \int_0^1 (X''(t))^T X''(t) dt = \frac{4}{5} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10 & 15 & 20 \\ 0 & 0 & 15 & 30 & 45 \\ 0 & 0 & 20 & 45 & 72 \end{pmatrix}$$



Com relação aos vínculos, temos o seguinte sistema de equações lineares,

$$\begin{pmatrix} \gamma(0) \\ \gamma(1/2) \\ \gamma(1) \end{pmatrix} = \begin{pmatrix} U(0) \\ U(1/2) \\ U(1) \end{pmatrix} X = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & \frac{1}{2} & \frac{1}{4} & \frac{1}{8} & \frac{1}{16} \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix} X = \begin{pmatrix} 3 \\ 18 \\ 12 \end{pmatrix}$$

onde considerando

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & \frac{1}{2} & \frac{1}{4} & \frac{1}{8} & \frac{1}{16} \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix} \text{ e } b = \begin{pmatrix} 3 \\ 18 \\ 12 \end{pmatrix} \text{ teremos,}$$

$$AX = b$$

Logo o problema variacional consiste em,

$$\text{minimize } \frac{1}{2} X^T H X$$

$$\text{sujeito aos vínculos } AX = b$$

Para transformar este problema em um outro sem vínculos, utilizemos a técnica dos multiplicadores de Lagrange,

tomando  $\lambda = \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{pmatrix}$ , e temos o problema de otimização quadrática,

$$\text{minimize } f \begin{pmatrix} X \\ \lambda \end{pmatrix} = \frac{1}{2} X^T H X + (AX - b)^T \lambda$$

Derivando em relação à  $X$  e  $\lambda$ , e igualando à 0, temos o sistema linear 8x8,

$$\begin{aligned} HX + A^T \lambda + 0 &= 0 \\ AX + 0\lambda - b &= 0 \end{aligned}$$

$$\begin{pmatrix} H & A^T \\ A & 0 \end{pmatrix} \begin{pmatrix} X \\ \lambda \end{pmatrix} = \begin{pmatrix} 0 \\ b \end{pmatrix}$$

resolvendo este sistema, em relação à  $X$  e  $\lambda$ , teremos,

$$\begin{aligned} X &= \begin{pmatrix} 3 & 41 & 8 & -80 & 40 \end{pmatrix}^T \\ \lambda &= \begin{pmatrix} 256 & -512 & 256 \end{pmatrix}^T \end{aligned}$$

e, no conjunto dos polinômios do quarto grau, aquele que satisfaz o problema de otimização proposto é dado por,

$$\gamma(t) = 3 + 41t + 8t^2 - 80t^3 + 40t^4$$

## 4.2 Resolvendo o problema de otimização na base das B-Wavelets

Considere o problema de encontrar a curva

$$\begin{aligned} \gamma : [0, 1] &\longrightarrow R \\ t &\longmapsto \gamma(t) \end{aligned}$$

satisfazendo as condições,

$$\text{minimize a energia de deformação } \int_0^1 |\gamma(t)''|^2 dt$$

$$\text{sujeita aos vínculos } \begin{cases} \gamma(t_1) = b_1 \in R \\ \vdots \\ \gamma(t_2) = b_2 \in R \\ \vdots \\ \gamma(t_m) = b_m \in R \end{cases}$$

Vamos resolver este problema usando o método de Ritz. A representação da curva  $\gamma(t)$  na base das B-Splines

$$\Phi^j = (\phi_0^j \cdots \phi_{2^j+d-1}^j)$$

de nível  $j$  de resolução é dada por,

$$\begin{aligned} \gamma(t) &= \sum_i X_i \phi_i^j(t) = \sum_i \phi_i^j(t) X_i = \\ &= (\phi_0^j(t) \cdots \phi_{2^j+d-1}^j(t)) \begin{pmatrix} X_0 \\ \vdots \\ X_{2^j+d-1} \end{pmatrix} = \Phi^j(t) X, \end{aligned}$$

pelo método dos multiplicadores de Lagrange, consideremos  $(\lambda_1 \cdots \lambda_m)^T$  e resolvamos o problema de otimização quadrática,

$$\text{minimize } f \begin{pmatrix} X \\ \lambda \end{pmatrix} = \frac{1}{2} X^T H X + (A X - b)^T \lambda$$

onde a matriz Hessiana  $H$  é dada por  $H_{ik} = \int \ddot{\Phi}_i^j(t) \ddot{\Phi}_k^j(t) dt$

e a matriz  $A$  é dada por  $A = \begin{pmatrix} \Phi^j(t_1) \\ \vdots \\ \Phi^j(t_m) \end{pmatrix} = \begin{pmatrix} \phi_0^j(t_1) & \cdots & \phi_{2^j+d-1}^j(t_1) \\ \vdots & \ddots & \vdots \\ \phi_0^j(t_m) & \cdots & \phi_{2^j+d-1}^j(t_m) \end{pmatrix}$

Isto nos leva a resolver o sistema de equações lineares,

$$\begin{pmatrix} H & A^T \\ A & 0 \end{pmatrix} \begin{pmatrix} X \\ \lambda \end{pmatrix} = \begin{pmatrix} 0 \\ b \end{pmatrix} \text{ em relação a base } \Phi^j$$

Com o propósito de melhorar o número de condicionamento da matriz deste sistema, vamos considerar o problema de otimização na base  $(\Phi^{j-k} \Psi^{j-k} \dots \Psi^{j-1})$ .

**Proposição 4.2.1** *Considere no nível  $j$  de resolução, as coordenadas de*

$$X = C^j = (c_0^j \cdots c_{2^j+d-1}^j)^T$$

*em relação a base  $\Phi^j$  das B-Splines de grau  $d$ , e as coordenadas em relação as B-Splines e B-Wavelets do nível  $j-2$  de resolução,*

$$\bar{X} = \begin{pmatrix} C^{j-2} \\ D^{j-2} \\ D^{j-1} \end{pmatrix}$$

onde

$$C^{j-2} = (c_0^{j-2} \cdots c_{2^{j-2}+d-1}^{j-2})^T$$

$$D^{j-2} = (d_0^{j-2} \cdots d_{2^{j-2}-1}^{j-2})^T$$

$$D^{j-1} = (d_0^{j-1} \cdots d_{2^{j-1}-1}^{j-1})^T$$

então, considerando matrizes em bloco, temos a relação entre  $X$  e  $\bar{X}$

$$X = (H^j H^{j-1} H^j G^{j-1} G^j) \bar{X}$$

*Demonstração: Como*

$$X = C^j = (H^j G^j) \begin{pmatrix} C^{j-1} \\ D^{j-1} \end{pmatrix}$$

e

$$C^{j-1} = (H^{j-1} G^{j-1}) \begin{pmatrix} C^{j-2} \\ D^{j-2} \end{pmatrix}$$

Daí,

$$\begin{aligned} X &= H^j C^{j-1} + G^j D^{j-1} = H^j (H^{j-1} C^{j-2} + G^{j-1} D^{j-2}) + G^j D^{j-1} = \\ &= H^j H^{j-1} C^{j-2} + H^j G^{j-1} D^{j-2} + G^j D^{j-1} \end{aligned}$$

Logo,

$$X = \begin{pmatrix} H^j H^{j-1} & H^j G^{j-1} & G^j \end{pmatrix} \begin{pmatrix} C^{j-2} \\ D^{j-2} \\ D^{j-1} \end{pmatrix}$$

**Proposição 4.2.2** Na base  $(\Phi^{j-k} \Psi^{j-k} \Psi^{j-k+1} \dots \Psi^{j-1})$  para  $k$  níveis de resolução, onde  $1 \leq k \leq j$ , o problema de otimização é resolvido pelo sistema de equações lineares,

$$\begin{pmatrix} \bar{H} & \bar{A}^T \\ \bar{A} & 0 \end{pmatrix} \begin{pmatrix} \bar{X} \\ \lambda \end{pmatrix} = \begin{pmatrix} 0 \\ b \end{pmatrix}$$

onde

$$\bar{H} = (M^{j,k})^T H M^{j,k}$$

e

$$\bar{A} = A M^{j,k}$$

Depois de resolvido este sistema em relação a  $\bar{X}$ , retorna-se a base  $\Phi^j$  usando,

$$X = M^{j,k} \bar{X}$$

considerando a matriz em blocos  $M^{j,k}$  gerada pela aplicação progressiva dos filtros  $(H^{j-k} \ G^{j-k})$ ,  $0 \leq k \leq j$ , como mostramos abaixo para  $k = 0, 1, 2$ :

$$M^{j,0} = \begin{pmatrix} H^j & G^j \end{pmatrix}$$

$$M^{j,1} = \begin{pmatrix} H^j H^{j-1} & H^j G^{j-1} & G^j \end{pmatrix}$$

$$M^{j,2} = \begin{pmatrix} H^j H^{j-1} H^{j-2} & H^j H^{j-1} G^{j-2} & H^j G^{j-1} & G^j \end{pmatrix}$$

*Demonstração:* como  $X = M^{j,k} \bar{X}$ , temos

$$\begin{aligned} f \begin{pmatrix} X \\ \lambda \end{pmatrix} &= \frac{1}{2} X^T H X + (A X - b)^T \lambda = \frac{1}{2} (M^{j,k} \bar{X})^T H (M^{j,k} \bar{X}) + (A (M^{j,k} \bar{X}) - b)^T \lambda = \\ &= \frac{1}{2} (\bar{X})^T (M^{j,k} H M^{j,k}) \bar{X} + ((A M^{j,k}) \bar{X} - b)^T \lambda = 0 \end{aligned}$$

Derivando em relação a  $\bar{X}$  e  $\lambda$  e igualando a zero, temos o sistema de equações lineares da proposição.

**Proposição 4.2.3** Na proposição anterior considerando  $k = 0$  os produtos,

$$\bar{H} = (M^{j,0})^T H M^{j,0} = M^{j,0} = (H^j \ G^j)^T H (H^j \ G^j)$$

e

$$\bar{A} = A M^{j,0} = A (H^j \ G^j)$$

são tais que  $\widehat{H} = H (H^j \ G^j)$  e  $A (H^j \ G^j)$  representam a aplicação de um filtro passa-baixa  $H^j$  e passa-alta  $G^j$  em cada linha das matrizes  $H$  e  $A$  respectivamente e  $(H^j \ G^j)^T \widehat{H}$  representa a aplicação dos filtros  $H^j$  e  $G^j$  sobre cada coluna de  $\widehat{H}$ .

*Demonstração:* Basta considerar a decomposição das matrizes  $H$  e  $A$  em blocos e efetuar os produtos,

$$\widehat{H} = \begin{pmatrix} H^1 \\ \vdots \\ H^{2^j+d} \end{pmatrix} (H^j \ G^j) = \begin{pmatrix} H^1 H^j & H^1 G^j \\ \vdots & \vdots \\ H^{2^j+d} H^j & H^{2^j+d} G^j \end{pmatrix}$$

$$\bar{A} = \begin{pmatrix} A^1 \\ \vdots \\ A^{2^j+d} \end{pmatrix} (H^j \ G^j) = \begin{pmatrix} A^1 H^j & A^1 G^j \\ \vdots & \vdots \\ A^{2^j+d} H^j & A^{2^j+d} G^j \end{pmatrix}$$

e,

$$\begin{aligned} \bar{H} &= (H^j \ G^j)^T \widehat{H} = \begin{pmatrix} (H^j)^T \\ (G^j)^T \end{pmatrix} \begin{pmatrix} \widehat{H}_1 & \cdots & \widehat{H}_{2^j+d} \end{pmatrix} = \\ &= \begin{pmatrix} (H^j)^T \widehat{H}_1 & \cdots & (H^j)^T \widehat{H}_{2^j+d} \\ (G^j)^T \widehat{H}_1 & \cdots & (G^j)^T \widehat{H}_{2^j+d} \end{pmatrix} \end{aligned}$$

onde  $d$  é o grau da B-Spline. Depois da mudança de base, obtida aplicando os filtros passa-baixa  $H^j$  e passa-alta  $G^j$ , a matriz  $\bar{H}$ , passa ter a forma,

$$\begin{pmatrix} \text{Low\_Low} & \text{High\_low} \\ \text{Low\_High} & \text{High\_High} \end{pmatrix}$$

### 4.3 Aplicação

Consideraremos o experimento apresentado em [32], de ajuste de uma curva B-Spline com  $2^6 + 1 = 65$  pontos de controle, onde se usa B-Splines cúbicas bi-ortogonais definidas sobre intervalo com extensão simétrica e periódica dos pontos de controle nos bordos do intervalo.

As mudanças de níveis são feitas considerando a proposição 4.2.3 com os filtros bi-ortogonais estendidos nos bordos, dados por,

$$h[0..4] = \frac{1}{8} \times \{1, 4, 6, 4, 1\}$$

$$g[0..10] = \frac{1}{256} x \{5, 20, 1, -96, -70, 280, -70, -96, 1, 20, 5\}$$

$$\tilde{h}[-3..7] = \frac{1}{256} \times \{-5, 20, -1, -96, 70, 280, 70, -96, -1, 20, -5\}$$

$$\tilde{g}[3..7] = \frac{1}{8} \times \{1, -4, 6, -4, 1\}$$

Gortler considera o nível máximo de resolução  $L = 6$ , com  $2^6 + 1 = 65$  B-splines. Como temos 3 vínculos, teremos um total de  $65 + 3 = 68$  variáveis.

A matriz que resolve o problema é dada por,

$$M = \begin{pmatrix} \bar{H} & \bar{A}^T \\ \bar{A} & 0 \end{pmatrix}$$

e como é necessário que ela seja positiva definida, considera o problema equivalente de resolver o sistema,

$$\begin{pmatrix} \bar{H} & \bar{A}^T \\ \bar{A} & 0 \end{pmatrix}^T \begin{pmatrix} \bar{H} & \bar{A}^T \\ \bar{A} & 0 \end{pmatrix} \begin{pmatrix} \bar{X} \\ \lambda \end{pmatrix} = \begin{pmatrix} \bar{H} & \bar{A}^T \\ \bar{A} & 0 \end{pmatrix}^T \begin{pmatrix} 0 \\ b \end{pmatrix}$$

pois de acordo com [37], se  $M$  é inversível, então

$$M^T M \text{ é simétrica e positiva definida.}$$

O sistema acima tem um número de condicionamento que é o quadrado do original e isto influencia o número de iterações necessárias para determinar a solução, perfazendo um total de 1024. Considerando o sistema no nível 0 de resolução, com as B-Wavelets definidas na reta, que é obtido filtrando a submatriz *Low-Low* de  $\bar{A}$  e  $\bar{H}$ , notamos que são apenas necessárias 64 iterações.

Desta forma Gortler, usando a base B-Spline, vide 4.1, com B-Wavelets cúbicas *bi-ortogonais, com extensão simétrica e periódica para fora do intervalo*, cuja FWT é dada pelos filtros de análise mostrado na figura 4.2e a IWT é dada pelos filtros de síntese apresentado na figura 4.3, obtem os resultados que podem ser vistos nas ilustrações 4.4e 4.5.

Foi feita uma simulação análoga considerando as B-splines cúbicas *semi-ortogonais sobre intervalos*, onde próximo as bordas os filtros matriciais foram

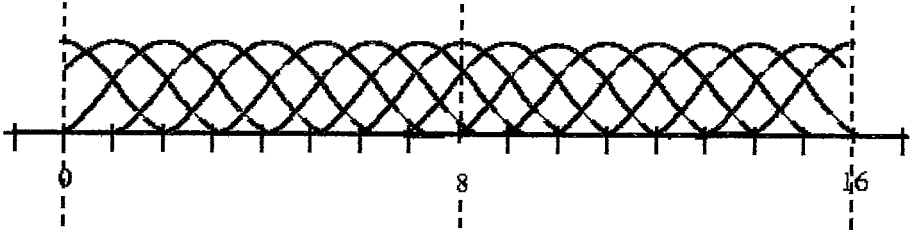


Figura 4.1: Base com  $16+1=17$  B-Splines cúbicas sobre o intervalo  $[0,16]$

*adaptados*, pelo algoritmo de Inserção de Nós de W. Boehm, conforme os valores das máscaras representadas pelas linhas ou colunas, próximas as laterais das matrizes de análise e síntese.

Observou-se que em um problema análogo de interpolar 3 pontos do plano por uma curva B-Spline cúbica, temos um menor número de iterações na aplicação do Algoritmo do Gradiente Conjugado quanto maior for a quantidade de wavelets empregadas.

Assim, no nível 6 de resolução sem wavelets, temos 464 iterações, na base com o total de  $2^6 + 3 = 67$  B-Splines cúbicas:

$$\Phi^6 = (\phi_0^6 \cdots \phi_{66}^6)$$

No nível 3 de resolução, temos 35 iterações, na base com  $2^3 + 3 = 11$  B-Splines e  $67 - (2^3 + 3) = 56$  B-Wavelets:

$$(\phi^3 \quad \psi^3 \quad \psi^4 \quad \psi^5 \quad \psi^6)$$

No nível 0 de resolução, temos 25 iterações, na base com  $2^0 + 3 = 4$  B-Splines e  $67 - (2^0 + 3) = 63$  B-Wavelets:

$$(\phi^0 \quad \psi^0 \quad \psi^1 \quad \psi^2 \quad \psi^3 \quad \psi^4 \quad \psi^5 \quad \psi^6)$$

Isto pode ser visto nas figuras, 4.6 , 4.7 , 4.8 e 4.9 .

Temos assim uma solução *matricial com adaptação nos extremos do intervalo que evita os erros* nos bordos obtidos por extensões do filtro tal como foi utilizado por Gortler.

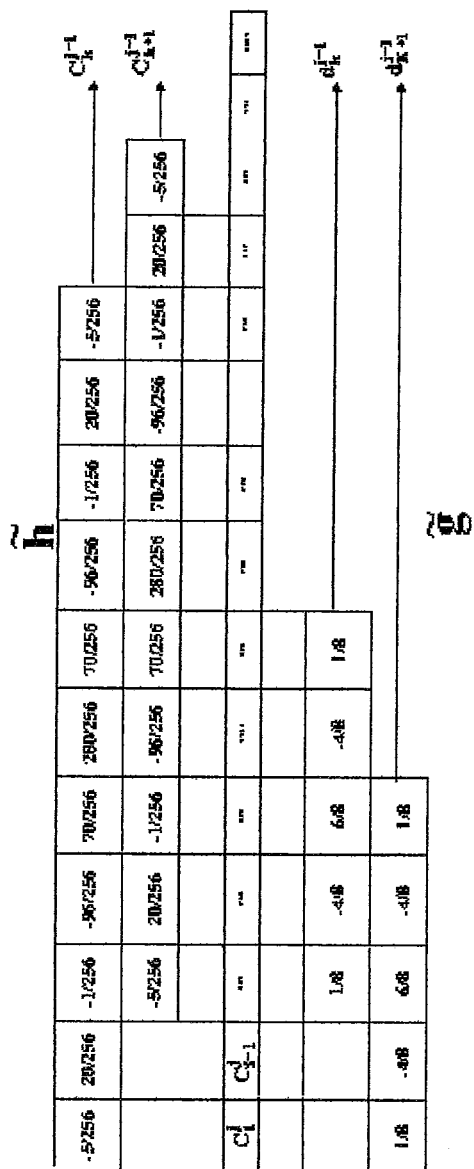


Figura 4.2: Filtro de análise. Convolução com decimação.



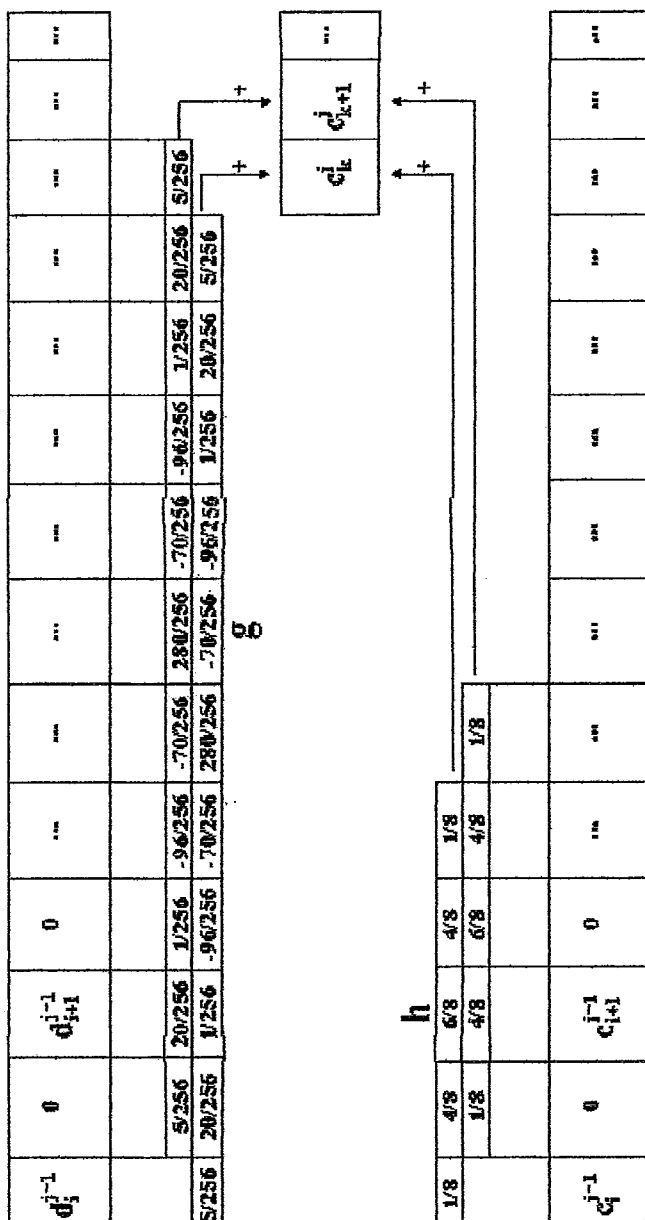


Figura 4.3: Filtro de síntese. Interpolação com convolução.

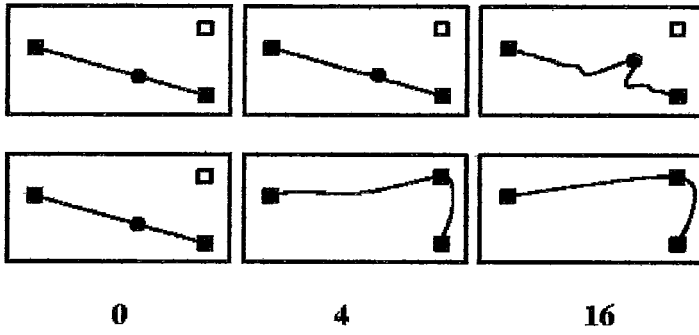


Figura 4.4: Nível 6 de resolução com B-Splines (curva superior ) e o nível 0 com B-Wavelets (curva inferior).Iterações 0, 4 e 16.

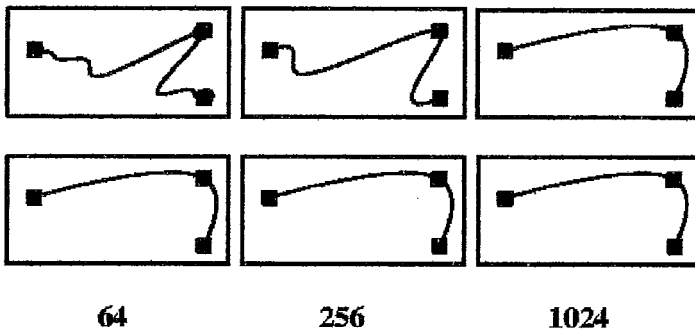


Figura 4.5: Nível 6 de resolução com B-Splines (curva superior ) e o nível 0 com B-Wavelets (curva inferior). Iterações 64, 256 e 1024

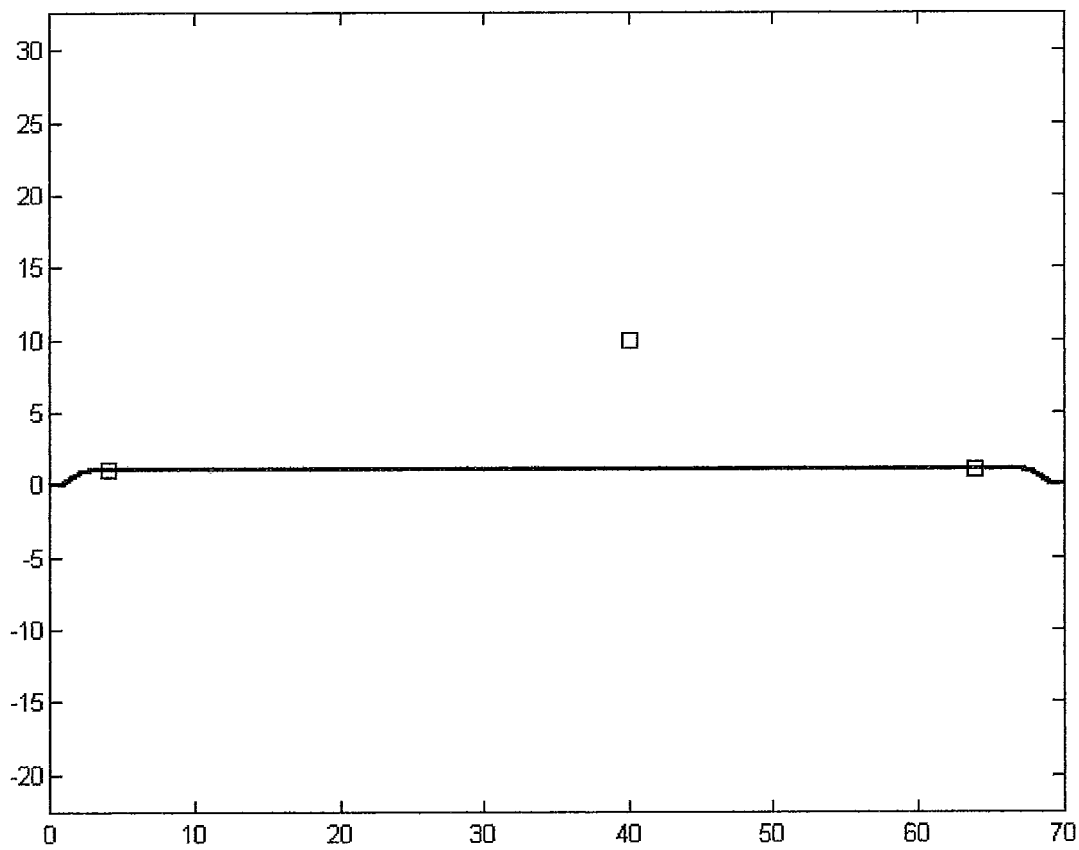


Figura 4.6: Posição inicial da curva

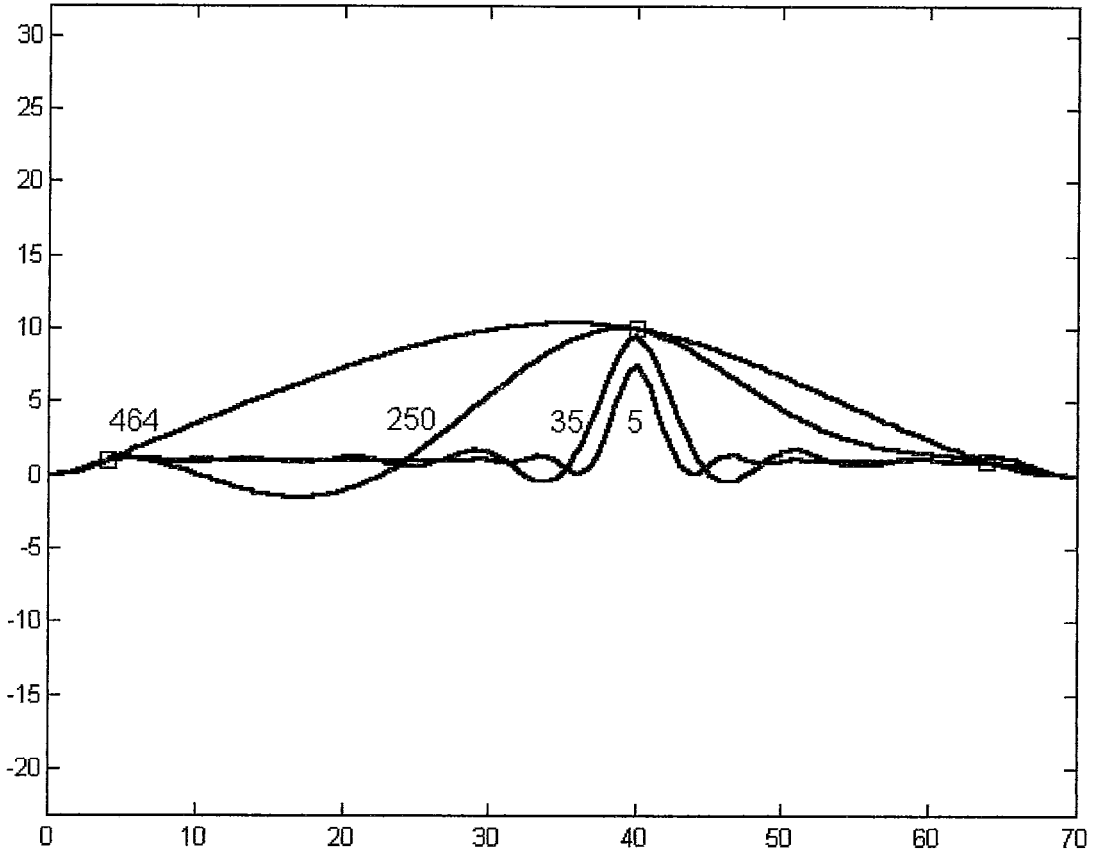


Figura 4.7: Interpolação dos três pontos, na base B-Spline cúbica no nível 6, com 464 iterações

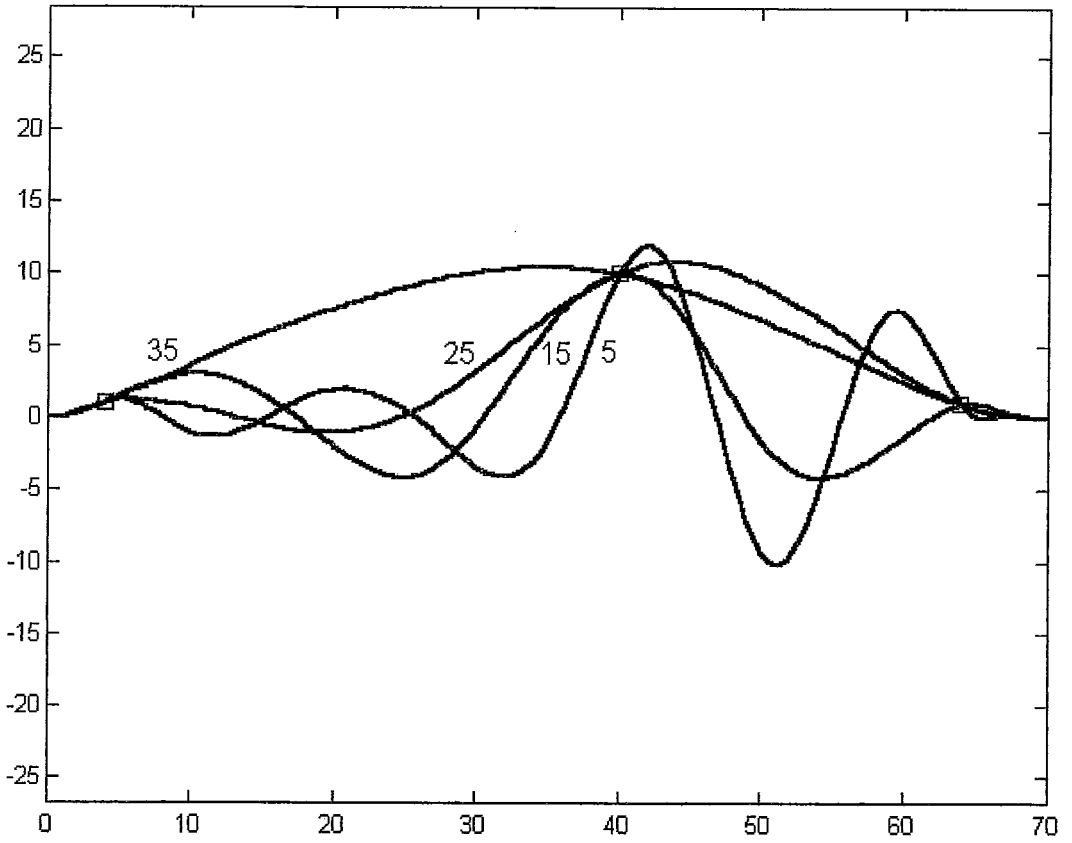


Figura 4.8: Interpolação dos três pontos, na base com B-Wavelets cúbicas no nível 3, com 35 iterações

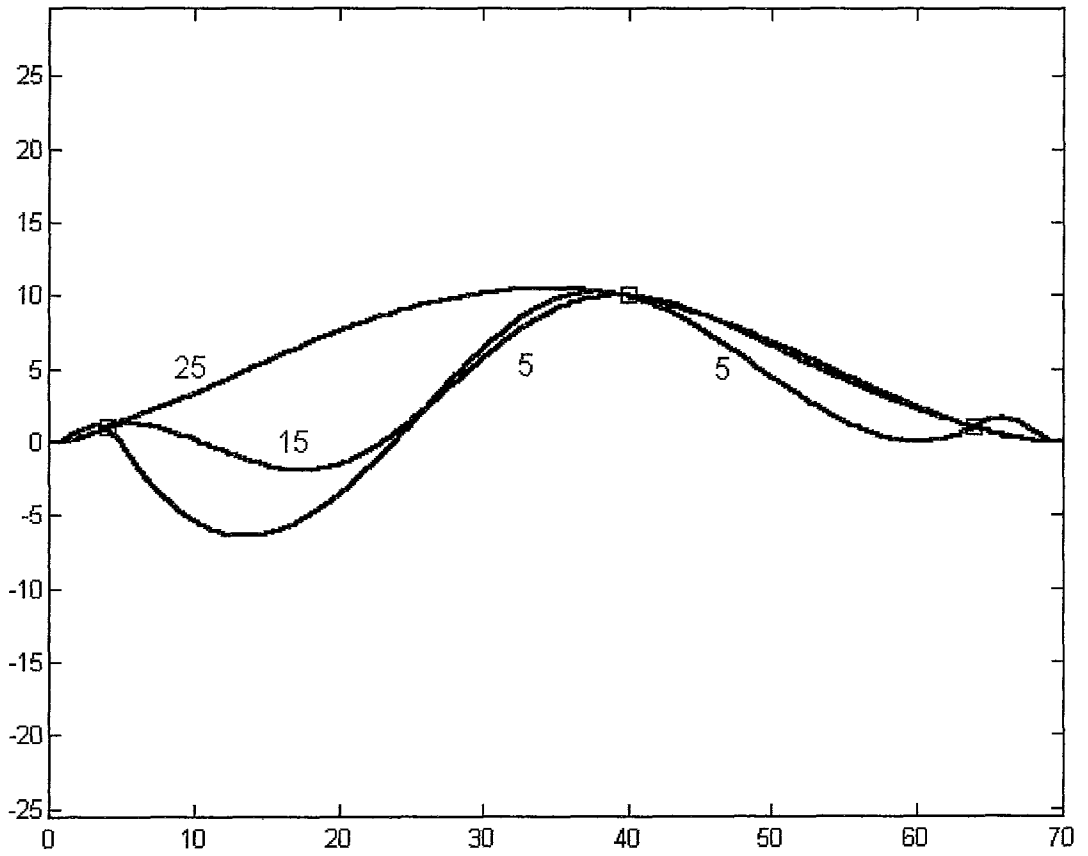


Figura 4.9: Interpolação dos três pontos, na base com B-Wavelets cúbicas no nível 0, com 25 iterações

# Capítulo 5

## Aplicação das B-Wavelets tensoriais em filtragem de imagens em multiresolução

Neste capítulo, *nossa contribuição* se refere a apresentação de uma generalização de filtros wavelets para imagens, *construídos por produto tensoriais de filtros unidimensionais “distintos”, junto com o conceito de nível de resolução  $(j, i)$  em relação a (largura, altura) da imagem*, o que possibilitará uma análise em multi-resolução ao longo das linhas (colunas) e posteriormente ao longo das colunas (linhas) por diferentes filtros unidimensionais. Estes resultados são particularizados para as B-wavelets e aplicados na análise em multi-resolução das imagens de Lena e Radon.

Detalhes relativos a resultados de álgebra linear podem ser encontrados em [37] e sobre produto tensoriais em [38].

### 5.1 Matriz associada ao operador linear na base das wavelets

**Definição 5.1.1** *Considere  $\bar{V}^j$  e  $V^i$  espaços vetoriais de dimensões  $m$  e  $n$  respectivamente. Se  $T : \bar{V}^j \rightarrow V^i$  é um operador linear e*

$$\bar{\Phi}^j = (\bar{\phi}_0^j \dots \bar{\phi}_{m-1}^j)$$

*é uma base de  $\bar{V}^j$  então por definição,*

$$T(\bar{\Phi}^j) := (T(\bar{\phi}_0^j) \dots T(\bar{\phi}_{m-1}^j))$$

**Definição 5.1.2** Considere  $\bar{V}^j$  e  $V^i$  espaços vetoriais de dimensões  $m$  e  $n$  respectivamente. Se  $T : \bar{V}^j \rightarrow V^i$  é um operador linear, denominamos matriz no nível  $(j, i)$  de resolução em relação a (largura, altura), associada ao operador  $T$  considerando as bases,  $\bar{\Phi}^j$  de  $\bar{V}^j$  no nível  $j$  de resolução e  $\Phi^i$  de  $V^i$  no nível  $i$  de resolução, a matriz denotada por  $[T]_{\bar{\Phi}^j, \Phi^i} = (T_{rs})$  tal que,

$$\begin{pmatrix} T(\bar{\phi}_0^j) \\ \vdots \\ T(\bar{\phi}_{m-1}^j) \end{pmatrix}^T = \begin{pmatrix} T_{00} & \cdots & T_{0n-1} \\ \vdots & \ddots & \vdots \\ T_{m-10} & \cdots & T_{m-1n-1} \end{pmatrix} \begin{pmatrix} \bar{\phi}_0^j \\ \vdots \\ \bar{\phi}_{m-1}^j \end{pmatrix}^T$$

ou tomando a transposta,

$$T(\bar{\Phi}^j) = \bar{\Phi}^j [T]_{\bar{\Phi}^j, \Phi^i}$$

**Proposição 5.1.1** Se  $f \in \bar{V}^j$  então

$$[T(f)]_{\Phi^i} = [T]_{\bar{\Phi}^j, \Phi^i} [f]_{\bar{\Phi}^j}$$

*Demonstração: como*

$$f = \bar{\Phi}^j \bar{C}^j = \sum_{r=0}^{m-1} \bar{c}_r^j \bar{\phi}_r^j$$

daí, aplicando o operador  $T$ ,

$$T(f) = \sum_{r=0}^{m-1} \bar{c}_r^j T(\bar{\phi}_r^j) = T(\bar{\Phi}^j) \bar{C}^j = \Phi^i [T]_{\bar{\Phi}^j, \Phi^i} [f]_{\bar{\Phi}^j}$$

logo,

$$[T(f)]_{\Phi^i} = [T]_{\bar{\Phi}^j, \Phi^i} [f]_{\bar{\Phi}^j}$$

### 5.1.1 Representação matricial de um operador linear em níveis distintos de resolução

**Proposição 5.1.2** Considere as bases  $(\bar{\Phi}^{j-1} \bar{\Psi}^{j-1})$  e  $\bar{\Phi}^j$  de níveis de resolução  $j-1$  e  $j$ , respectivamente para  $\bar{V}^j$ . Temos assim a relação entre estas bases e as respectivas bases duais, dadas por,

$$(\bar{\Phi}^{j-1} \bar{\Psi}^{j-1}) = \bar{\Phi}^j (\bar{H}^j \bar{G}^j)$$

e

$$(\tilde{\Phi}^{j-1} \tilde{\Psi}^{j-1}) = \tilde{\Phi}^j (\tilde{H}^j \tilde{G}^j)$$



onde

$$\left(\widetilde{H}^j \widetilde{G}^j\right)^T = \left(\overline{H}^j \overline{G}^j\right)^{-1}$$

Considere também, as bases  $(\Phi^{i-1} \Psi^{i-1})$  e  $\Phi^i$  de níveis de resolução  $i - 1$  e  $i$ , respectivamente para  $V^i$ . Temos assim a relação entre estas bases e as respectivas bases duais, dadas por,

$$\left(\Phi^{i-1} \Psi^{i-1}\right) = \Phi^i \left(H^i G^i\right)$$

e

$$\left(\widetilde{\Phi}^{i-1} \widetilde{\Psi}^{i-1}\right) = \widetilde{\Phi}^i \left(\widetilde{H}^i \widetilde{G}^i\right)$$

onde

$$\left(\widetilde{H}^i \widetilde{G}^i\right)^T = \left(H^i G^i\right)^{-1}$$

Se temos a representação matricial de  $T$  nas bases  $\overline{\Phi}^j$  e  $\Phi^i$  dada por  $[T]_{\overline{\Phi}^j, \Phi^i}$  no nível  $(j, i)$  de resolução e a representação de  $T$  nas bases  $(\overline{\Phi}^{j-1} \overline{\Psi}^{j-1})$  e  $(\Phi^{i-1} \Psi^{i-1})$  dada por  $[T]_{(\overline{\Phi}^{j-1} \overline{\Psi}^{j-1}), (\Phi^{i-1} \Psi^{i-1})}$  no nível  $(j - 1, i - 1)$  de resolução em relação a (largura, altura), então as duas representações satisfazem a relação de semelhança dada por,

$$[T]_{(\overline{\Phi}^{j-1} \overline{\Psi}^{j-1}), (\Phi^{i-1} \Psi^{i-1})} = \begin{pmatrix} \left(\widetilde{H}^i\right)^T \\ \left(\widetilde{G}^i\right)^T \end{pmatrix} [T]_{\overline{\Phi}^j, \Phi^i} \left(\overline{H}^j \overline{G}^j\right)$$

Demonstração: por 5.1.1 temos,

$$[T(f)]_{\Phi^i} = [T]_{\overline{\Phi}^j, \Phi^i} [f]_{\overline{\Phi}^j}$$

e

$$[T(f)]_{(\Phi^{i-1} \Psi^{i-1})} = [T]_{(\overline{\Phi}^{j-1} \overline{\Psi}^{j-1}), (\Phi^{i-1} \Psi^{i-1})} [f]_{(\overline{\Phi}^{j-1} \overline{\Psi}^{j-1})}$$

da segunda igualdade e usando 3.2.4 temos,

$$\begin{pmatrix} \left(\widetilde{H}^i\right)^T \\ \left(\widetilde{G}^i\right)^T \end{pmatrix} [f]_{\Phi^i} = [T]_{(\overline{\Phi}^{j-1} \overline{\Psi}^{j-1}), (\Phi^{i-1} \Psi^{i-1})} \begin{pmatrix} \left(\widetilde{H}^j\right)^T \\ \left(\widetilde{G}^j\right)^T \end{pmatrix} [f]_{\overline{\Phi}^j}$$

logo,

$$[f]_{\Phi^i} = \left[ \begin{pmatrix} \left(\widetilde{H}^i\right)^T \\ \left(\widetilde{G}^i\right)^T \end{pmatrix}^{-1} [T]_{(\overline{\Phi}^{j-1} \overline{\Psi}^{j-1}), (\Phi^{i-1} \Psi^{i-1})} \begin{pmatrix} \left(\widetilde{H}^j\right)^T \\ \left(\widetilde{G}^j\right)^T \end{pmatrix} \right] [f]_{\overline{\Phi}^j}$$

donde, comparando com a primeira igualdade,

$$[T]_{\bar{\Phi}^j, \bar{\Phi}^i} = \left( \begin{pmatrix} (\widetilde{H}^i)^T \\ (\widetilde{G}^i)^T \end{pmatrix} \right)^{-1} [T]_{(\bar{\Phi}^{j-1}, \bar{\Psi}^{j-1}), (\bar{\Phi}^{i-1}, \bar{\Psi}^{i-1})} \left( \begin{pmatrix} (\widetilde{H}^j)^T \\ (\widetilde{G}^j)^T \end{pmatrix} \right)$$

daí,

$$[T]_{(\bar{\Phi}^{j-1}, \bar{\Psi}^{j-1}), (\bar{\Phi}^{i-1}, \bar{\Psi}^{i-1})} = \left( \begin{pmatrix} (\widetilde{H}^i)^T \\ (\widetilde{G}^i)^T \end{pmatrix} \right) [T]_{\bar{\Phi}^j, \bar{\Phi}^i} \left( \begin{pmatrix} (\widetilde{H}^j)^T \\ (\widetilde{G}^j)^T \end{pmatrix} \right)^{-1}$$

Usando a relação 3.3 temos a proposição.

### 5.1.2 Representação tensorial de um operador linear em níveis distintos de resolução

**Definição 5.1.3** Considere as matrizes,  $A = (a_{rs}) \in M_{n_1 \times m_1}$  e  $B = (b_{rs}) \in M_{n_2 \times m_2}$ . Definimos o produto tensorial ou de Kronecker de  $A$  por  $B$ , pela matriz  $A \otimes B \in M_{(n_1 \times n_2) \times (m_1 \times m_2)}$  da seguinte forma,

$$A \otimes B = \begin{pmatrix} a_{11}B & \cdots & a_{1m_1}B \\ \vdots & \ddots & \vdots \\ a_{n_11}B & \cdots & a_{n_1m_1}B \end{pmatrix}$$

#### Propriedades do produto tensorial:

Para as matrizes  $A, B, C, D$  e o real  $c$ , temos,

1.  $A \otimes B \neq B \otimes A$
2.  $(A + B) \otimes C = A \otimes C + B \otimes C$
3.  $(A \otimes B) \otimes C = A \otimes (B \otimes C)$
4.  $(A \otimes B)^T = A^T \otimes B^T$
5.  $(A \otimes B)^{-1} = A^{-1} \otimes B^{-1}$
6.  $(A \otimes B)(C \otimes D) = (AC) \otimes (CD)$

**Definição 5.1.4** Denominamos de representação tensorial do operador linear  $T : \bar{V}^j \mapsto V^i$  definido por

$$T \left( \bar{\phi}_r^j \right) = \sum_s c_{sr} \phi_s^i$$

a expressão dada por,

$$T = \sum_{s=0, r=0}^{2^i+d-1, 2^j+\bar{d}-1} c_{sr} \tilde{\phi}_s^i \otimes \bar{\phi}_r^j$$

onde

$$\left( \tilde{\phi}_s^i \otimes \bar{\phi}_r^j \right) (t) = \tilde{\phi}_s^i (t) \bar{\phi}_r^j (t)$$

ou matricialmente,

$$T = \left( \tilde{\Phi}^i \otimes \bar{\Phi}^j \right) C$$

onde  $C$  é o vetor coluna

$$C = (c_{00} \cdots c_{0m-1} \cdots c_{n-10} \cdots c_{n-1m-1})^T$$

e

$$n = 2^i + d \text{ e } m = 2^j + \bar{d}$$

**Proposição 5.1.3** Se no nível  $(j, i)$  de resolução temos a representação tensorial do operador linear  $T$  dado por,

$$T = \left( \tilde{\Phi}^i \otimes \bar{\Phi}^j \right) C$$

então no nível  $(j-1, i-1)$  de resolução temos a representação tensorial,

$$\begin{aligned} T = & \left( \tilde{\Phi}^{i-1} \otimes \bar{\Phi}^{j-1} \right) \left( (H^i)^T \otimes \left( \widetilde{H}^j \right)^T \right) C + \left( \tilde{\Phi}^{i-1} \otimes \bar{\Psi}^{j-1} \right) \left( (H^i)^T \otimes \left( \widetilde{G}^j \right)^T \right) C + \\ & + \left( \tilde{\Psi}^{i-1} \otimes \bar{\Phi}^{j-1} \right) \left( (G^i)^T \otimes \left( \widetilde{H}^j \right)^T \right) C + \left( \tilde{\Psi}^{i-1} \otimes \bar{\Psi}^{j-1} \right) \left( (G^i)^T \otimes \left( \widetilde{G}^j \right)^T \right) C \end{aligned} \quad (5.1)$$

onde, temos os filtros passa-baixa e passa-alta de nível  $i$ ,  $H^i$  e  $G^i$ , respectivamente e os filtros passa-baixa e passa-alta de nível  $j$ ,  $\widetilde{H}^j$  e  $\widetilde{G}^j$  respectivamente.

Demonstração: da proposição 3.2.1, para o nível de resolução  $j$  temos

$$\bar{\Phi}^j = \left( \bar{\Phi}^{j-1} \bar{\Psi}^{j-1} \right) \begin{pmatrix} \left( \widetilde{H}^j \right)^T \\ \left( \widetilde{G}^j \right)^T \end{pmatrix}$$

e da proposição, 3.2.2, para o nível de resolução  $i$  temos,

$$\tilde{\Phi}^i = \left( \tilde{\Phi}^{i-1} \quad \tilde{\Psi}^{i-1} \right) \begin{pmatrix} (H^i)^T \\ (G^i)^T \end{pmatrix}$$

Operando em blocos e usando as propriedades do produto tensorial,

$$\begin{aligned} T &= \left( \tilde{\Phi}^i \otimes \bar{\Phi}^j \right) C = \\ &= \left( \left( \tilde{\Phi}^{i-1} \quad \tilde{\Psi}^{i-1} \right) \begin{pmatrix} (H^i)^T \\ (G^i)^T \end{pmatrix} \right) \otimes \left( \left( \bar{\Phi}^{j-1} \quad \bar{\Psi}^{j-1} \right) \begin{pmatrix} (\widetilde{H}^j)^T \\ (\widetilde{G}^j)^T \end{pmatrix} \right) C = \\ &= \left( \left( \tilde{\Phi}^{i-1} \quad \tilde{\Psi}^{i-1} \right) \otimes \left( \bar{\Phi}^{j-1} \quad \bar{\Psi}^{j-1} \right) \right) \left( \begin{pmatrix} (H^i)^T \\ (G^i)^T \end{pmatrix} \otimes \begin{pmatrix} (\widetilde{H}^j)^T \\ (\widetilde{G}^j)^T \end{pmatrix} \right) C = \\ &= \left( \tilde{\Phi}^{i-1} \otimes \bar{\Psi}^{j-1} \quad \tilde{\Phi}^{i-1} \otimes \bar{\Psi}^{j-1} \quad \tilde{\Psi}^{i-1} \otimes \bar{\Phi}^{j-1} \quad \tilde{\Psi}^{i-1} \otimes \bar{\Psi}^{j-1} \right) \begin{pmatrix} (H^i)^T \otimes (\widetilde{H}^j)^T \\ (H^i)^T \otimes (\widetilde{G}^j)^T \\ (G^i)^T \otimes (\widetilde{H}^j)^T \\ (G^i)^T \otimes (\widetilde{G}^j)^T \end{pmatrix} C. \end{aligned}$$

Fazendo agora o produto matricial em blocos destas três matrizes, temos a confirmação da proposição.

Desta forma, com a notação tensorial, no nível  $(j, i)$  de resolução se,

$$T = \sum_{s=0, r=0}^{2^i+d-1, 2^j+\bar{d}-1} c_{sr}^{j,i} \tilde{\phi}_s^i \otimes \bar{\phi}_r^j$$

então em relação ao nível  $(j-1, i-1)$  temos,

$$\begin{aligned} T &= \sum_{s=0, r=0}^{2^{i-1}+d-1, 2^{j-1}+\bar{d}-1} c_{sr}^{j-1, i-1} \tilde{\phi}_s^{i-1} \otimes \bar{\phi}_r^{j-1} + \sum_{s=0, r=2^{j-1}+\bar{d}}^{2^{i-1}+d-1, 2^j+\bar{d}-1} d_{sr}^{j-1, i-1} \tilde{\phi}_s^{i-1} \otimes \bar{\psi}_r^{j-1} + \\ &+ \sum_{s=2^{i-1}+d, r=0}^{2^i+d-1, 2^{j-1}+\bar{d}-1} d_{s,r}^{j-1, i-1} \tilde{\psi}_s^{i-1} \otimes \bar{\phi}_r^{j-1} + \sum_{s=2^{i-1}+d, r=2^{j-1}+\bar{d}}^{2^i+d-1, 2^j+\bar{d}-1} d_{s,r}^{j-1, i-1} \tilde{\psi}_s^{i-1} \otimes \bar{\psi}_r^{j-1} \end{aligned} \quad (5.2)$$

Assim, a representação matricial de  $T$ , no nível  $(j - 1, i - 1)$  de resolução, seria dada por

$$[T]_{(\overline{\Phi}^{j-1} \overline{\Psi}^{j-1}), (\Phi^{i-1} \Psi^{i-1})} = \begin{pmatrix} (c_{sr}^{j-1, i-1}) & (d_{sr}^{j-1, i-1}) \\ (d_{s,r}^{j-1, i-1}) & (d_{s,r}^{j-1, i-1}) \end{pmatrix} \quad (5.3)$$

Se  $Low^i$  e  $High^i$  representam o resultado das filtrações dos coeficientes  $C$  de  $T$  pelos filtros de nível  $i$  e  $Low^j$  e  $High^j$  representam o resultado das filtrações dos coeficientes  $C$  de  $T$  pelos filtros de nível  $j$  temos a matriz 5.3, na forma

$$[T]_{(\overline{\Phi}^{j-1} \overline{\Psi}^{j-1}), (\Phi^{i-1} \Psi^{i-1})} = \begin{pmatrix} Low^i Low^j & High^i Low^j \\ Low^i High^j & High^i High^j \end{pmatrix}$$

**Definição 5.1.5** Se temos a matriz  $C = (c_{sr}^{j,i})$ , definimos a sub-matriz  $Low^i Low^j = (c_{sr}^{j-1, i-1})$  como a representação dos coeficientes  $C$  no nível de resolução  $(j - 1, i - 1)$  e as demais sub-matrizes  $High^i Low^j = (d_{sr}^{j-1, i-1})$ ,  $Low^i High^j = (d_{s,r}^{j-1, i-1})$  e  $High^i High^j = (d_{s,r}^{j-1, i-1})$  como a representação dos detalhes de  $C$  no nível de resolução  $(j - 1, i - 1)$  em relação ao nível  $(j, i)$ .

## 5.2 Transformada discreta B-Wavelet bidimensional e tensorial

Tendo em vista as proposições 5.1.2 e 5.1.3 temos,

**Definição 5.2.1** Consideremos a matriz  $M \in M_{n \times m}$ , onde  $n = 2^i + d$  e  $m = 2^j + \bar{d}$ , sendo  $i$  e  $j$  os níveis de resolução do número de linhas (altura) e do número de colunas (largura) respectivamente e  $d$  e  $\bar{d}$  representam o grau das splines. Definimos a transformada discreta, denominada B-Wavelet bidimensional e tensorial, denotada por  $FWT2T(\cdot)$  (Forward Wavelet Transform 2-dimension Tensorial) pelo produto matricial,

$$FWT2T(M) = \begin{pmatrix} (\widetilde{H}^i)^T \\ (\widetilde{G}^i)^T \end{pmatrix} M \begin{pmatrix} \overline{H}^j & \overline{G}^j \end{pmatrix}$$

onde  $\widetilde{H}^i$  e  $\widetilde{G}^i$  são os filtros de análise passa-baixa e passa-alta do nível  $i$  de resolução das linhas de  $M$ , respectivamente e  $\overline{H}^j$  e  $\overline{G}^j$  são os filtros de síntese passa-baixa e passa-alta do nível  $j$  de resolução das colunas de  $M$ , respectivamente.

**Definição 5.2.2** Consideremos, como acima, a matriz  $M \in M_{n \times m}$ , onde  $n = 2^i + \bar{d}$  e  $m = 2^j + \bar{d}$ , sendo  $i$  e  $j$  os níveis de resolução do número de linhas (altura) e do número de colunas (largura) respectivamente e  $\bar{d}$  e  $\bar{d}$  representam o grau das splines. Definimos a transformada inversa discreta B-Wavelet bidimensional e tensorial, denotada por  $IWT2T(\cdot)$  (Inverse Wavelet Transform 2-dimension Tensorial) pelo produto matricial,

$$IWT2T(M) = \begin{pmatrix} H^i & G^i \end{pmatrix} M \begin{pmatrix} \left( \widetilde{H}^j \right)^T \\ \left( \widetilde{G}^j \right)^T \end{pmatrix}$$

onde  $\widetilde{H}^j$  e  $\widetilde{G}^j$  são os filtros de análise passa-baixa e passa-alta do nível  $j$  de resolução das colunas de  $M$ , respectivamente e  $H^i$  e  $G^i$  são os filtros de síntese passa-baixa e passa-alta do nível  $i$  de resolução das linhas de  $M$ , respectivamente.

**Proposição 5.2.1** A Transformada Discreta B-Wavelet Bidimensional e Tensorial,  $FWT2T(\cdot)$ , aplicada em uma matriz  $M$ , de nível de resolução  $(j - 1, i - 1)$  em relação a (largura, altura), pode ser calculada aplicando uma Transformada B-Wavelet unidimensional :

I- primeiro ao longo de todas as linhas da matriz  $M$ , com os filtros  $(\overline{H}^j \ \overline{G}^j)$  e posteriormente, sobre a matriz resultante, ao longo de todas as colunas com os filtros  $(\widetilde{H}^i \ \widetilde{G}^i)$ , ou

II- primeiro ao longo de todas as colunas da matriz  $M$  e posteriormente, sobre a matriz resultante, ao longo de todas as linhas.

*Demonstração:* em I, consideremos  $M^r$  a  $r$ -ésima linha e  $M_s$  a  $s$ -ésima coluna da matriz  $M$ . Operando em bloco sobre a matriz  $M$ , temos,

$$\begin{aligned} \begin{pmatrix} \left( \widetilde{H}^i \right)^T \\ \left( \widetilde{G}^i \right)^T \end{pmatrix} M \begin{pmatrix} \overline{H}^j & \overline{G}^j \end{pmatrix} &= \begin{pmatrix} \left( \widetilde{H}^i \right)^T \\ \left( \widetilde{G}^i \right)^T \end{pmatrix} \begin{pmatrix} M^1 \\ M^r \\ M^n \end{pmatrix} \begin{pmatrix} \overline{H}^j & \overline{G}^j \end{pmatrix} = \\ &= \begin{pmatrix} \left( \widetilde{H}^i \right)^T \\ \left( \widetilde{G}^i \right)^T \end{pmatrix} \begin{pmatrix} M^1 \overline{H}^j & M^1 \overline{G}^j \\ \vdots & \vdots \\ M^r \overline{H}^j & M^r \overline{G}^j \\ \vdots & \vdots \\ M^n \overline{H}^j & M^n \overline{G}^j \end{pmatrix} = \end{aligned}$$

$$\begin{aligned}
&= \begin{pmatrix} (\widetilde{H}^i)^T \\ (\widetilde{G}^i)^T \end{pmatrix} (\widehat{M}_1 \cdots \widehat{M}_s \cdots \widehat{M}_m) = \begin{pmatrix} (\widetilde{H}^i)^T \widehat{M}_1 \cdots (\widetilde{H}^i)^T \widehat{M}_s \cdots (\widetilde{H}^i)^T \widehat{M}_m \\ (\widetilde{G}^i)^T \widehat{M}_1 \cdots (\widetilde{G}^i)^T \widehat{M}_s \cdots (\widetilde{G}^i)^T \widehat{M}_m \end{pmatrix} = \\
&= \begin{pmatrix} Low^j\_Low^i & High^j\_Low^i \\ Low^j\_High^i & High^j\_High^i \end{pmatrix}
\end{aligned}$$

onde  $Low^j$ ,  $High^j$  representam respectivamente o resultado dos filtros passa-baixa e passa-alta de nível  $j$ , ao longo das linhas e  $Low^i$ ,  $High^i$  representam respectivamente o resultado dos filtros passa-baixa e passa-alta de nível  $i$ , ao longo das colunas da matriz  $M$ .

Em II, operando em bloco sobre a matriz  $M$ , temos,

$$\begin{aligned}
&\begin{pmatrix} (\widetilde{H}^i)^T \\ (\widetilde{G}^i)^T \end{pmatrix} M (\overline{H}^j \ \overline{G}^j) = \begin{pmatrix} (\widetilde{H}^i)^T \\ (\widetilde{G}^i)^T \end{pmatrix} (M_1 \cdots M_s \cdots M_m) (\overline{H}^j \ \overline{G}^j) = \\
&= \begin{pmatrix} (\widetilde{H}^i)^T M_1 \cdots (\widetilde{H}^i)^T M_s \cdots (\widetilde{H}^i)^T M_m \\ (\widetilde{G}^i)^T M_1 \cdots (\widetilde{G}^i)^T M_s \cdots (\widetilde{G}^i)^T M_m \end{pmatrix} (\overline{H}^j \ \overline{G}^j) = \\
&= \begin{pmatrix} \widehat{M}^1 \\ \widehat{M}^r \\ \widehat{M}^n \end{pmatrix} (\overline{H}^j \ \overline{G}^j) \\
&= \begin{pmatrix} \widehat{M}^1 \overline{H}^j & \widehat{M}^1 \overline{G}^j \\ \vdots & \vdots \\ \widehat{M}^r \overline{H}^j & \widehat{M}^r \overline{G}^j \\ \vdots & \vdots \\ \widehat{M}^n \overline{H}^j & \widehat{M}^n \overline{G}^j \end{pmatrix} = \\
&= \begin{pmatrix} Low^j\_Low^i & High^j\_Low^i \\ Low^j\_High^i & High^j\_High^i \end{pmatrix}
\end{aligned}$$

onde  $Low^j$ ,  $High^i$  representam respectivamente o resultado dos filtros passa-baixa e passa-alta de nível  $i$  para nível  $i - 1$ , ao longo das colunas e  $Low^j$ ,  $High^j$  representam respectivamente o resultado dos filtros passa-baixa e passa-alta de nível  $j$  para o nível  $j - 1$ , ao longo das linhas da matriz  $M$ .

### 5.3 Aplicação

Como aplicação do produto tensorial, consideremos  $H^i, G^i, \overline{H}^j, \overline{G}^j$  filtros de Haar unidimensionais, mais precisamente os filtros construídos a partir das funções escalas e wavelets modeladas pelas B-Splines de grau 0, e os 4 filtros bidimensionais gerado pelo produto tensorial dos mesmos e seus duais,

$$\begin{pmatrix} L^i L^j \\ L^i H^j \\ H^i L^j \\ H^i H^j \end{pmatrix} = \begin{pmatrix} (H^i)^T \otimes (\overline{H}^j)^T \\ (H^i)^T \otimes (\overline{G}^j)^T \\ (G^i)^T \otimes (\overline{H}^j)^T \\ (G^i)^T \otimes (\overline{G}^j)^T \end{pmatrix}$$

Assim, se  $M$  é o vetor coluna que representa a imagem de Lena com  $2^8 = 256$  linhas e  $2^7 = 128$  colunas, portanto no nível  $(j, i) = (7, 8)$  de resolução, pela proposição 5.1.3 a filtragem da imagem de Lena, figura 5.1, por filtros de Haar, seria dada pelas 4 sub-matrizes  $128 \times 64$  construídas a partir do vetor coluna,

$$\begin{pmatrix} LowLow \\ LowHigh \\ HighLow \\ HighHigh \end{pmatrix} = \begin{pmatrix} (L^8 L^7) M \\ (L^8 H^7) M \\ (H^8 L^7) M \\ (H^8 H^7) M \end{pmatrix}$$

cujas imagens podem ser vistas nas figuras 5.2, 5.3, 5.4, 5.5.

De acordo com a proposição 5.2.1 a FWT2T pode ser descrita pela filtragem das linhas seguida das filtragens das colunas, de tal forma que se

$$C = (c_{sr}^{j,i})$$

então,

$$FWT2T(C) = \begin{pmatrix} (c_{sr}^{j-1,i-1}) & (d_{sr}^{j-1,i-1}) \\ (d_{s,r}^{j-1,i-1}) & (d_{s,r}^{j-1,i-1}) \end{pmatrix}$$

conforme o esquema apresentado na figura 5.6.

Desta forma, considerando a filtragem ao longo das linhas e posteriormente ao longo das colunas pelos filtros de Haar, podemos filtrar a imagem de Lena, figura 5.1, do nível de resolução  $(7, 8)$ , usando o esquema acima 2 vezes, para o nível de resolução  $(5, 6)$ , e temos o resultado apresentado na figura 5.7.

Por outro lado, na figura 5.8 temos a imagem de Radon sendo filtrada





Figura 5.1: Imagem original de Lena com 256 linhas e 128 colunas



Figura 5.2: Decomposição LowLow da Imagem da Lena, usando o produto tensorial dos filtros de Haar.



Figura 5.3: Decomposicao HighLow da Imagem da Lena usando o produto tensorial dos filtros de Haar.



Figura 5.4: Decomposicao LowHigh da Imagem da Lena usando o produto tensorial dos filtros de Haar.

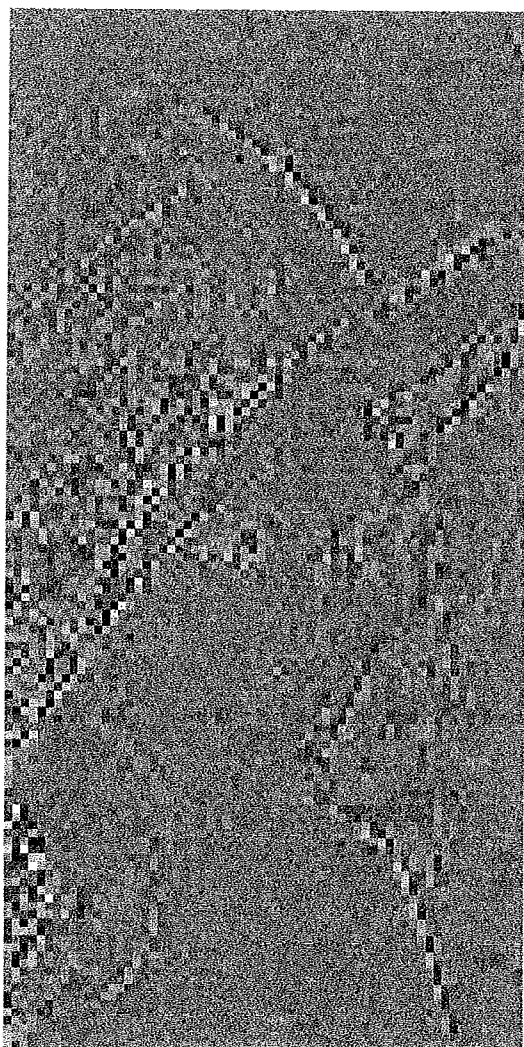


Figura 5.5: Decomposicao HighHigh da Imagem da Lena usando o produto tensorial dos filtros de Haar.

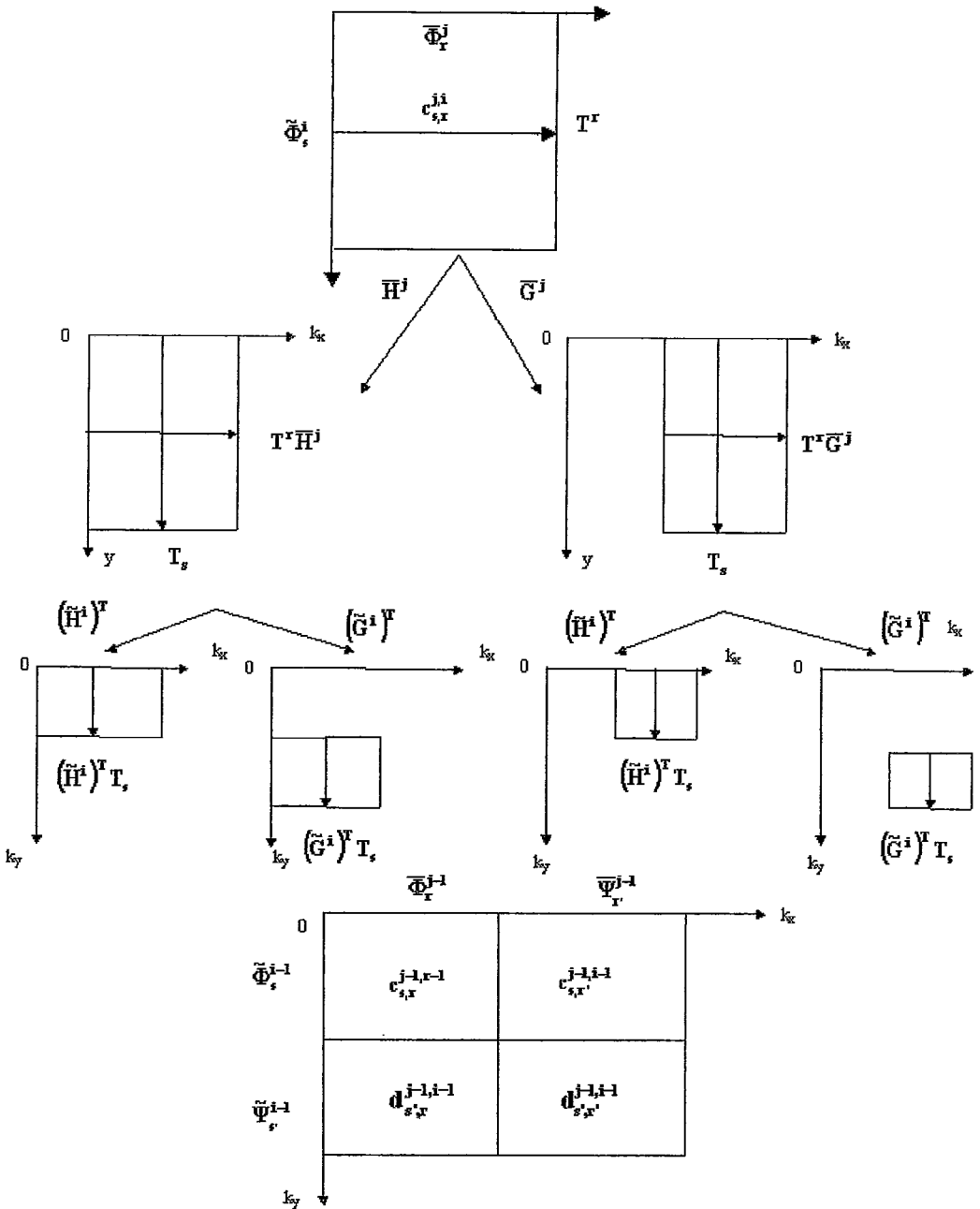


Figura 5.6: A transformada wavelet direta, bidimensional e tensorial



Figura 5.7: Filtragem da imagem de Lena com Haar, do nível (7,8) para o nível (5,6)

Imagem original de Radon no nível (8,8) de resolução

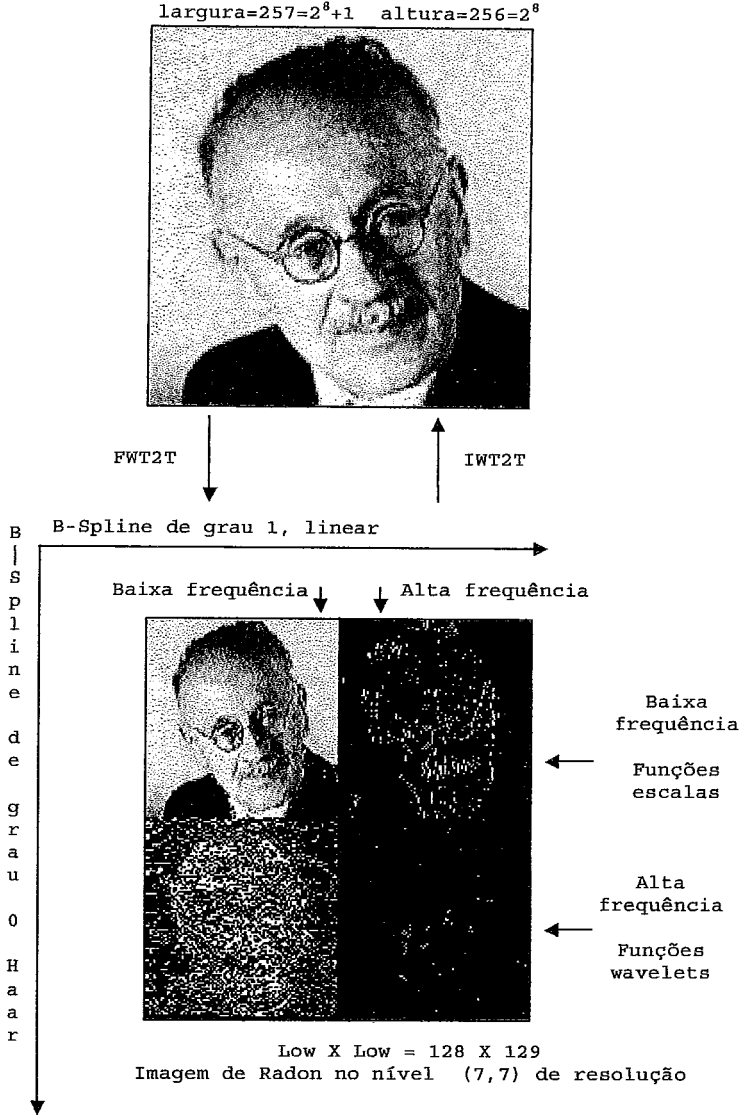


Figura 5.8: Filtragem da imagem de Radon com os filtros Linear e de Haar, do nível (8,8) para o nível (7,7)



por filtros B-Splines de grau 1 ( linear), passa-baixa e passa-alta, primeiro ao longo das linhas e depois por filtros B-Spline de grau 0 (Haar),passa-baixa e passa-alta, ao longo das colunas.

# Capítulo 6

## Aplicação das B-Wavelets em reconstrução de imagens

A aplicação das wavelets em reconstrução de imagens se torna evidente, quando a região a ser reconstruída é iluminada por :

1. pulsos de wavelets de raios X em tomografia computadorizada, e temos a representação da Transformada de Radon por wavelets,
2. pulsos de wavelets eletromagnéticas [39, 40] em imagem por ressonância magnética, e temos a Transformada Wavelet Discreta sendo utilizada no lugar da Transformada Discreta de Fourier,
3. por pulsos de wavelets de sons [41] em aparelhos de ultrassom, e temos a representação, da correspondente generalização da Transformada de Radon para ondas, na base das wavelets.

KAISER [21] faz uma apresentação teórica das wavelets eletromagnéticas e sonares, no contexto da representação de grupos de transformações, explorando o fato de que as equações que governam estes fenômenos são invariantes por escalas e translações (grupo afim), que são justamente as transformações usadas na definição das wavelets.

A relação entre Transformada de Radon e wavelets torna-se clara quando se examina a expressão

$$F_{\theta}^{-1} \left( |k| W(k) F_{\theta} \left( \overset{\vee}{g}(\rho, \theta) \right) \right)$$

que aparece na fórmula de inversão da transformada discreta 6.5. Pode ser representada por uma convolução no domínio do espaço, para cada direção fixada pelo ângulo  $\theta$ ,

$$F_{\theta}^{-1} (|k| W(k)) * \overset{\vee}{g}(\rho, \theta)$$

e temos assim a filtragem no domínio do espaço, do senograma  $\check{g}(\rho, \theta)$  pelo filtro  $f(\rho) = F_\theta^{-1}(|k|W(k))$ .

O produto  $|k|W(k)$  no domínio da frequência é um filtro passa-alta  $|k|$  multiplicado por uma função janela arbitrária  $W(\cdot)$  e o que se procura é uma função  $W(\cdot)$  que se anule em  $k = 0$  para que  $|k|W(k)$  varie suavemente em  $k = 0$ , e assim  $F_\theta^{-1}(|k|W(k))$  tenha suporte limitado. Como

$$m_n = \int_{-\infty}^{+\infty} x^n f(x) dx = (j)^n \frac{d^n \hat{f}(0)}{dk^n}$$

é o  $n$ -ésimo momento da função  $f(\cdot)$ , o que se procura é um filtro passa-alta  $f(\rho)$  de momento pelo menos igual a 0 e a wavelet é uma função que possui esta propriedade.

Podemos apresentar três aplicações importantes do uso das wavelets na reconstrução em multi-resolução de uma imagem através de sua projeção:

1. reconstrução progressiva de uma imagem através de acesso remoto a um aparelho de tomografia computadorizada, [42],
2. reconstrução local de uma imagem envolvendo apenas a aquisição da imagem próxima a região de interesse, [43, 44],
3. reconstrução da imagem em multi-resolução tendo como objetivo a eliminação de ruídos, [45, 46].

No que se segue faremos a apresentação das definições, tomando por referência [47, 38], tendo em vista a formulação algébrica em multi-resolução da transformada algébrica de Radon, que denominaremos “Transformada B-Radonlet Algébrica”.

## 6.1 Transformada Discreta de Fourier

**Definição 6.1.1** Se  $x = (x_1 \cdots x_n)^T$  então definimos a transformada direta discreta de Fourier por,

$$X = DFT(x) = (X_1 \cdots X_n)^T$$

onde

$$X_k = \sum_{n=1}^N x_n e^{-i\left(\frac{2\pi}{N}\right)kn}$$

**Definição 6.1.2** Se  $X = (X_1 \cdots X_n)^T$  então definimos a transformada inversa discreta de Fourier por,

$$x = IDFT(x) = (x_1 \cdots x_n)^T$$

onde

$$x_n = \sum_{k=1}^N x_k e^{i\left(\frac{2\pi}{N}\right)kn}$$

## 6.2 Transformada de Radon

### 6.2.1 Transformada contínua

**Definição 6.2.1** Considere  $g : (x, y) \in R^2 \mapsto z \in R$ . Definimos a transformada de Radon contínua por

$$R(g)(\rho, \theta) = \overset{\vee}{g}(\rho, \theta) = \int_{-\infty}^{+\infty} g\left(\rho \vec{\theta} + s(\vec{\theta})^\perp\right) ds$$

onde

$$\vec{\theta} = (\cos(\theta), \sin(\theta)) \quad e \quad (\vec{\theta})^\perp = (-\sin(\theta), \cos(\theta))$$

Considere  $g : (\rho, \theta) \in R^2 \mapsto z \in R$ . Definimos a transformada inversa de Radon contínua por

$$R^{-1}\left(\overset{\vee}{g}(\rho, \theta)\right)(x, y) = BF_\theta^{-1}\left(|k| F_\theta\left(\overset{\vee}{g}(\rho, \theta)\right)\right) \quad (6.1)$$

onde temos a transformada direta de Fourier em relação a  $\rho$ ,

$$F_\theta\left(\overset{\vee}{g}(\rho, \theta)\right) = \int_{-\infty}^{+\infty} \overset{\vee}{g}(\rho, \theta) e^{-i2\pi k\rho d\rho} \quad (6.2)$$

e a transformada inversa de Fourier em relação a  $k$ ,

$$F_\theta^{-1}(G(k, \theta)) = \int_{-\infty}^{+\infty} G(k, \theta) e^{i2\pi k d k} \quad (6.3)$$

e o operador de retro-projeção

$$B(\bar{g}(\rho, \theta)) = \int_0^\pi \bar{g}(x \cos(\theta) + y \sin(\theta), \theta) d\theta \quad (6.4)$$

Na figura 6.1 temos a transformada de Radon contínua.

$|k|$  é um filtro passa-alta no domínio da frequência. Isto sugere o uso de filtros wavelets

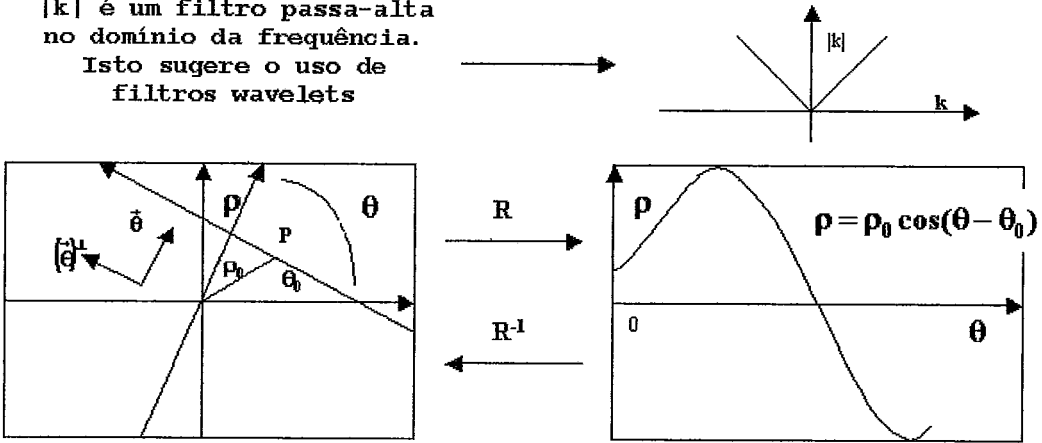


Figura 6.1: Transformada de Radon contínua

### 6.2.2 Transformada discreta

Para sinais discretos ,

$$g : (x_m, y_n) \in R^2 \mapsto z_k \in R$$

onde os parâmetros são dados por,

$$\begin{aligned} x &= x_m = x_{\min} + m\Delta x, m = 0, 1, \dots, M - 1 \\ y &= y_n = y_{\min} + n\Delta y, n = 0, 1, \dots, N - 1 \\ \theta &= \theta_t = \theta_{\min} + t\Delta\theta, t = 0, 1, \dots, T - 1 \\ \rho &= \rho_r = \rho_{\min} + r\Delta\rho, r = 0, 1, \dots, R - 1 \end{aligned}$$

onde em geral considera-se

$$\theta_{\min} = 0 \quad \text{e} \quad \Delta\theta = \frac{\pi}{T}$$

temos as seguintes definições:

**Definição 6.2.2** A transformada discreta de Radon é definida por,

$$R(g)(r, t) = \check{g}(r, t) = \sum_{j=0}^{J-1} g \left( \rho_r \vec{\theta}_t + s_j \left( \vec{\theta}_t \right)^\perp \right) \Delta s$$

onde

$$\vec{\theta}_t = (\cos(\theta_t), \sin(\theta_t)) \quad \text{e} \quad \left( \vec{\theta}_t \right)^\perp = (-\sin(\theta_t), \cos(\theta_t))$$

e os valores de  $g$  são obtidos por interpolação.

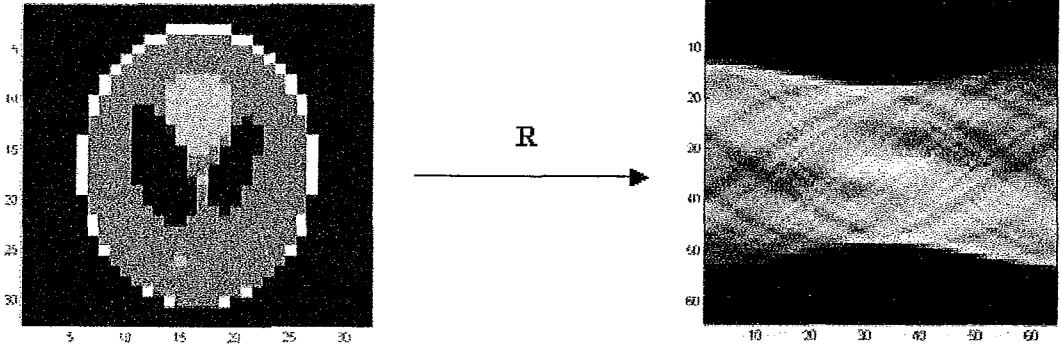


Figura 6.2: Transformada de Radon discreta

Na figura 6.2 temos a transformada discreta de Radon do phantom-head  $32 \times 32$  em um senograma  $64 \times 64$ .

**Definição 6.2.3** A transformada inversa discreta de Radon é definida considerando a discretização da formulação contínua na definição 6.1

$$R^{-1} \left( \check{g}(\rho, \theta) \right) (x, y) = BF_{\theta}^{-1} \left( |k| W(k) F_{\theta} \left( \check{g}(\rho, \theta) \right) \right) \quad (6.5)$$

onde  $W(k)$  é um filtro para atenuar as altas frequências favorecidas pelo filtro passa-alta  $|k|$ , denominado "rampa", e devemos considerar, em 6.2 a transformada discreta de Fourier, em 6.3 a transformada inversa discreta de Fourier e em 6.4 a discretização correspondente

$$g(x_m, y_n) = \Delta\theta \sum_{t=0}^{T-1} \bar{g}(x_m \cos(\theta_t) + y_n \sin(\theta_t), \theta_t)$$

### 6.2.3 Transformada algébrica

Consideremos,  $M$  o número de colunas,  $N$  o número de linhas da imagem,  $R$  o número de raios ao longo da reta  $\rho$ , e  $T$  o número de ângulos entre 0 e  $\pi$  e tomemos

$$\Delta x = \Delta y = \Delta\theta = \Delta\rho = 1$$

e uma base interpolante,

$$\Phi = \left\{ \phi_{m,n} = \phi(x - m, y - n), \right\}$$

onde

$$m = 0, \dots, M - 1, n = 0, \dots, N - 1$$

assim uma função  $g : (x, y) \in R^2 \mapsto z \in R$  pode ser expressa como,

$$g(x, y) = \sum_m \sum_n g(m, n) \phi(x - m, y - n)$$

sendo a transformada de radon linear temos,

$$\check{g} = \sum_m \sum_n \check{\phi}_{m,n} g(m, n)$$

e usando a notação matricial podemos escrever,

$$\begin{aligned} & \left( \check{g}(0, 0) \quad \cdots \quad \check{g}(0, T-1) \quad \cdots \quad \check{g}(R-1, T-1) \right)^T = \\ = & \begin{pmatrix} \check{\phi}_{0,0}(0, 0) & \cdots & \check{\phi}_{M-1, N-1}(0, 0) \\ \vdots & & \vdots \\ \check{\phi}_{0,0}(0, T-1) & \cdots & \check{\phi}_{M-1, N-1}(0, T-1) \\ \vdots & & \vdots \\ \check{\phi}_{0,0}(R-1, 0) & \cdots & \check{\phi}_{M-1, N-1}(R-1, 0) \\ \vdots & & \vdots \\ \check{\phi}_{0,0}(R-1, T-1) & \cdots & \check{\phi}_{M-1, N-1}(R-1, T-1) \end{pmatrix} \begin{pmatrix} g(0, 0) \\ \vdots \\ g(0, N-1) \\ \vdots \\ g(M-1, 0) \\ \vdots \\ g(M-1, N-1) \end{pmatrix} \end{aligned}$$

onde as matrizes  $\check{g}(x, y)$  e  $g(m, n)$  foram escritas, justapondo as linhas, na forma de um vetor com  $R \times T$  e  $M \times N$  elementos, respectivamente. Temos assim o sistema de equações algébricas,

$$b = Ax$$

onde a matriz  $A$  tem as dimensões  $(R \times T) \times (M \times N)$ .

**Definição 6.2.4** *Define-se a Transformada Algébrica de Radon  $b$  de  $x \in R^{M \times N}$  pela equação matricial acima, ou seja,*

$$b = Ax$$

**Definição 6.2.5** *Define-se a Transformada Inversa Algébrica de Radon  $x$  de  $b \in R^{R \times T}$ , a determinação de  $x$  como inversa generalizada, pelo Algoritmo do Gradiente Conjugado dos Mínimos Quadrados (GCMQ), na equação matricial*

$$b = Ax$$

Sinograma ( $b$ ) Em 2 dimensões : $2^6 \times 2^6 = 64 \times 64$ Em 1 dimensão: $2^{12} = 4096$	Shepp-Logan phantom head ( $x$ ) Em 2 dimensões : $2^5 \times 2^5 = 32 \times 32$ Em 1 dimensão: $2^{10} = 1024$
---------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------

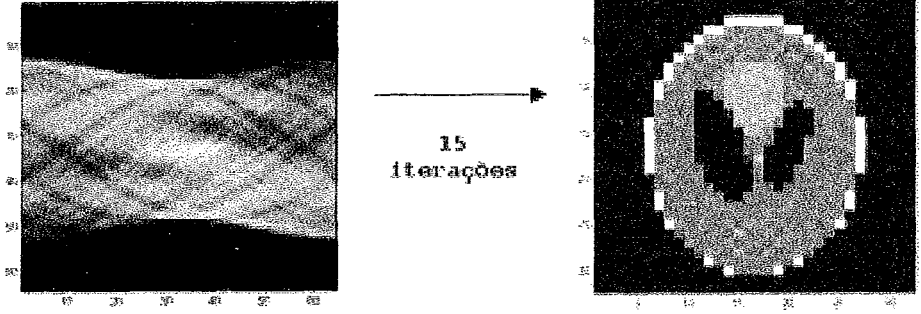


Figura 6.3: Transformada inversa algébrica de Radon usando o algoritmo do gradiente conjugado dos mínimos quadrados

Consideremos, por exemplo,

$$A = \left( \bigvee \phi_{m,n}(r, t) \right)$$

$$\phi_{m,n} = \phi(x - m, y - n)$$

$$\phi(x, y) = Haar(x) \times Haar(y)$$

$A$  é uma matriz  $4096 \times 1024$

$b$  é um senograma  $2^6 \times 2^6 = 64 \times 64$  ou  $2^{12} = 4096$

$x$  é o Shepp-Logan phantom head  $2^5 \times 2^5 = 32 \times 32$  ou  $2^{10} = 1024$

Na figura 6.3 podemos ver a transformada inversa algébrica de Radon, onde se faz o cálculo da solução aproximada de  $x$  a partir do senograma unidimensional  $b$  usando o algoritmo do gradiente conjugado com o erro dos mínimos quadrados. Obtemos assim a solução aproximada em 15 iterações.



## 6.3 Aplicação

### 6.3.1 A Transformada algébrica de Radon com B-Wavelets (B-RADONLET)

O objetivo agora é *propor uma definição envolvendo a unificação da transformada algébrica de Radon com as B-Wavelets, tendo em vista a reconstrução de imagens, em tomografia computadorizada, em vários níveis de resolução.* Nesta definição a imagem e o senograma, representados por vetores colunas, podem estar em níveis diferentes de resolução e a B-Wavelet aplicadas no estudo da imagem pode possuir o grau distinto da B-Wavelet usada na representação do senograma. Isto ficará claro na notação  $(j, i)$ , onde o inteiro  $j$  representa o nível de resolução da imagem e o inteiro  $i$  representa o nível de resolução do senograma.

Para tanto consideremos  $R : \overline{V}^j \mapsto V^i$  a transformada de Radon e  $\overline{\Phi}^j$  base de B-Splines para  $\overline{V}^j$  no nível  $j$  de resolução e  $\Phi^i$  base de B-Splines para  $V^i$  no nível  $i$  de resolução. Temos assim a representação matricial da transformada de Radon no nível  $(j, i)$  de resolução,

$$A^{j,i} = [R]_{\overline{\Phi}^j, \Phi^i}$$

Daí, propomos as seguintes definições:

**Definição 6.3.1** *Denominamos Transformada Direta B-Radonlet, no nível  $(j, i)$  de resolução, a transformação denotada por  $FR\_letT^{j,i}(\cdot)$  e definida por*

$$FR\_letT^{j,i}(\overline{C}^j) = A^{j,i}\overline{C}^j$$

onde

$$\overline{C}^j = [\overline{X}]_{\overline{\Phi}^j}$$

**Definição 6.3.2** *Denominamos Transformada Inversa Generalizada B-Radonlet, no nível  $(j, i)$  de resolução, a transformação denotada por  $GIR\_letT^{j,i}(\cdot)$  e definida por*

$$\overline{C}^j = GIR\_letT^{j,i}(b^i)$$

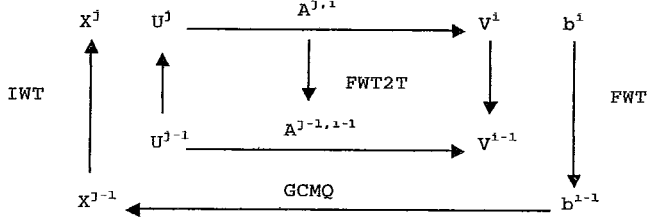
onde  $\overline{C}^j$  é solução obtida, pelo algoritmo do gradiente conjugado com o erro mínimo quadrado (GCMQ), na equação,

$$A^{j,i}\overline{C}^j = b^i$$

Na figura 6.4 podemos ver uma aplicação da Transformada Inversa Generalizada B-Radonlet, no nível (9,11) de resolução, onde a inversa generalizada que fornece a imagem, é obtida em 15 iterações usando o Algoritmo do Gradiente Conjugado dos Mínimos Quadrados (GCMQ).

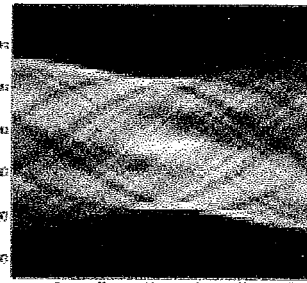
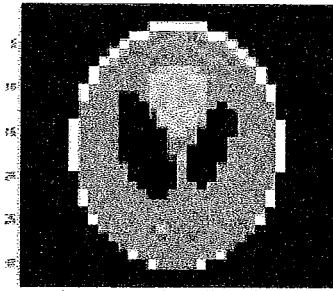
Transformada Algébrica de Radon com wavelet  
RADONLET Algébrica

$X^j$  vetor no nível  $j$  de resolução,  
 $b^i$  vetor no nível  $i$  de resolução,  
 $A^{j,i}$  matriz com (largura, altura) no nível  $(j,i)$  de resolução.



Phantom Head no nível 10

Sinograma no nível 12



↑ IWT (HAAR)

↓ FWT (HAAR)

Phantom head no nível 9

Sinograma no nível 11

GCMQ (15 iterações)

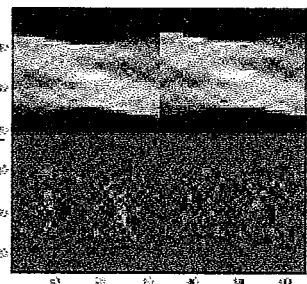
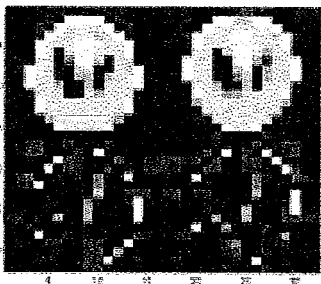


Figura 6.4: Transformada inversa generalizada algébrica de Radon, com wavelets, usando o algoritmo GCMQ.

# Capítulo 7

## Conclusão

O desenvolvimento ao longo da tese demonstrou que é possível uma apresentação matricial unificada do conceito de multi-resolução, com B-Splines e as B-Wavelets associadas, no contexto de modelagem geométrica, filtragem e reconstrução de imagens. Para tanto foi introduzido o conceito de nível de resolução  $(j, i)$  para uma imagem ou matriz junto com a possibilidade de se poder usar filtros passa-baixa com B-Splines distintas ao longo das linhas e colunas e filtros passa-alta com B-Wavelets distintas ao longo das linhas e colunas.

Notemos que os resultados apresentados não dependem das B-Splines, logo é possível procurar adaptar os filtros escolhendo as B-Splines que possam resolver, por exemplo, os seguintes problemas:

I- Para B-wavelets semi-ortogonais os filtros de análise, ver exemplo 3.4.1,  $\widetilde{H}^j$  e  $\widetilde{G}^j$  em geral são densos e o ideal é que fossem esparsos e para tanto faz-se necessário construir as correspondentes matrizes  $H^j$ ,  $G^j$ ,  $\widetilde{H}^j$  e  $\widetilde{G}^j$  para B-Wavelets bi-ortogonais.

II- Outro aspecto importante é determinar as matrizes de refinamento de tal forma que todas tenham as dimensões das linhas e colunas como sendo potências de 2, desta forma podemos generalizar as B-Wavelets para “B-Wavelets Packets”. Neste caso a técnica de multi-resolução poderá ser aplicada não sómente na banda LL, como é o caso das Transformadas Wavelets, mas também nas outras bandas HL, LH e HH

Com relação as B-Randolets, o que se pretende dentre os vários estudos possíveis, é a reconstrução local de imagens com atenuação dos ruídos.

# Apêndice A

## Condicionamento de matrizes e o algoritmo do gradiente conjugado

Neste apêndice faremos uma revisão de dois tópicos importante em análise numérica, condicionamento de matrizes e o algoritmo do gradiente conjugado para a solução de sistemas de equações lineares.

Se temos um sistema de equações lineares

$$AX = b$$

o conceito de condicionamento nos mostra o quanto a solução  $X$  é estável em relação a pequenas perturbações  $\delta A$  nos elementos da matriz  $A$  dos coeficientes e pequenas perturbações  $\delta b$  no termo não homogêneo  $b$ , provocados por pequenos erros de avaliação. Nas aplicações o que interessa é que pequenos erros em  $A$  e  $b$  acarretem um pequeno erro no cálculo de  $X$  e o número de condicionamento  $\kappa(A)$  associado a matriz  $A$  nos fornece uma medida para esta estabilidade.

Por outro lado, o algoritmo do gradiente conjugado é um método iterativo para a determinação de uma solução do sistema de equações lineares que se tem demonstrado atrativo tanto em modelagem geométrica [33] como em reconstrução de imagens [48], por algumas razões:

I- é simples,

II- são executadas poucas operações minimizando assim os erros provenientes dos cálculos,

III- em cada iteração, o algoritmo faz uma avaliação da melhor direção no espaço das soluções que leva a solução final, se adaptando automaticamente aos os erros numéricos.

Apresentaremos também uma versão otimizada [49] do algoritmo do gradiente conjugado, para a determinação da solução aproximada do sistema de equações lineares, segundo o erro mínimo quadrado, que é útil no contexto da inversa generalizada presente na inversão da formulação algébrica da Transformada de Radon.

As revisões a seguir estão baseadas nas referências [50], [51] e [49].

## A.1 Normas de vetores e matrizes

**Definição A.1.1** *Considere  $X \in R^n$  e  $Y \in R^n$  vetores. Denominamos norma de  $X$ , a função  $\|\cdot\| : R^n \rightarrow R$  que satisfaz as propriedades:*

$$\|X\| \geq 0 \quad e \quad \|X\| = 0 \iff X = 0$$

$$\|cX\| = c\|X\| \quad \text{para qualquer real } c$$

$$\|X + Y\| \leq \|X\| + \|Y\|$$

São de interesses em análise numérica as seguintes normas para  $X = (x_1 \cdots x_n)^T$ ,

**Exemplo A.1.1** *Norma da soma dos módulos,*

$$\|X\|_1 = |x_1| + \dots + |x_n|$$

**Exemplo A.1.2** *Norma dos mínimos quadrados,*

$$\|X\|_2 = (|x_1|^2 + \dots + |x_n|^2)^{1/2} = \sqrt{X^T X}$$

**Exemplo A.1.3** *Norma do máximo, do sup ou do infinito,*

$$\|X\|_\infty = \max_i |x_i|$$

**Definição A.1.2** *Considere  $X \in R^n$  vetor e  $A \in M_{m \times n}$  matriz. Denominamos norma de  $X$ , a função  $\|\cdot\| : M_{m \times n} \rightarrow R$  dada por*

$$\|A\| = \max_{\|X\|=1} \|AX\|$$

### A.1.1 Propriedades

Consideremos as matrizes,  $A, B \in M_{m \times n}$ , o vetor  $X \in R^n$  e o real  $c$ ,

1.  $\|A\| \geq 0$  e  $\|A\| = 0 \iff A = 0$
2.  $\|cA\| = c\|A\|$  para qualquer real  $c$
3.  $\|A + B\| \leq \|A\| + \|B\|$
4.  $\|AX\| \leq \|A\| \|X\|$
5.  $\|AB\| \leq \|A\| \|B\|$
6.  $\|I\| = 1$

Seja uma matriz  $A \in M_{m \times n}$ , temos então as seguintes normas, derivadas de suas correspondentes para vetores:

### A.1.2 Exemplos

**Exemplo A.1.4** Norma do máximo das somas dos valores absolutos das colunas,

$$\|A\|_1 = \max_j \sum_{i=1}^m |a_{ij}|$$

**Exemplo A.1.5** Norma dos mínimos quadrados,

$$\|A\|_2 = [\text{autovalor máximo de } A^T A]^{1/2}$$

**Exemplo A.1.6** Norma do máximo das somas dos valores absolutos das linhas,

$$\|A\|_\infty = \max_i \sum_{j=1}^m |a_{ij}|$$

## A.2 Condicionamento de matrizes

**Definição A.2.1** O número real  $\kappa(A) = \|A\| \|A^{-1}\|$  é dito Número de Condicionamento de  $A$ .

O Número de Condicionamento de uma matriz  $A$  nos fornece um meio de avaliar o erro na solução  $X_e$ , de um sistema de equações  $(A + \delta A)X = \delta b$ , em função do erro nos coeficientes de  $A$  e no vetor  $b$ .

**Proposição A.2.1** Considere  $P \in M_{n \times n}$  e  $\|\cdot\|$  uma norma. Se  $\|P\| < 1$  então

$I + P$  é inversível, e

$$\frac{1}{1 + \|P\|} \leq \|(I + P)^{-1}\| \leq \frac{1}{1 - \|P\|}$$

*Demonstração.*  $I + P$  é inversível, se e somente se,  $(I + P)X = 0 \iff X = 0$ , o que é verdade uma vez que  $X = -PX \implies 0 \leq \|X\| = \|PX\| \leq \|P\| \|X\| \leq \|X\| \implies X = 0$ , daí existe  $B = (I + P)^{-1}$ . Como  $I = B(I + P)$ , então

$$1 = \|I\| = \|B(I + P)\| \leq \|B\| \cdot \|(I + P)\| \leq \|B\| \cdot (\|I\| + \|P\|) = \|B\| \cdot (1 + \|P\|)$$

Logo,

$$\frac{1}{1 + \|P\|} \leq \|B\|$$

Por outro lado  $B = I - BP$ , donde

$$\|B\| = \|I - BP\| \leq \|I\| + \|BP\| \leq 1 + \|B\| \cdot \|P\|$$

daí,

$$\|B\| \leq \frac{1}{1 - \|P\|}$$

**Proposição A.2.2** Considere  $A \in M_{n \times n}$  inversível e  $R \in M_{n \times n}$ . Considere  $\alpha = \|A^{-1}R\| < 1$ , então

$A + R$  é inversível, e

$$\|(A + R)^{-1}\| \leq \frac{\|A^{-1}\|}{1 - \alpha}$$

*Demonstração:* Como  $A$  é inversível,

$$A + R = A + A(A^{-1}R) = A(I + A^{-1}R)$$

então  $P + A^{-1}R$  é uma perturbação da identidade. Como  $\alpha = \|A^{-1}R\| < 1$  e pela proposição A.2.1  $A + R$  é inversível e

$$(A + R)^{-1} = (I + A^{-1}R)^{-1}A^{-1}$$

donde,

$$\|(A + R)^{-1}\| \leq \|(I + A^{-1}R)^{-1}\| \|A^{-1}\| = \frac{\|A^{-1}\|}{1 - \alpha}$$

**Proposição A.2.3** *Seja  $A \in M_{n \times n}$  e considere o sistema inversível de equações lineares dado por,  $AX = b$ . Consideremos a solução do sistema numericamente perturbado dado por,*

$$(A + \delta A)(X + \delta X) = b + \delta b \text{ onde } \|(\delta A)A^{-1}\| < 1 \quad (\text{A.1})$$

Então o erro relativo em  $X$  satisfaz a desigualdade:

$$\frac{\|\delta X\|}{\|X\|} \leq M \cdot \kappa(A) \left( \frac{\|\delta b\|}{\|b\|} + \frac{\|\delta A\|}{\|A\|} \right) \quad (\text{A.2})$$

onde

$$M = \left(1 - \|(\delta A)A^{-1}\|\right)^{-1}$$

e

$\kappa(A)$  é o número de condicionamento da matriz  $A$

*Demonstração: A perturbação  $\delta X$  de  $X$  satisfaz a equação matricial dada por A.1, logo*

$$(A + \delta A)\delta X = (b + \delta b) - (A + \delta A)X = (b - AX) + (\delta b - (\delta A)X) = \delta b - (\delta A)X$$

Como  $\|(\delta A)A^{-1}\| < 1$ , pela proposição A.2.2 existe  $(A + \delta A)^{-1}$ , donde

$$\delta X = (A + \delta A)^{-1}(\delta b - (\delta A)X)$$

Fazendo na proposição A.2.2  $R = \delta A$  como sendo a perturbação de  $A$  e  $M = (1 - \|(\delta A)A^{-1}\|)^{-1} = (1 - \alpha)^{-1}$  temos,

$$\delta X = \|(A + \delta A)^{-1}\| \|\delta b - (\delta A)X\| \leq M \cdot \|A^{-1}\| (\|\delta b\| + \|(\delta A)\| \|X\|)$$

Logo,

$$\frac{\|\delta X\|}{\|X\|} \leq M \cdot \|A^{-1}\| \left( \frac{\|\delta b\|}{\|X\|} + \|(\delta A)\| \right)$$

Como  $b = AX$  temos que  $\|b\| \leq \|A\| \|X\|$  donde

$$\|X\| \geq \frac{\|b\|}{\|A\|}$$

Logo,

$$\frac{\|\delta b\|}{\|X\|} + \|(\delta A)\| \leq \frac{\|\delta b\| \|A\|}{\|b\|} + \|(\delta A)\| = \|A\| \left( \frac{\|\delta b\|}{\|b\|} + \frac{\|\delta A\|}{\|A\|} \right)$$

e assim fica demonstrada a desigualdade A.2.



## A.3 Algoritmo do Gradiente Conjugado

### A.3.1 Definições

Se  $A$  é uma matriz  $n \times n$  e  $X$  é um vetor no  $R^n$ , então

$$AX = b \iff AX - b = 0 \iff \nabla \left( \frac{1}{2} X^T A X - X^T b \right) = 0$$

Assim para determinar uma solução  $X^*$  do sistema de equações lineares,

$$AX^* = b$$

é equivalente a encontrar um extremo  $X^*$  da função quadrática,

$$f(X) = \frac{1}{2} X^T A X - X^T b$$

Para descrever o algoritmo do gradiente conjugado (veja [51]) necessitamos de algumas definições,

**Definição A.3.1**  $A \in M_{n \times n}$  é dita simétrica se  $A^T = A$ .

**Definição A.3.2**  $A \in M_{n \times n}$  é dita positiva definida se  $X^T A X > 0$  para  $X \neq 0$ .

**Definição A.3.3** Consideremos a forma bilinear  $B : R^n \times R^n \rightarrow R$  definida por  $B(X, Y) = X^T A Y$ . Se  $A$  é simétrica e positiva definida então  $B$  é um produto interno no  $R^n$ . Dizemos que  $X \in R^n$  e  $Y \in R^n$  são  $A$ -ortogonais se  $B(X, Y) = X^T A Y = 0$ .

**Definição A.3.4**  $X \in R^n$  e  $Y \in R^n$ , são ditas direções conjugadas, em relação a matriz  $A$ , se  $B(X, Y) = 0$ .

**Definição A.3.5** Denominamos norma da energia de  $X \in R^n$ , o real dado por

$$\|X\|_E = \sqrt{B(X, X)} = \sqrt{X^T A X}$$

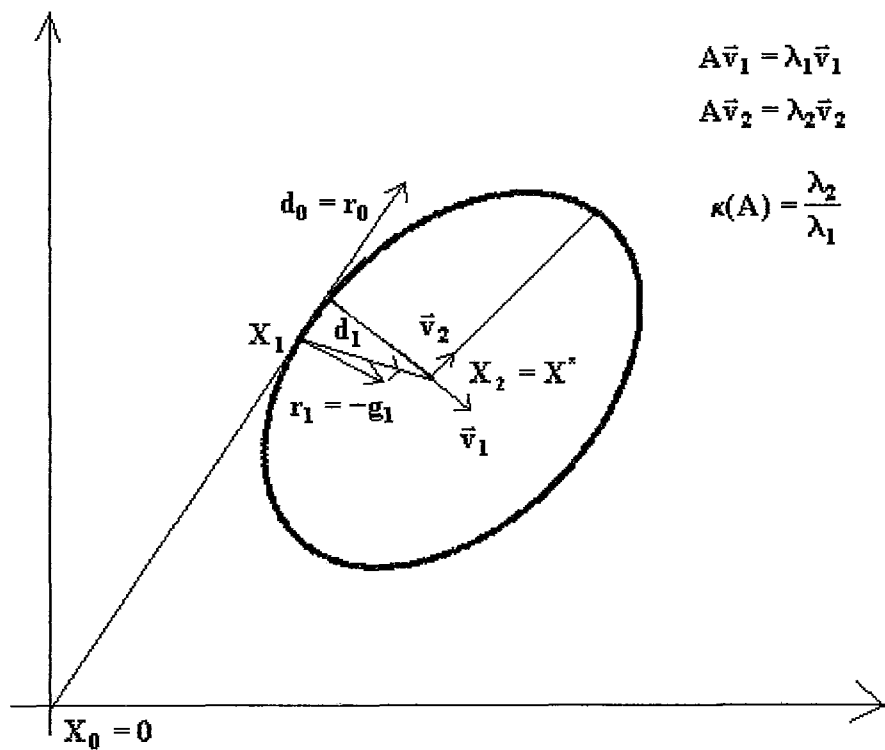


Figura A.1: O algoritmo do gradiente conjugado

### A.3.2 O algoritmo do gradiente conjugado

**Proposição A.3.1 (O algoritmo do gradiente conjugado)** Considere  $g(X) = AX - b = \nabla f(X)$ .

Inicialize o ponto  $X_0$  e o erro  $d_0 = -g_0 = b - AX_0$

Para  $k = 0, \dots, n$  faça,

determine  $X_{k+1} = X_k + \alpha_k d_k$  onde  $\alpha_k = -\frac{g_k^T d_k}{d_k^T A d_k}$ .

determine  $d_{k+1} = -g_{k+1} + \beta_k d_k$  onde  $\beta_k = \frac{g_{k+1}^T A d_k}{d_k^T A d_k}$

Fim

*Demonstração:* Em cada iteração  $k$  o novo ponto  $X_{k+1} = X_k + \alpha_k d_k$  é determinado a partir do ponto anterior  $X_k$ , em função do desvio  $d_k$  que por sua vez é calculado considerando o gradiente  $g_k$ .

Para calcular o ponto  $X_{k+1}$ ,  $\alpha_k$  é determinado de tal maneira que  $f(X_k + \alpha_k d)$  tenha um mínimo local ao longo da reta  $\{X_k + \alpha d, \alpha \in \mathbb{R}\}$  no ponto  $X_{k+1}$  (veja as figuras A.1e A.2). Daí,

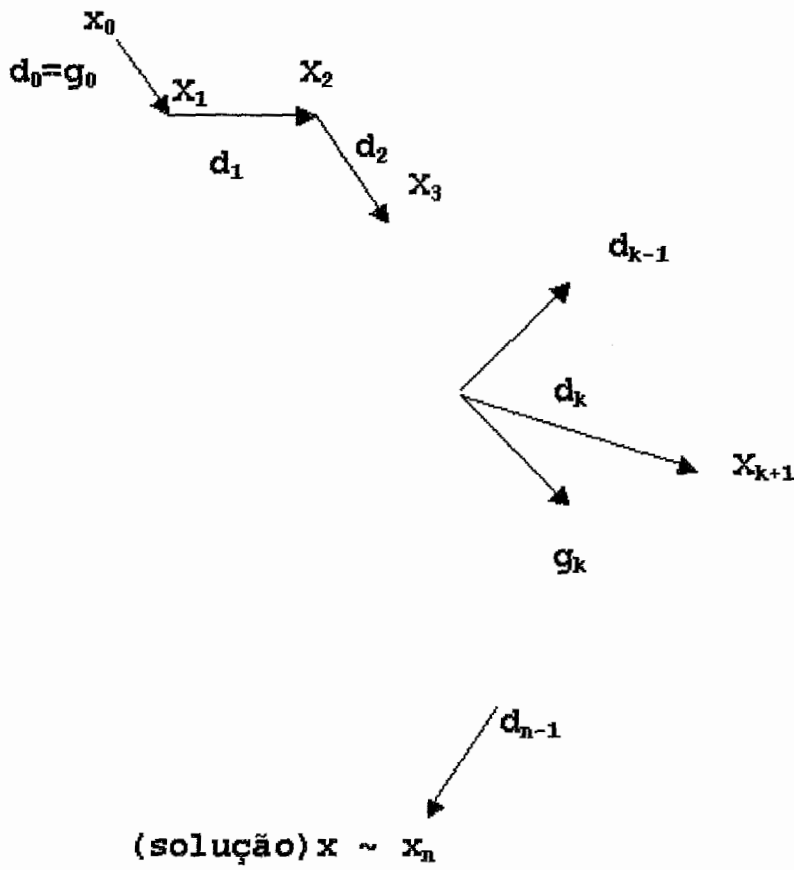


Figura A.2: O algoritmo do gradiente conjugado em  $n$  iterações

$$\begin{aligned}
\frac{d}{d\alpha_k} (f(X_k + \alpha_k d_k)) &= 0 \\
\nabla f(X_k + \alpha_k d)^T \cdot \frac{d}{d\alpha_k} (X_k + \alpha_k d_k) &= 0 \\
(A(X_k + \alpha_k d_k) - b)^T d_k &= 0 \\
((AX_k - b) + \alpha_k Ad_k)^T d_k &= 0 \\
g_k^T d_k + \alpha_k d_k^T A^T d_k &= 0
\end{aligned}$$

Então como a matriz  $A$  é simétrica, podemos concluir que

$$\alpha_k = -\frac{g_k^T d_k}{d_k^T A d_k} \quad (\text{A.3})$$

Para determinar o novo desvio  $d_{k+1}$  em função do novo gradiente  $g_{k+1}$  e do velho desvio  $d_k$ , devemos determinar  $\beta_k$  de tal maneira que

$$d_{k+1} = -g_{k+1} + \beta_k d_k$$

seja uma direção conjugada da anterior  $d_k$ , ou seja,

$$d_{k+1}^T A d_k = 0$$

logo,

$$\begin{aligned}
d_{k+1}^T A d_k &= 0 \\
(-g_{k+1} + \beta_k d_k)^T A d_k &= 0 \\
-g_{k+1}^T A d_k + \beta_k d_k^T A d_k &= 0
\end{aligned}$$

donde,

$$\beta_k = \frac{g_{k+1}^T A d_k}{d_k^T A d_k} \quad (\text{A.4})$$

**Proposição A.3.2** Considere  $g_k(X) = AX - b$  então  $g_{k+1} = g_k + \alpha_k Ad_k$   
*Demonstração:* como

$$X_{k+1} = X_k + \alpha_k d_k$$

então,

$$g_{k+1} = AX_{k+1} - b = A(X_k + \alpha_k d_k) - b = (AX_k - b) + \alpha_k Ad_k = g_k + \alpha_k Ad_k$$

**Proposição A.3.3** Considere  $g_k(X) = AX - b$  então,

$$g_k \perp d_i, i < k$$

e

$$g_{k+1} \perp g_k$$

*Demonstração:* usando a proposição A.3.2 e A.3 temos,

$$d_k^T g_{k+1} = d_k^T (g_k + \alpha_k Ad_k) = d_k^T g_k + \alpha_k d_k^T Ad_k = d_k^T g_k + \left( -\frac{g_k^T d_k}{d_k^T Ad_k} \right) d_k^T Ad_k = 0$$

Por indução sobre  $i$  teremos

$$d_i^T g_k, i < k$$

e, considerando a  $A$ -ortogonalidade das direções conjugadas (veja ...),  $d_i^T Ad_k = 0$ , temos

$$d_i^T g_{k+1} = d_i^T (g_k + \alpha_k Ad_k) = d_i^T g_k + \alpha_k d_i^T Ad_k = 0$$

Por outro lado por A.4 e a primeira parte da demonstração,

$$g_{k+1}^T g_k = g_{k+1}^T (-d_k + \beta_{k-1} d_{k-1}) = -g_{k+1}^T d_k + \beta_{k-1} g_{k+1}^T d_{k-1} = 0$$

### A.3.3 O algoritmo do gradiente conjugado e o método dos mínimos quadrados

Em situações em que o sistema  $AX = b$  é tal que  $A$  é uma matriz retangular e portanto não inversível, podemos assim mesmo estimar uma solução aproximada usando o erro dos mínimos quadrados, e determinando

$$X \in R^n \text{ tal que } X = \min \{ \|AX - b\|_2^2 \}$$

#### Proposição A.3.4

$$X \in R^n \text{ tal que } X = \min \{ \|AX - b\|_2^2 \}$$

então

$$X \in R^n \text{ é solução da equação normal } (A^T A) X = A^T b$$

*Demonstração:* o ponto  $X$  de mínimo é dado calculando aquele que anula as derivadas parciais de  $e(X) = \|AX - b\|_2^2$ . Consideremos,  $f(X) = \|X\|_2^2 =$

$x_1^2 + \dots + x_n^2$  e  $g(X) = AX - b$  donde,  $\nabla f(X) = 2X$  e  $\frac{\partial g}{\partial x_i} = A(0 \dots 1 \dots 0)^T = i$ -ésima columna de  $A = A^i$ . Daí aplicando a regra da cadeia,

$$\frac{\partial e}{\partial x_i}(X) = \frac{\partial}{\partial x_i}(f \circ g)(X) = \nabla f(AX - b) \frac{\partial g}{\partial x_i}(X) = 2(AX - b)A^i = 0$$

Ou ainda,

$$(A^T A)X - A^T b = 0$$

e temos a equação normal.

Podemos otimizar o algoritmo do gradiente conjugado para o caso de uma equação normal  $(A^T A)X = A^T b$  e para tanto precisaremos dos seguintes resultados.

**Proposição A.3.5** Considere  $g_k(X) = A^T A X - A^T b$  então  $g_{k+1} = g_k + \alpha_k A^T A d_k$

*Demonstração:* como

$$X_{k+1} = X_k + \alpha_k d_k$$

então,

$$g_{k+1} = A^T A X_{k+1} - A^T b = A^T A (X_k + \alpha_k d_k) - A^T b = g_k + \alpha_k A^T A d_k$$

**Proposição A.3.6** No caso da equação normal  $(A^T A)X = A^T b$ ,  $\alpha_k$  em A.3 pode ser escrito como

$$\alpha_k = \frac{\|g_k\|_2^2}{\|q_k\|_2^2}$$

onde

$$g_k(X) = A^T A X - A^T b \text{ e } q_k = A d_k$$

*Demonstração:* Para a equação normal, com  $g_k$  acima, temos,

$$\alpha_k = -\frac{g_k^T d_k}{d_k^T A^T A d_k}$$

Por outro lado, usando a proposição A.3.3 e  $d_k = -g_k + \beta d_{k-1}$  temos o produto interno com  $g_k$ ,

$$g_k^T d_k = -g_k^T g_k + \beta g_k^T d_{k-1} = -g_k^T g_k + 0 = -g_k^T g_k$$

Logo, como  $\|X\|_2^2 = X^T X$ ,

$$\alpha_k = -\frac{g_k^T d_k}{d_k^T A^T A d_k} = \frac{g_k^T g_k}{(A d_k)^T A d_k} = \frac{\|g_k\|_2^2}{\|A d_k\|_2^2} = \frac{\|g_k\|_2^2}{\|q_k\|_2^2}$$

**Proposição A.3.7** No caso da equação normal  $(A^T A) X = A^T b$ ,  $\beta_k$  em A.4 pode ser escrito como

$$\beta_k = \frac{\|g_{k+1}\|_2^2}{\|g_k\|_2^2}$$

onde

$$g_k(X) = A^T A X - A^T b$$

*Demonstração:* Usando, A.3.3, A.3.5, A.3.6, A.4,  $q_k = Ad_k$  e  $\|X\|_2^2 = X^T X$  temos,

$$\begin{aligned} \beta_k &= \frac{\|g_{k+1}\|_2^2}{\|g_k\|_2^2} = \frac{g_{k+1}^T g_{k+1}}{\|g_k\|_2^2} = \frac{g_{k+1}^T (g_k + \alpha_k A^T q_k)}{\|g_k\|_2^2} = \\ &= \frac{g_{k+1}^T g_k}{\|g_k\|_2^2} + g_{k+1}^T \left( \frac{\|g_k\|_2^2}{\|q_k\|_2^2} \right) \frac{A^T q_k}{\|g_k\|_2^2} = 0 + \frac{g_{k+1}^T A^T Ad_k}{(Ad_k)^T Ad_k} = \frac{g_{k+1}^T A^T Ad_k}{(Ad_k)^T Ad_k} \end{aligned}$$

**Proposição A.3.8 (Algoritmo do gradiente conjugado dos mínimos quadrados para equações normais)** Considere  $g(X) = A^T (AX - b)$

Inicialize  $X_0$ ,  $g_0 = d_0 = A^T (AX_0 - b)$  e  $q_0 = Ad_0$

Para  $k = 0, \dots, n$  faça,

$$\alpha_k = \frac{\|g_k\|_2^2}{\|q_k\|_2^2}$$

$$X_{k+1} = X_k + \alpha_k d_k$$

$$g_{k+1} = g_k + \alpha_k A^T q_k$$

$$\beta_k = \frac{\|g_{k+1}\|_2^2}{\|g_k\|_2^2}$$

$$d_{k+1} = -g_{k+1} + \beta_k d_k$$

$$q_k = Ad_k$$

Fim

*Demonstração:* consequência imediata das proposições A.3.1, A.3.6 e A.3.7.

**Proposição A.3.9** Se  $\kappa(A)$  é o número de condicionamento da matriz  $A$  simétrica e positiva definida, então a distância entre  $X_{k+1}$  e a solução  $X^*$  no algoritmo do gradiente conjugado é dada por,

$$\|X_{k+1} - X^*\|_E \leq \left( \frac{\kappa(A) - 1}{\kappa(A) + 1} \right)^k \|X_0 - X^*\|_E$$

onde

$$\|X\|_E = \sqrt{(X^T A X)}$$

Demonstração: basta usar para  $0, \dots, k$  a desigualdade

$$\|X_{k+1} - X^*\|_E \leq \left( \frac{\kappa(A) - 1}{\kappa(A) + 1} \right) \|X_k - X^*\|_E$$

encontrada em [51]

A desigualdade da proposição anterior nos faz ver, que a taxa de convergência do algoritmo do gradiente conjugado aumenta na medida que o número de condicionamento  $\kappa(A)$  da matriz  $A$  se aproxima de 1. Assim se  $\kappa(A) = 1$  o algoritmo converge em uma única iteração.

No caso do sistema  $AX = 0$  onde  $A = I$ , temos  $\kappa(A) = \|A\| \|A\|^{-1} = 1$  e a forma quadrática associada é,

$$f(X) = \frac{1}{2} X^T I X - X^T 0 = x^2 + y^2$$

onde as curvas de níveis são círculos. Assim se  $X_0 \in R^2$ , a direção contrária do gradiente,  $-\nabla f(X_0) = -2X_0$ , em uma única iteração nos leva a solução exata de  $IX = 0$  dada por  $X^* = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ .

Geométricamente o comprimento dos eixos principais dos círculos são os auto-valores de  $A$ , donde  $\lambda_{\min} = \lambda_{\max} = 1$ , logo para  $A$  simétrica e positiva definida,  $\kappa(A) = \frac{\lambda_{\max}}{\lambda_{\min}} = 1$ . Assim o número de condicionamento de  $A$  é uma medida de quanto uma elipse está mais próxima de um círculo e portanto melhora a eficiência do algoritmo do gradiente conjugado. .

A mudança de “escala” ao longo dos eixos principais que transformaria elipse em círculo nos sugere o uso das wavelets com o método do gradiente conjugado.



# Bibliografia

- [1] de Casteljaou, P., *Courbes et surfaces á pôles*. Citroen, 1959.
- [2] Bézier, P. E., *Emploi des machines á commande nuérique*. Paris: Masson net Cie, 1970.
- [3] Bézier, P. E., Mathematical and pratical possibilities of unisurf. In: *Computer Aided Geometric Design* ( Barnhill, R. E., Riesenfeld, R. F., eds.), pp. 127–152, Academic Press, New York, 1974.
- [4] Bézier, P. E., *Essai de Définition Numériques des Courbes et des Surfaces Expérimentales*. Tese de Doutoramento, Université Pierre et Marie Curie, Paris, February 1977.
- [5] Bernstein, S., Démonstration du théoreme de weierstrass fondeé sur le calcul des probabilités. In: *Matem ob-va*, v. 13, pp. 1–2, Harkov Soobs, 1912.
- [6] Leighton, W., *Equações Diferenciais Ordinárias*. 2<sup>a</sup> ed., Rio de Janeiro,RJ: Livros Técnicos e Científicos Editora, 1978.
- [7] Becker, E. B., Carey, G. F., Oden, J. T., *Finete Element - An Introduction*, v. 1. 1<sup>a</sup> ed., Prentice-Hall, 1981.
- [8] Riesenfeld, R. F., *Applications of B-Spline Aproximation to Geometric Problems of Computer-Aided Design*. Tese de Doutoramento, Syracuse University, Syracuse, NY, May 1973.
- [9] de Boor, C. On calculating with b-splines. *Journal of Approximation Theory*, v. 6, n. 1, pp. 50–62, July 1972.
- [10] de Boor, C., B(asic)-splines. preprint.
- [11] Fowler, B., Geometric manipulation of tensor product surfaces. In: *Symposium on Interactive 3D Graphics*, pp. 101–108, New York: ACM, 1992.

- [12] Daubechies, I., *Ten Lectures on Wavelets*, v. 61 de *Regional Conference Series in Applied Mathematics*. 1<sup>a</sup> ed., SIAM, 1992.
- [13] Chui, C. K., *An Introduction to Wavelets*. 1<sup>a</sup> ed., Academic Press, 1992.
- [14] Aldroubi, M. U. A., Eden, M. B-spline signal processing: Part i-theory. *IEEE Trans. Transactions on Signal Processing*, v. 41, n. 2, pp. 821–833, 1993.
- [15] Aldroubi, M. U. . A., Eden, M. B-spline signal processing: Part ii-efficient design and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 41, n. 2, pp. 834–848, 1991.
- [16] Unser, M., Ten good reasons for using spline wavelets. In: *Wavelets Applications in Signal and Image Processing V*, v. 3169, pp. 422–431, SPIE, 1997.
- [17] Radon, J. Uber die bestimmung von funktionen durch ihre integralwerte langs gewisser mannigfaltigkeiten. *Berichte Sachsische Akademie der Wissenschaften-Mathematische-Physikalische Klasse*, v. 69, pp. 262–267, 1917.
- [18] Radon, J. On the determination of functions from their integral values along certain manifolds. *IEEE Transactions on Medical Imaging*, v. MI-5, pp. 170–176, 1986.
- [19] Kormack, A. M. Reconstruction of densities from their projections with applications in radiological physics. *Phys. Med. biol.*, v. 18, n. 2, pp. 195–207, 1973.
- [20] Hounsfield, G. N. Computerized transverse axial scanning (tomography): Part i. description of system. *British Journal of Radiology*, v. 46, pp. 1016–1022, 1993.
- [21] Kaiser, G., *A Friendly Guide to Wavelets*. 1<sup>a</sup> ed., Birkhauser, 1994.
- [22] Bartels, R. H., Beaty, J. C., Barsky, B. A., *An Introduction to Splines for use in Computer Graphics and Geometric Modeling*. 1<sup>a</sup> ed., Morgan Kaufmann Publishers, INC, 1987.
- [23] Stollnitz, E. J., Rose, T. D., Salesin, D. H., *Wavelets for Computer Graphics*. 1<sup>a</sup> ed., Morgan Kaufmann Publishers, Inc, 1996.
- [24] Persiano, R. M., *Bases da Modelagem Geométrica*. 10 Escola de Computação, Unicamp, 1996.

- [25] Farin, G., *Curves and Surfaces for CAGD - A Practical Guide*. 4<sup>th</sup> ed., Academic Press, 1997.
- [26] Boehm, W. Inserting new knots into b-splines curves. *Computer Aided Des.*, v. 12, n. 4, pp. 199–201, July 1980.
- [27] Chui, C. K., *Wavelets: A Mathematical Tool for Signal Processing*. 1<sup>st</sup> ed., SIAM, Society for Industrial and Applied Mathematics, 1997.
- [28] Stollnitz, E. J., Rose, T. D., Salesin, D. H. Wavelets for computer graphics: A primer, part 1. *IEEE Computer Graphics and Applications*, v. 15, n. 3, pp. 76–84, May 1995.
- [29] Stollnitz, E. J., Rose, T. D., Salesin, D. H. Wavelets for computer graphics: A primer, part 2. *IEEE Computer Graphics and Applications*, v. 15, n. 4, pp. 75–85, July 1995.
- [30] Brito, J. A. P., Métodos computacionais meshless: classificação, teoria básica e aplicações. Exame de Qualificação-COPPE-LCG.
- [31] Liu, W. K., Jun, S., Sihling, D. T., Chen, Y., Hao, W. Multiresolution reproducing kernel particle method for computational fluid dynamics. *International Journal for Numerical Methods in Fluids*, v. 24, pp. 1391–1415, 1997.
- [32] Gortler, S. J., Cohen, M. F., Hierarchical and variational geometric modeling with wavelets. Rel. Téc. 25, Microsoft Research, July 1995.
- [33] Szeliski, R. Fast surface interpolation using hierarchical basis functions. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, v. 12, n. 6, pp. 513–528, June 1990.
- [34] Pentland, A. P. Fast solutions to physical equilibrium and interpolation problems. *The Visual Computer*, v. 8, pp. 303–314, 1992.
- [35] Yaou, M.-H., Chang, W.-T. Fast surface interpolation using multiresolution wavelet transform. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, v. 16, n. 7, pp. 673–688, July 1994.
- [36] Bueno, L. P., *Minimização da Energia de Deformação na Modelagem de Terreno*. Tese de Doutorado, Engenharia de Sistemas e Computação, COPPE,UFRJ, Março 1996.
- [37] Lipschutz, S., *Álgebra Linear. Coleção Schaum*. McGraw-Hill do Brasil, 1981.

- [38] Jain, A. K., *Fundamentals of Digital Image Processing*. 1<sup>st</sup> ed., Prentice Hall, 1989.
- [39] Jr, D. M. H., Weaver, J. B. Two applications of wavelet transforms in magnetic resonance imaging. *IEEE Transactions On Information Theory*, v. 38, n. 2, pp. 1–13, March 1992.
- [40] Panych, L. P. Theoretical comparison of fourier and wavelet encoding in magnetic resonance imaging. *IEEE Transactions On Medical Imaging*, v. 15, n. 2, pp. 141–153, March 1996.
- [41] Zou, H., Lu, J., Greenleaf, F., Obtaining limited diffraction beams with the wavelet transform. In: *IEEE Ultrasonic Symposium*, Baltimore, MD, 1993.
- [42] Méxas, J. G. F., de Oliveira, A., Esperança, C., Reconstruction by using a wavelet representation of the algebraic radon transform. In: *SIB-GRAPI 2000*, Gramado, Brasil: IEEE Computer Society, October 2000.
- [43] Olson, T., Stefano, J. Wavelet localization of the radon transform. *IEEE Transactions on Signal Processing*, v. 42, n. 8, pp. 2055–2067, August 1994.
- [44] Delaney, A., Bresler, Y. Multiresolution tomographic reconstruction using wavelets. *IEEE Transactions on Image Processing*, v. 4, n. 6, pp. 799–813, June 1995.
- [45] Sahiner, B., Yagle, A. E. Image reconstruction from projections under wavelet constraints. *IEEE Transactions on Signal Processing*, v. 41, n. 12, pp. 3579–3584, December 1993.
- [46] Kolaczyk, E. D., A wavelet approach to tomographic image reconstruction. Preprint.
- [47] Deans, S. R., Radon and abel transforms. In: *The Transforms and Applications Handbook* ( Poularikas, A. D., ed.), pp. 631–717, CRC Press, 1995.
- [48] Strohmer, T., Computationally attractive reconstruction of band-limited images from irregular samples. preprint: to appear in *IEEE Transactions on Image Processing*.
- [49] Kawata, S., Nalcioglu, O. Constrained iterative reconstruction by the conjugate gradient method. *IEEE Transactions On Medical Imaging*, v. MI-4, n. 2, pp. 65–71, June 1985.

- [50] Noble, B., Daniel, J. W., *Álgebra Linear Aplicada*. 2<sup>a</sup> ed., Rio de Janeiro,RJ: Prentice Hall do Brasil, 1977.
- [51] Luenberg, D. G., *Introduction to Linear and Nonlinear Programing*. Wellesley-Cambridge Press, 1973.