

**SÉRGIO SHIGUEMI FURUIE**

**Eng.Eletrônico, ITA, 1977**

**MsC em Eng.Biomédica, COPPE/UFRJ, 1980**

**RECONSTRUÇÃO TOMOGRÁFICA DE IMAGENS COM RÚIDO POISSON: ESTIMATIVA DAS  
PROJEÇÕES**

**Tese apresentada à Escola Politécnica da USP  
—Departamento de Engenharia da Eletricidade —  
para a obtenção do título de Doutor em  
Engenharia**

**Orientador: Prof.Dr.Nelson Delfino d'Ávila Mascarenhas**

**Instituto de Pesquisas Espaciais**

**São Paulo, março de 1990**

## AGRADECIMENTOS

Gostaria de manifestar os meus agradecimentos a todos os amigos e amigas que tornaram possível a realização desta tese.

Em especial, agradeço:

- ao Nelson D. Mascarenhas, pesquisador do Departamento de Processamento de Imagens do INPE, pela sua dedicada e objetiva orientação;
- ao Takehiro Tomitani, pesquisador do National Institute of Radiological Sciences (NIRS/Japan), pela motivação em tomografia e pela confiança;
- ao André F. Kohn, professor da EPUSP, pelo incentivo e orientação inicial;
- ao Cândido P. Melo, Diretor da Coordenadoria de Informática do Instituto do Coração/HCFMUSP, pela amizade e incentivo;
- ao Lincoln A. Moura Jr, Diretor do Serviço de P&D da Coordenadoria de Informática/INCOR, pela amizade e apoio;
- aos meus colegas do Serviço de P&D, pela compreensão e respeito;
- ao Instituto do Coração pela infra-estrutura e oportunidade;

A todos, agradeço também a confiança depositada neste trabalho.

À Raquel e aos meus filhos Rafael e Vinicius.

## ERRATA

- 1) pág.8, parág. 1, linha 2: substituir " isótopos radioativos" por "radionuclídeos"  
parág.3, linha 1: "detetores colimados (câmeras de cintilação)" por "câmeras de cintilação"  
linha 2: "sensor" por "amostra"  
linha 8: "dosagem de" por "atividade do"  
linha 9: "planares" por "planas"  
linha 10:"detetor" por "pixel"  
linha 11:"detetor" por "amostra"  
linha 13:"detetores colimados" por "colimadores"  
linha 15:"detetor" por "posição"
- 2) pág.9, parág.1, linha 4: "efeito" por "devido, p. ex., ao efeito"
- 3) pág.11 e pág.17, linha 2 : "inversa da transformada" por "transformada inversa"
- 4) pág.27, parág.2,linha 4 : "é suposto" por "supõe-se"
- 5) pág.34, parág.2,linha 2 : "pontuais" por "pontuais"
- 6) pág.40, parág.4, linha 4 : "serem" por "ser"
- 7) pág.47, linha 5 : "pontual" por "pontual"
- 8) pág.64, linha 6 : "amostral" por "de amostragem"
- 9) pág.70, antepenúltima linha:"2.59" por "2.59  $\Delta T$ "
- 10) pág.97, últimas 6 linhas, coluna "tempo":  
Substituir "75.0" pelos valores abaixo:  
Pois\_MAP 78.25  
Pois\_Wiener 77.72  
Ansc\_MAP 80.03  
Ansc\_Heur. 82.12  
Ansc\_custo( $\beta=1$ ) 94.14  
Ansc\_custo( $\beta=5$ ) 94.14

## LISTA DE CONTEÚDOS

Agradecimentos .....	ii
Lista de abreviaturas .....	vi
Índice das tabelas .....	vii
Resumo .....	viii
Abstract .....	ix
<b>1-INTRODUÇÃO</b>	
1.1. Situando o problema e objetivos .....	2
1.2. Contribuição dada por esta tese .....	4
1.3. Notação .....	5
1.4. Organização desta tese .....	6
<b>2-TÉCNICAS DE RECONSTRUÇÃO PARA O CASO DE EMISSÃO</b>	
2.1 Reconstrução tomográfica e restauração de imagens .....	11
2.1.1 A transformada de Radon e a equação de Fredholm .....	12
2.2. Métodos da transformada .....	17
2.2.1. Aplicação Direta da Transformada de Fourier (DTF) .....	18
2.2.2. Convolução-Retroprojeção (CRP) e Filtragem-Retroprojeção (FRP) .....	20
2.3. Métodos algébricos - ART (Algebraic Reconstruction Technique) .....	21
2.4. Método estatístico ML (Maximum Likelihood) .....	23
2.4.1. Máxima verossimilhança pelo algoritmo EM (Expectation-Maximization) .....	23
2.5. Conclusões .....	25
<b>3-MODELOS</b>	
3.1. Modelo para o ruído: Poisson .....	29
3.2. Formação das projeções .....	30
3.3. Modelo para as projeções sem ruído Poisson .....	33
3.4. Modelo estatístico do sistema .....	35
3.5. Modelo aditivo para ruído Poisson .....	36

#### 4-METODOLOGIA PROPOSTA

4.1. Descrição do método proposto .....	41
4.2. Estimadores para processo Poisson .....	46
4.2.1. Maximum a Posteriori -MAP (Bayesiano) .....	46
4.2.2. Estimador local linear com menor erro médio quadrático (Wiener) .....	50
4.3. Estimativa envolvendo transformação não linear de Anscombe .....	53
4.3.1. Estimador Bayesiano MAP .....	58
4.3.2. Estimador de Wiener local .....	60
4.3.3. Estimador heurístico .....	61
4.3.4. Minimização de função custo (rugosidade) .....	63
4.4. Reconstrução DTF: refinamentos .....	69
4.4.1. Filtragem .....	69
4.4.2. Interpolação no domínio da frequência .....	72
4.4.3. Implementação do DTF .....	74
4.5. Conclusões .....	75

#### 5-RESULTADOS

5.1. Simulações .....	77
5.2. Medida de qualidade da reconstrução do cilindro .....	80
5.3. Reconstrução para projeções sem ruído .....	83
5.4. Reconstrução do cilindro na presença do ruído Poisson .....	88
5.5. Reconstrução de estruturas mais complexas na presença de ruído Poisson .....	99
5.6. Conclusões .....	104

6-CONCLUSÕES E DISCUSSÕES .....	105
---------------------------------	-----

#### 7-APÊNDICES

A) Probabilidade na detecção de fótons de um emissor que segue processo Poisson .....	107
B) Algumas propriedades do operador $E[.]$ em variáveis aleatórias multivariadas .....	108
C) Operador Laplaciano 1-dimensional .....	112

D) Estimativa por Máxima verossimilhança em ECT e sua relação com Pseudo Inversa .. 113

E) Listagem de parte dos programas desenvolvidos ..... 114

8-REFERÊNCIAS BIBLIOGRÁFICAS ..... 149

### LISTA DE ABREVIATURAS

ART	: Algebraic Reconstruction Technique
CBP	: Convolution Backprojection
CRP	: Convolução-Retroprojeção
DTF	: Aplicação Direta da Transformada de Fourier
E[.]	: valor esperado
ECT	: Emission Computerized Tomography
EM	: Expectation Maximization
fdp	: função densidade de probabilidade
FFT	: Fast Fourier Transform
FRP	: Filtragem-Retroprojeção
FWHM	: Full-Width at Half-Maximum
MAP	: Maximum a Posteriori
ML	: Maximum Likelihood
PET	: Positron Emission Tomography
PSF	: Point Spread Function
RFR	: Retroprojeção Filtrada com função Rampa
SPECT	: Single Photon Emission Computerized Tomography
SIPSF	: Space Invariant Point Spread Function
SVPSF	: Space Variant Point Spread Function
v.a	: variável aleatória
var[.]	: variância
cov[.]	: covariância

## ÍNDICE DAS TABELAS

Tabela 2.1 .....	25
Tabela 4.1 .....	68
Tabela 4.2 .....	68
Tabela 5.1 .....	87
Tabela 5.2 .....	97
Tabela 5.3 .....	103



## RESUMO

A reconstrução tomográfica de imagens com ruído Poisson tem grandes aplicações em medicina nuclear. A demanda por informações mais completas, como por exemplo, várias secções de um órgão, e a necessidade de reduzir a dosagem radioativa a que o paciente é submetido, requerem métodos adequados para a reconstrução de imagens com baixa contagem, no caso, baixa relação sinal/ruído. A abordagem estatística, utilizando a máxima verossimilhança (ML) e o algoritmo Expectation-Maximization (EM), produz melhores resultados do que os métodos tradicionais, pois incorpora a natureza estatística do ruído no seu modelo.

A presente tese apresenta uma solução alternativa, considerando também o modelo de ruído Poisson, que produz resultados comparáveis ao do ML-EM, porém com custo computacional bem menor.

A metodologia proposta consiste, basicamente, em se estimar as projeções, considerando o modelo de formação das projeções ruidosas, antes do processo da reconstrução. São discutidos vários estimadores otimizados, inclusive Bayesianos. Em especial, é mostrado que a transformação de ruído Poisson em ruído aditivo Gaussiano e independente do sinal (transformação de Anscombe), conjugada à estimativa, produz bons resultados. Se as projeções puderem ser consideradas, aproximadamente, transformadas de Radon da imagem a ser reconstruída, então pode ser aplicado um dos métodos da transformada para a reconstrução tomográfica. Dentre estes métodos, o da aplicação direta da transformada de Fourier foi avaliado mais detalhadamente devido ao seu grande potencial para reconstruções rápidas com processamento vetorial e processamento paralelo.

A avaliação do método proposto foi realizada através de simulações, onde foram geradas as imagens originais e as projeções com ruído Poisson. Os resultados foram comparados com métodos clássicos como a filtragem-retroprojeção, o ART e o ML-EM. Em particular, a transformação de Anscombe conjugada ao estimador heurístico (filtro de Maeda), mostrou resultados próximos aos do ML-EM, porém com tempo de processamento bem menor. Os resultados obtidos mostram a viabilidade da presente proposta vir a ser utilizada em aplicações clínicas na medicina nuclear.

## ABSTRACT

Tomographic reconstruction of images with Poisson noise is an important problem in nuclear medicine. The need for more complete information, like the reconstruction of several sections of an organ, and the necessity to reduce patient absorbed radioactivity, suggest better methods to reconstruct images with low-count and low signal-to-noise ratio. Statistical approaches using Maximum Likelihood (ML) and the Expectation-Maximization (EM) algorithm lead to better results than classical methods, since ML-EM considers in its model the stochastic nature of the noise.

This thesis presents an alternative solution, also using a Poisson noise model, that produces similar results as compared to ML-EM, but with much less computational cost.

The proposed technique basically consists of projection estimation before reconstruction, taking into account a model for the formation of the noisy projections. Several optimal and Bayesian estimators are analysed. It is shown that the transformation of Poisson noise into Gaussian additive and independent noise (Anscombe Transformation), followed by estimation, yields good results. If the projection can be assumed as Radon transform of the image to be reconstructed, then it is possible to reconstruct using one of the transform methods. Among these methods, the Direct Fourier Method was analysed in detail, due to its applicability for fast reconstruction using array processors and parallel processing.

Computer simulations were used in order to assess this proposed technique. Phantoms and phantom projections with Poisson noise were generated. The results were compared with traditional methods like Filtering-Backprojection, Algebraic Reconstruction Technique (ART) and ML-EM. Specifically, the Anscombe transformation together with a heuristic estimator (Maeda's filter) produced results comparable to ML-EM, but spending only a fraction of the processing time.

## 1-INTRODUÇÃO

1.1. Situando o problema e objetivos

1.2. Contribuição dada por esta tese

1.3. Notação

1.4. Organização desta tese

### 1.1 Situando o Problema e objetivos

A tomografia computadorizada (C.T.) tem sido aplicada em várias áreas, envolvendo desde dimensões moleculares (microscopia eletrônica) a dimensões cósmicas (radio astronomia). A capacidade de visualizar estruturas internas, como a secção transversal do corpo humano, transformou a C.T. em um instrumento inestimável para aplicações em medicina.

Especificamente, nas aplicações médicas, devido ao compromisso entre segurança do paciente, qualidade da imagem e tempo para reconstrução, a quase totalidade dos tomógrafos comerciais são destinados para imagens com elevada relação sinal/ruído (p.ex. raio-X), onde são empregadas as técnicas derivadas analiticamente, conhecidas por métodos da transformada (33).

No entanto, existe uma importante classe de fonte de informações metabólico/funcionais (radioisótopos, pósitrons), cujas imagens podem ter baixa relação sinal/ruído, e onde o ruído não é aditivo, e nem independente do sinal. Os tomógrafos do tipo PET (Positron Emission Tomography) e SPECT (Single Photon Emission Tomography) visam, frequentemente, reconstruir as imagens dentro do ambiente acima citado. Alguns equipamentos do tipo PET para elevada estatística (contagem da ordem de 5 milhões) fornecem imagens de boa qualidade. Porém, em medicina, os protocolos de exame buscam reduzir a dosagem a que o paciente é submetido. Nestas condições, as técnicas clássicas de inversão determinística já não são efetivas e precisam-se de modos alternativos para a solução. A natureza estocástica das medidas e a importância do ruído transformam o problema de reconstrução em um problema de estimação. As técnicas mais recentes (7) empregadas para reconstrução de imagens ruidosas ainda apresentam inconvenientes como lentidão, artefatos e perda de resolução.

O PET e o SPECT têm amplas aplicações clínicas (54) principalmente devido a sua capacidade de fornecer informações complementares metabólico/funcionais. No entanto, a demanda real é pela visualização de volumes e estruturas tri-dimensionais, que possibilitem o estudo de processos fisiológicos, a monitoração e diagnóstico de patologias, e o estudo dinâmico de estruturas móveis como o coração. A reconstrução de estruturas tri-dimensionais pode ser abordada por métodos específicos tri-dimensionais (2) ou por empilhamento de secções transversais. Neste sentido, a pesquisa em PET está orientada para sensores em múltiplos anéis paralelos, os quais envolvem a reconstrução de imagens de

várias secções, tornando fundamental um algoritmo de reconstrução rápido e eficiente.

Recentemente, trabalhamos com a incorporação de inovações tecnológicas em C.T. (15), como a utilização de "array processors" com aplicação direta da transformada de Fourier. Apesar desta abordagem reduzir o tempo de reconstrução em quase duas ordens de grandeza comparado ao clássico Convolução-Retroprojeção (CRP), a mesma ainda apresenta resultados inaceitáveis em imagens com baixa relação sinal/ruído.

O processamento digital de imagens tem, classicamente, assumido que o ruído é aditivo, independente do sinal, e o sistema é linear, possibilitando a aplicação da teoria estatística de estimação linear, incluindo o filtro de Wiener. Recentemente (26) a atenção tem sido voltada a modelos de formação de imagens mais realísticos, onde o ruído é modelado como dependente do sinal, como é o caso de ruídos em filmes, e ruídos fotoeletrônicos a baixo nível de intensidade. Nestas situações, se torna, muitas vezes, imperativo o emprego de estimadores não-lineares para um melhor desempenho em termos de menor erro médio quadrático na obtenção da imagem original. Por outro lado, as imagens são melhores modeladas se consideradas com média variante no espaço. Deste modo, o estimador também deve considerar aspectos locais (adaptividade) e ser robusto no sentido de ser pouco sensível a distribuições particulares da imagem.

O objetivo do presente trabalho é propor uma técnica alternativa rápida para a reconstrução tomográfica de imagens com ruído Poisson, cuja relação sinal/ruído é baixa. São formulados modelos de ruído para a imagem a ser reconstruída e para a formação das projeções, e a partir destes modelos é derivada a proposta de se estimar as projeções (pré-processar) antes da reconstrução. São propostos alguns estimadores ótimos, inclusive após a aplicação da transformação de Anscombe (3) sobre as projeções. Em relação ao método de reconstrução, é avaliado o método da aplicação Direta da Transformada de Fourier (DTF) com alguns refinamentos. O resultado do conjunto destas propostas é comparado com outros métodos como o clássico CRP, ART(Algebraic Reconstruction Technique) e o EM(Expectation-Maximization), sobre projeções simuladas de PET e SPECT, supondo que as mesmas já foram devidamente corrigidas quanto à atenuação, espalhamento e distorções devido aos instrumentos de medida.

### 1.1 Contribuição dada por esta tese

Dentro do modelo estabelecido para a formação das projeções, mostra-se que o problema da reconstrução pode ser resolvido em duas fases: estimando as projeções e então reconstruindo a imagem. Esta abordagem permite filtragem mais efetiva das projeções, pois pode-se utilizar filtros ótimos contra o ruído Poisson, além de ainda permitir o emprego de técnicas rápidas para a reconstrução como o DTF, se as projeções puderem ser consideradas transformada de Radon da imagem a ser reconstruída.

Neste trabalho, mostra-se também a eficácia da transformação de Anscombe na reconstrução, a qual transforma ruídos Poisson, que são dependentes do sinal, em ruídos aditivos e independentes do sinal.

Dentre os diversos filtros para as projeções, é proposto um filtro que minimiza a função custo, a qual envolve a norma do erro da estimação e a rugosidade do sinal a ser estimado. Este filtro é implementável utilizando o FFT, e tem desempenho comparável a outros estimadores ótimos, com vantagem adicional de poder utilizar, parcialmente, a informação eventualmente disponível sobre a rugosidade das projeções.

Em relação à técnica de reconstrução são propostos refinamentos ao DTF de modo a melhorar a qualidade da sua reconstrução, uma vez que esta técnica é mais rápida do que a filtragem-retroprojeção, para imagens maiores do que 64 x 64.

A reconstrução de imagens com ruído Poisson pelo método da máxima verossimilhança (ML), e implementada pelo algoritmo Expectation-Maximization (EM) produz bons resultados, pois incorpora a natureza estatística do ruído no seu modelo. O presente trabalho apresenta uma solução alternativa, que também considera a natureza estocástica do ruído, com qualidade comparável ao ML-EM, porém com custo computacional bem menor.

## 1.2 Notação

O vetor será representado por letras minúsculas com uma seta sobre as mesmas ( $\vec{f}$ ). Cada elemento do vetor será referenciado por um índice. Desse modo ,

$$\vec{f}^T = [f_1 \ f_2 \ \dots \ f_n] \quad \text{onde } (\cdot)^T \text{ é o operador transposto}$$

As imagens, no seu formato bi-dimensional, e as matrizes serão representadas por letras maiúsculas (H). O elemento na linha i e coluna j da matriz H será referenciado por  $H_{ij}$ . Para facilitar a representação matemática de operações entre imagens (matrizes), as mesmas também serão representadas por vetores (4). As nlin linhas de uma imagem com ncol elementos (colunas) cada, serão empilhadas, linha a linha, de modo a formar um vetor com  $N = nlin \times ncol$  elementos, mantendo, deste modo, a informação espacial. Sendo  $F_{nlin \times ncol}$  esta matriz (imagem), tem-se:

$\vec{f}_N$  representa também esta matriz, onde  $N = nlin \cdot ncol$  e

$$f_i = F_{lc} \quad \text{onde } i = (l-1) \cdot ncol + c \quad c = 1, ncol \quad l = 1, nlin$$

Dessa forma a aplicação de um operador projeção (H) ou de uma matriz de degradação variante no espaço sobre uma imagem  $\vec{f}$  pode ser representada, simplesmente, por:

$$\vec{y} = H\vec{f}$$

As variáveis aleatórias (v.a) serão distinguidas das amostras das mesmas, denotando as v.a. por letras maiúsculas com um til e as amostras por minúsculas. Dessa forma, a  $\text{Prob}[\tilde{X}=x \mid \vec{f}]$  significa a probabilidade da v.a  $\tilde{X}$  ser igual a x dado o vetor  $\vec{f}$ . Pelo contexto, não havendo a possibilidade de se confundir com a notação de matriz, a v.a. será denotada sem o til.

$E[X]$  : valor esperado de  $\tilde{X}$

$\text{var}[X]$  : variância de  $\tilde{X}$

$\text{cov}[\vec{Y}]$  : matriz de covariância de  $\vec{Y}$

### 1.3 Organização desta tese

No capítulo 2 são descritas algumas das principais técnicas de reconstrução de imagens. Mostra-se que os problemas da reconstrução e restauração de imagens podem ser formulados de uma forma unificada.

No capítulo 3 é delimitado o problema para imagens com ruído Poisson. Descreve-se o modelo para a formação das projeções e um modelo estatístico do sistema. É desenvolvido, de forma detalhada, o modelo aditivo para ruído Poisson, no sentido de se justificar alguns estimadores ótimos propostos nas seções 4.2 e 4.3.

No capítulo 4 são propostos alguns estimadores para as projeções e os refinamentos para o método de reconstrução DTF. Mostra-se que a aplicação de estimadores em conjunto com a transformação não-linear de Anscombe produz bons resultados.

No capítulo 5 são avaliados os diversos estimadores no contexto da reconstrução completa, e comparados, qualitativa e quantitativamente, a outros métodos como a Filtragem-Retroprojeção, o método algébrico ART e o estatístico ML-EM.

O capítulo 6 traz as principais conclusões desta tese, como também discussões e algumas sugestões para investigação futura.

Nos apêndices estão reunidos algumas demonstrações, propriedades e listagens da parte significativa dos programas.



## 2-TÉCNICAS DE RECONSTRUÇÃO PARA O CASO DE EMISSÃO

### 2.1 Reconstrução tomográfica e restauração de imagens

#### 2.1.1 A transformada de Radon e a equação de Fredholm

### 2.2. Métodos da transformada

#### 2.2.1. Aplicação Direta da Transformada de Fourier (DTF)

#### 2.2.2. Convolução-Retroprojeção (CRP) e Filtragem-Retroprojeção (FRP)

### 2.3. Métodos algébricos - ART (Algebraic Reconstruction Technique)

### 2.4. Método estatístico ML (Maximum Likelihood)

#### 2.4.1. Máxima verossimilhança pelo algoritmo EM (Expectation-Maximization)

### 2.5. Conclusões

As imagens e projeções em medicina nuclear são obtidas (57) administrando-se, ao paciente, compostos contendo isótopos radioativos, onde a distribuição destes compostos ocorre de acordo com processos fisiológicos e metabólicos. A natureza, o local e a forma de administração (oral, venosa, arterial,..) das substâncias radioativas são determinadas pelo órgão ou região a ser estudada. Desse modo, os fótons emitidos fornecem informações sobre a atividade da região.

A tomografia consiste em se estimar a distribuição de um parâmetro físico interno a partir das observações medidas externamente. No caso da medicina nuclear, a tomografia por emissão (ECT) abrange o SPECT (tomografia por emissão de fóton único) e o PET (tomografia por emissão de pósitrons).

No SPECT, as projeções (imagens) ao redor do objeto são obtidas por detectores colimados (câmeras de cintilação). Nessas condições, os feixes correspondentes a cada sensor, para um dado ângulo de projeção, são paralelos entre si. As gama câmeras atuais (43) têm um diâmetro da ordem de 40 cm e resolução intrínseca da ordem de 3 mm . Devido ao espalhamento e o uso do colimador a resolução efetiva é , aproximadamente, de 10 mm. Pela teoria da amostragem a matriz da imagem deveria ter dimensão maior ou igual a  $2 \cdot 400 / 10 = 80$ . Em geral, usam-se matrizes de  $64 \times 64$ . Desse modo, cada pixel corresponde a aproximadamente 6 mm. A contagem total de fótons por ângulo de projeção está limitada ao nível de dosagem de elemento radioativo que pode ser injetada no paciente, à eficiência dos detectores e ao tempo do exame. Valores típicos para imagens planares são de  $10^4$  a  $10^6$  (média por detector: 2.5 a 250 contagens), e para projeções tomográficas são de  $10^4$  a  $10^5$  por ângulo de projeção ( média por detector : 2.5 a 25). As gama câmeras são razoavelmente lineares na baixa taxa de contagem usualmente empregada ( menor do que 50000 contagens/s). Para o caso da reconstrução bi-dimensional, pode-se formular o problema de acordo com a figura 2.1 , onde um conjunto de detectores colimados recebem os fótons provenientes de feixes paralelos para cada ângulo de projeção  $\theta$ , onde um sistema de aquisição acumula as contagens para cada detector. Girando-se este conjunto em torno do objeto , obtêm-se as projeções necessárias para a reconstrução.

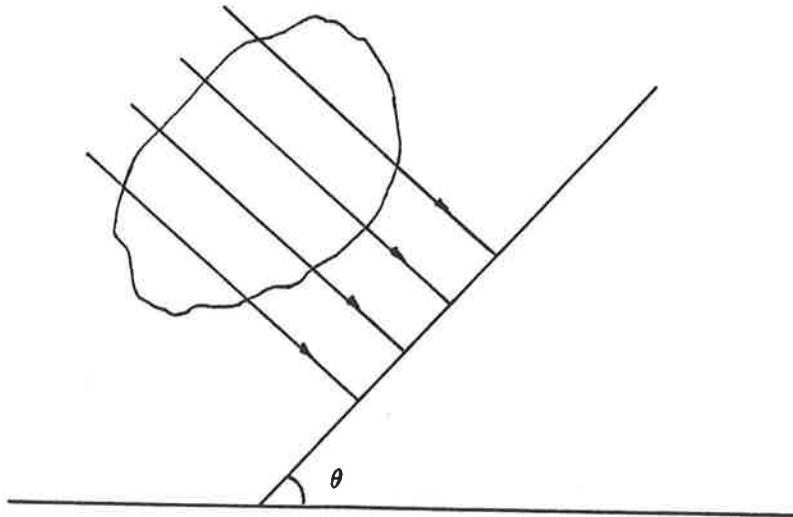


Figura 2.1 Esquema para aquisição das projeções em SPECT

O processo de formação das projeções em SPECT difere do caso ideal de uma integral de linha (43). A utilização direta dos métodos da transformada em SPECT apresenta os seguintes problemas:

- a) a estatística da contagem (ruído) é ignorada;
- b) todos os pontos ao longo do feixe têm o mesmo peso, i.é, a atenuação (efeito de absorção fotoelétrica) não é considerada;
- c) assume-se que a origem de todas as contagens vem de pixels ao longo do feixe, negligenciando o espalhamento (efeito Compton) e a resposta dos colimadores;

A correção da atenuação pode ser obtida, aproximadamente, por alguns métodos, como o de Sorenson e de Chang (43), que objetivam corrigir as projeções para a reconstrução, ou então a imagem já reconstruída.

No PET cada pósitron emitido é rapidamente aniquilado por um elétron próximo, gerando um par de fótons que se propagam, com a velocidade da luz, em direções quase que diametralmente opostas. A direção desta linha de propagação é aleatória e uniformemente distribuída no espaço (60). Utilizam-se um ou mais anéis de detectores discretos em torno do objeto a ser reconstruído (figura 2.2) onde a colimação é conseguida pela imposição da quase simultaneidade na detecção. Através de um

sistema computacional, acumulam-se as contagens para cada linha, obtendo-se as projeções, que serão então utilizadas para a reconstrução. Em PETs de alta estatística, tem-se da ordem de 2 a 5 milhões de contagens totais ,com média de 300 a 600 contagens/par de detetores.

No entanto , o PET e o SPECT sofrem influencia da atenuação e do espalhamento dos fótons , tornando a reconstrução, parcialmente, dependente do objeto. No caso do PET, a atenuação pode ser corrigida devido ao princípio físico envolvido (60). Considerando-se na figura 2.2, a direção  $i$  , se uma fonte emissora distando  $d$  do seu contorno , e sendo  $\mu$  o fator de atenuação (suposto constante), a probabilidade do sensor  $A$  detetar a emissão do pósitron em  $P$  é decrescida de  $e^{-\mu d}$  e pelo sensor  $B$  por  $e^{-\mu(L-d)}$  sendo  $L$  a trajetória total do feixe  $i$  no objeto.

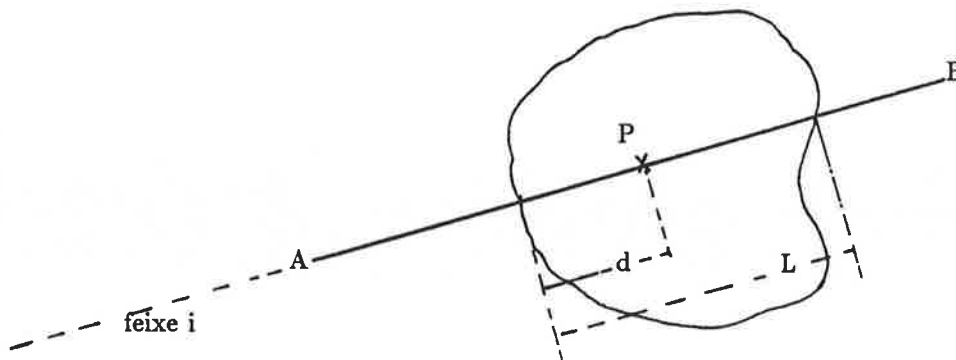


Figura 2.2 Atenuação em PET

Desse modo, a redução na detecção da simultaneidade dos fótons em  $A$  e  $B$  é dada por  $e^{-\mu d - \mu(L-d)} = e^{-\mu L}$ , e portanto, independente da posição do emissor. Basta conhecer o fator de atenuação e o contorno do objeto para efetuar-se a correção da atenuação nas projeções. O contorno pode ser obtido utilizando-se as próprias projeções em ECT (58). A correção da atenuação também pode ser obtida por uma sequência de duas medidas anteriores ao exame propriamente dito, buscando-se medir o fator de correção em cada projeção (60).

## 2.1 Reconstrução tomográfica e restauração de imagens

A tomografia é, basicamente, a solução do problema inverso, seja esta obtida através da inversa da transformada de Radon, ou então pela solução de um sistema de equações algébricas lineares ou não-lineares.

As técnicas de reconstrução de imagens podem ser divididas em várias classes: determinística, estocástica ou híbrida; linear ou não-linear; direta, iterativa ou recursiva; implementada no domínio do espaço ou no domínio da frequência. As determinísticas utilizam apenas as informações observadas, enquanto que as estocásticas usam, adicionalmente, um modelo estatístico do problema.

As técnicas mais usuais são as determinísticas e estocásticas. As determinísticas podem ser subdivididas em: métodos da transformada e os métodos da expansão em série (33). O método da transformada é uma formulação analítica do problema, sendo discretizado apenas para implementação digital, enquanto que o da expansão em série é uma abordagem discreta desde o início, relacionando um conjunto de dados conhecidos (projeções), a outro conjunto de incógnitas (imagem a ser reconstruída), resultando em um sistema de equações, cuja solução é obtida numericamente. As estocásticas são obtidas a partir de formulações estatísticas do problema, como por exemplo a solução de máxima verossimilhança.

Neste trabalho será assumido que, para cada ângulo de projeção, os feixes são paralelos. Tomografia por feixes divergentes pode ser abordada como uma extensão de feixes paralelos. No caso do PET, que utiliza anéis de sensores, antes da reconstrução os feixes são agrupados formando conjunto de raios paralelos para cada ângulo de projeção.

Nas próximas seções serão descritos, brevemente, dois métodos da transformada ( aplicação Direta da Transformada de Fourier e Filtragem-Retroprojeção ), um método da expansão em série (reconstrução algébrica ART) e um método estocástico (máxima verossimilhança - ML) implementado de forma iterativa pelo método Expectation-Maximization (EM).

Inicialmente será mostrado, analiticamente, que a clássica transformada de Radon empregada em tomografia, é um caso particular da equação de Fredholm utilizada para a descrição de um sistema

linear geral. Desta forma, pode-se unificar os métodos empregados em restauração e reconstrução de imagens.

### 2.1.1 A Transformada de Radon e a equação de Fredholm

Um sistema geral que representa vários processos de degradação de imagens pode ser descrito por um modelo linear (56), como o da figura 2.3.

Analicamente, este sistema pode ser descrito por (4):

$$g(\alpha, \beta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \cdot h(\alpha, \beta, x, y) \, dx \, dy \quad (2.1)$$

A integral da equação 2.1 é chamada de integral de Fredholm do primeiro tipo e representa um sistema linear geral, variante no espaço (SVPSF).

A função  $h(\alpha, \beta, x, y)$  é a resposta impulsiva do sistema linear (PSF). Se o PSF for invariante no espaço, tem-se :

$$h(\alpha, \beta, x, y) = h(\alpha - x, \beta - y) \quad (2.2)$$

Neste caso particular ( sistema linear invariante no espaço - SIPSF) a equação 2.1 se torna uma convolução em 2 dimensões, a qual pode então ser manipulada facilmente no domínio da frequência, utilizando a transformada de Fourier bi-dimensional.

A transformada de Radon (2) aplicada em tomografia por Raio-X, corresponde a (figura 2.4):

$$g(t, \theta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \cdot \delta[t - (x \cdot \cos\theta + y \cdot \sin\theta)] \, dx \, dy \quad (2.3)$$

onde  $\delta$  é a função delta de Dirac.

A equação 2.3 é a integral de linha sobre a reta L (figura 2.4), a qual dista 't' da origem 'O' e faz um ângulo  $\theta$  com o eixo y.

Ou seja,

$$g(t, \theta) = \int_L f(s) \cdot ds \quad (2.4)$$

Comparando-se as eq. (2.1) e (2.3), pode-se concluir que a transformada de Radon para o caso bi-dimensional é um caso particular da função de transferência de um SVPSF, ou seja,

$$h(t, \theta, x, y) = \delta[t - (x \cdot \cos\theta + y \cdot \sin\theta)] \quad (2.5)$$

Portanto a "imagem"  $g(t, \theta)$  formada pelas projeções pode ser vista como uma imagem borrada de 'f' onde o sistema tem resposta impulsiva descrita pela equação 2.5.

Os métodos da transformada buscam obter a transformada inversa de Radon, implementadas em diferentes modos : aplicação direta da transformada de Fourier (DTF), convolução e retroprojeção (CRP), filtragem rho (2), etc. No entanto, ressalte-se que estes métodos pressupõem que as projeções são obtidas a partir da transformada de Radon sobre a imagem. Esta condição não é válida para tomografia por emissão, a menos que haja correção efetiva da atenuação. Desse modo, o PET se adequa melhor ao método das transformadas, do que o SPECT, pois além de ser possível corrigir as projeções com mais precisão quanto à atenuação, a sua colimação é melhor.

Os métodos determinísticos da expansão em série em tomografia estão relacionados com a versão discretizada da equação de Fredholm. Esta discretização traz consigo erros devido à quantização, e a solução do sistema se torna, na maioria das vezes, não única ou indeterminada (2), agravada com a presença de ruídos.

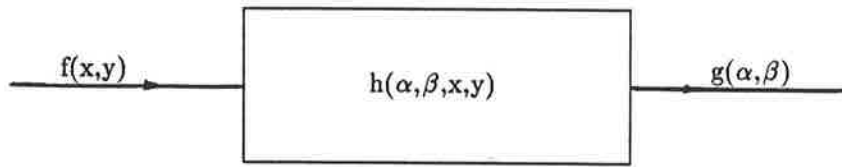


Figura 2.3 Modelo linear de degradação usado em restauração onde  $f(x,y)$  representa a imagem original e  $g(\alpha,\beta)$  a imagem degradada.

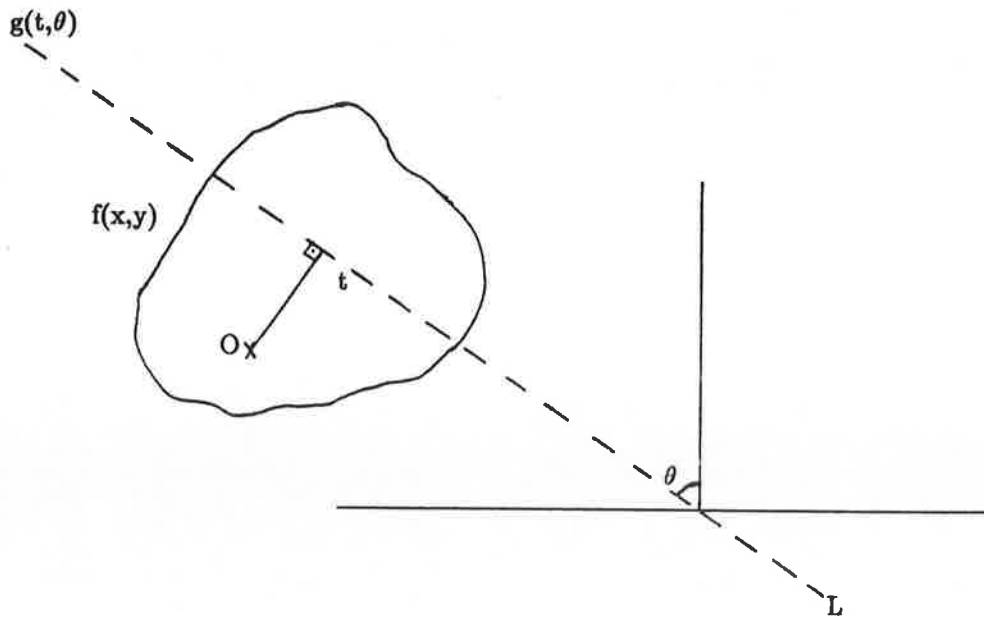


Figura 2.4 Modelo usado em reconstrução (transmissão) da imagem  $f(x,y)$  ,onde é válida a integral de linha para descrever as projeções  $g(t,\theta)$



Discretizando-se o modelo da figura (2.3), tem-se:

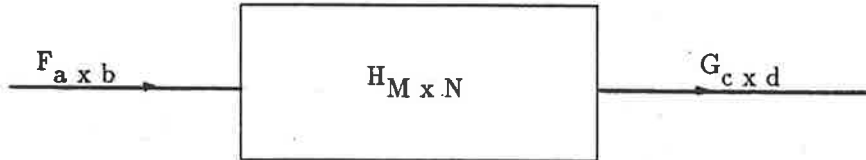


Figura 2.5 Modelo discretizado para restauração e reconstrução

onde :  $F_{a \times b}$  : matriz de dimensões  $a \times b$ ;

$G_{c \times d}$  : matriz de dimensões  $c \times d$ ;

$$F_{ij} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y) \cdot \delta(x-i.\Delta x, y-j.\Delta y). dx dy \quad (2.6)$$

Escrevendo sob a forma vetorial (4), tem-se:

$$\vec{g}_{M \times 1} = H_{M \times N} \cdot \vec{f}_{N \times 1} \quad M=c.d \quad N=a.b \quad (2.7)$$

onde  $\vec{f}$  vetor que contém os  $N$  pixels de  $F$ , empilhados linha a linha

( vetor no espaço Euclideano  $N$ -dimensional).

$\vec{g}$  idem para  $G$

$H$  matriz de transferência do sistema. Cada coluna 'j' de  $H$  representa o PSF correspondente ao j-ésimo pixel de ' $\vec{f}$ '. Ressalte-se que a dimensionalidade desta matriz é elevada, da ordem de  $10^4$  a  $10^5$ , sendo no entanto esparsa. Esta matriz pode também ser interpretada de outro modo: cada linha 'i' de  $H$  corresponde aos pesos dados a cada pixel da imagem original para se formar o pixel 'i' da imagem degradada ou projeção.

A equação 2.7 engloba a versão discretizada da restauração e reconstrução. Se o sistema for SIPSF, a matriz 'H' pode ser aproximada por uma matriz de blocos circulante (4), e portanto, utilizar o FFT para o cálculo da inversa. No caso da transformada de Radon, a matriz 'H' é constituída por '0's e '1's, sendo que os elementos '1's estão esparsos.

Para o caso do ECT, cada elemento 'j' da linha 'i' da matriz 'H' é constituída pela probabilidade do pixel 'j' da imagem objeto contribuir na projeção 'i' (60).

A rigor, a equação que descreve as projeções no caso ECT difere da equação 2.3, sendo dada por (50):

$$g(t, \theta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \cdot p(x, y, t) \cdot \delta[t - (x \cdot \cos \theta + y \cdot \sin \theta)] dx dy \quad (2.8)$$

onde  $p(x, y, t)$  é a probabilidade de que uma linha, cujo ângulo é uniformemente distribuído no espaço, passando por  $(x, y)$  na imagem, atinja o detector em 't'.

Assumindo-se que esta probabilidade é constante ao longo do feixe em 't', recai-se na integral de linha, podendo, desta forma, utilizar os procedimentos usuais de reconstrução empregados em CT por transmissão. O erro devido ao ruído estatístico é bem mais sério do que os erros introduzidos por esta simplificação (50).

No entanto, métodos determinísticos iterativos como o ART, ou métodos estocásticos como o ML-EM, podem incorporar estes fatores na matriz H e, portanto, relaxar os requisitos necessários de amostragem para os métodos analíticos, que pressupõem fenômenos satisfazendo a integral de linha.

## 2.2 Métodos da Transformada

Os métodos da transformada são soluções analíticas para a obtenção da inversa da transformada de Radon e envolvem: a) formulação analítica (funções contínuas ) de um modelo do problema; b) determinação da solução inversa ; c) adaptação desta fórmula para dados discretos (33). Estes métodos são válidos apenas para fenômenos onde as projeções podem ser descritas pela integral de linha, como é o caso, aproximadamente, da tomografia por transmissão com raio-X (33). Estes métodos podem ser empregados em ECT, desde que sejam tomadas as devidas precauções para que a equação 2.3 seja válida, ou seja,

- a) utilizando medidas com elevada relação sinal/ruído;
- b) corrigindo a atenuação nas projeções;
- c) melhorando a resposta dos colimadores;

Os métodos da transformada apresentam como vantagens:

- a) rapidez na reconstrução, pois são métodos não-iterativos;
- b) essas técnicas podem reconstruir razoavelmente bem, mesmo em situações onde há menos projeções do que pixels (sistema sub-determinado) (32).

Nas próximas seções serão discutidos : o método da retroprojeção filtrada, do qual CRP é uma das formas de implementação (clássica) e a aplicação Direta da Transformada de Fourier (DTF).

**2.2.1 Aplicação Direta da Transformada de Fourier (DTF)**

Este método é a aplicação direta do teorema das projeções, onde se demonstra (39) que a transformada de Fourier bi-dimensional de uma imagem  $f(x,y)$  está diretamente relacionada com as transformadas de Fourier uni-dimensional das projeções, projeções estas obtidas pela integral de linha (equação 2.3).

Seja a figura 2.4, onde a imagem é representada pela função  $f(x,y)$ . A transformada de Radon de  $f(x,y)$  para o ângulo  $\theta$  e distância  $t$  é dada por:

$$g(t;\theta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y) \cdot \delta(x \cdot \cos\theta + y \cdot \sin\theta - t) \cdot dx \cdot dy \tag{2.9}$$

A transformada de Fourier uni-dimensional de  $g(t;\theta)$  na primeira variável,

$$\begin{aligned} G_1(\rho;\theta) &= \mathcal{F}_1\{g(t;\theta)\} = \int_{-\infty}^{\infty} g(t,\theta) \cdot e^{-j2\pi\rho t} \cdot dt \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y) \left[ \int_{-\infty}^{\infty} \delta(x \cdot \cos\theta + y \cdot \sin\theta - t) \cdot e^{-j2\pi\rho t} \cdot dt \right] dx \cdot dy \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y) \cdot e^{-j2\pi\rho(x \cdot \cos\theta + y \cdot \sin\theta)} dx \cdot dy \end{aligned} \tag{2.10}$$

Como a transformada de Fourier bi-dimensional de  $f(x,y)$  em coordenadas cartesianas, é dada por:

$$F(u,v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y) \cdot e^{-j2\pi(ux+vy)} \cdot dx \cdot dy \tag{2.11}$$

Segue,

$$G_1(\rho;\theta) = F(u,v) \quad \text{onde } u = \rho \cdot \cos\theta \tag{2.12}$$

$$v = \rho \cdot \sin\theta$$

A imagem reconstruída pode então ser obtida por:

$$f(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u,v) \cdot e^{j2\pi(ux+vy)} du \cdot dv \tag{2.13}$$

O método DTF consiste em utilizar esta última equação (2.13) diretamente, uma vez obtida a função  $F(u,v)$  pelas equações 2.10 e 2.12. Deste modo o algoritmo consiste, basicamente, em:

- a) obter a transformada de Fourier das projeções para cada ângulo de projeção  $\theta$ . Devido à natureza discreta das projeções e a disponibilidade de algoritmos rápidos para o cálculo da série de Fourier, usa-se o FFT para o cálculo da transformada de Fourier das projeções.
- b) obter a transformada de Fourier bi-dimensional em coordenadas cartesianas  $F(u,v)$  (equação 2.12). Como dispõe-se de amostras da função  $F(u,v)$  em coordenadas polares, torna-se necessário estimar os valores (complexos) de  $F(u,v)$  em pontos discretos no sistema de coordenadas cartesianas. Esta estimativa é crítica, pois os erros resultantes estarão presentes em toda a imagem reconstruída, uma vez que a transformada de Fourier é um operador global.
- c) obter a transformada de Fourier inversa bi-dimensional de  $F(u,v)$ , resultando na imagem reconstruída  $f(x,y)$ .

Entre os passos a) e b) ou entre b) e c) podem ser aplicados filtros de modo a reduzir os efeitos de ruídos de alta frequência.

O método DTF tem sido pouco utilizado justamente devido às dificuldades na interpolação mencionadas no item b) e à aproximação da transformada de Fourier inversa implementada pelo FFT. No entanto, tomando-se algumas precauções, este método pode apresentar resultados com qualidade comparável ao CRP, e ainda com desempenho melhor quanto à velocidade para imagens de dimensões maiores (33).

Na seção 4.4 são propostos refinamentos a este método, de modo a torná-lo mais preciso.

Em termos de número de operações de multiplicação, para uma imagem  $n \times n$ , com  $n$  amostras por ângulo de projeção e  $m$  ângulos, este método envolve, aproximadamente:

$$m(\text{FFT-1D}) + \text{interpolação} + (\text{FFT-2D}) \text{ inversa} = m \cdot n \log_2 n + 5n^2 + 2n(n \log_2 n) = n(m+2n) \log_2 n + 5n^2$$

Para  $m=2n$ , tem-se  $n^2 \cdot (4 \log_2 n + 5)$  operações de multiplicação.

### 2.2.2 Convolução-Retroprojeção(CRP) e Filtragem-retroprojeção (FRP)

Convolução-Retroprojeção consiste, essencialmente, em filtrar as projeções de cada ângulo de projeção através da convolução no domínio do espaço, e retroprojetar os resultados sobre a imagem. A sua derivação pode ser obtida diretamente pela substituição da equação 2.12 na equação 2.13. Após a mudança de variáveis, tem-se:

$$f(x,y) = \int_0^{\pi} \int_{-\infty}^{\infty} F(\rho \cos \theta, \rho \sin \theta) \cdot e^{j2\pi \rho t} \cdot |\rho| \cdot d\rho \cdot d\theta \quad \text{onde } t = x \cdot \cos \theta + y \cdot \sin \theta \quad (2.14)$$

$$= \int_0^{\pi} \left[ \int_{-\infty}^{\infty} G_1(\rho; \theta) \cdot |\rho| \cdot e^{j2\pi \rho t} \cdot d\rho \right] \cdot d\theta \quad (2.15)$$

$$= \int_0^{\pi} [g(t; \theta) * \mathcal{F}^{-1}\{|\rho|\}] \cdot d\theta \quad (2.16)$$

a qual corresponde, portanto, à convolução das projeções, para cada ângulo de projeção, com a inversa da transformada de Fourier de uma rampa, seguida da retroprojeção.

O número de operações de multiplicação envolvido, para uma imagem  $n \times n$ , com  $n$  amostras por ângulo de projeção e  $m$  ângulos, é de , aproximadamente:

$$m(\text{convoluções}) + n^2 \cdot \text{retroprojeções} = m(n \cdot n) + n^2(m \text{ somas})$$

Para  $m=2n$ , têm-se aproximadamente  $2n^3$  operações de multiplicação. Na realidade, o número de operações depende do filtro. Se o mesmo tiver poucos coeficientes diferentes de zero, o número de operações de multiplicação torna-se proporcional a  $n^2$ .

Uma alternativa para o CRP clássico, é utilizar a equação 2.15 diretamente, método este conhecido por Filtragem-Retroprojeção, onde:

- a) calcula-se  $G_1(\rho; \theta)$  para todos os  $\theta$ , utilizando o FFT uni-dimensional;
- b) filtra-se cada  $g(t; \theta)$  no domínio da frequência, que consiste em simples multiplicação por  $|\rho|$ ;
- c) para eliminar os efeitos de ruídos de alta frequência, utilizam filtros simples do tipo Hamming, Hann ou Parzen (43);
- d) obtem-se a inversa da transformada de Fourier uni-dimensional dos resultados de c);
- e) para cada pixel  $(x,y)$  calcula-se a retroprojeção entre 0 e  $\pi$ , realizando-se interpolações onde

necessário, uma vez que as projeções filtradas são definidas apenas em pontos discretos.

Neste caso, para as mesmas condições de projeção do CRP, são necessárias:

$$m(\text{FFT1D}+n(\text{multiplicações complexas})+\text{FFT1D inversa})+n^2(\text{retroprojeções})=$$

$$m(n.\log_2 n+2n + n\log_2 n) (\text{multiplicações}) + n^2.(m \text{ somas})$$

Na prática, usa-se expansão com zeros nas projeções, de modo a duplicar o número de pontos no domínio da frequência (33). Nestas condições, em termos de operações de multiplicação, tem-se:

$$m.(2n\log_2 2n+4n+2n\log_2 2n)$$

Para  $m=2n$ , tem-se  $8n^2.(\log_2 n+2)$  multiplicações.

### 2.3 Método Algébrico ART (Algebraic Reconstruction Technique)

O método ART objetiva a solução do sistema de equações lineares expressa pela equação 2.7. Como a matriz  $H$  é de elevadas dimensões, Gordon, Bender e Herman propuseram, em 1970, um método iterativo para a reconstrução, o qual equivale ao algoritmo de Kaczmarz para a solução de sistemas de equações lineares (7). Re-escrevendo a equação 2.7, tem-se

$$\vec{g}_M = H_{M \times N} \cdot \vec{f}_N \quad \text{onde}$$

$\vec{g}_M$  : representa as  $M$  projeções medidas;

$H$  : matriz de projeção, a qual é suposta conhecida.

O método consiste em buscar, iterativamente, a solução  $\vec{f}$ , a partir de um valor inicial arbitrário  $\vec{f}^0$ . Em cada iteração  $k$ , a imagem obtida  $\vec{f}^{(k)}$  é refinada para  $\vec{f}^{(k+1)}$ , considerando uma única projeção, por exemplo  $i$ . A discrepância entre o valor medido  $g_i$  e a reprojeção de  $\vec{f}^k$  em  $i$  é distribuída proporcionalmente sobre a imagem, de acordo com os pesos  $H_{i,j}$  para cada pixel  $j$ . O algoritmo pode ser resumido nos seguintes passos (7):

a) inicialização:  $\vec{f}^0 \in \mathbb{R}^N$  é arbitrário

$k=0$

b)  $i=(k \text{ mod } M) + 1$

$$\vec{f}^{(k+1)} = \vec{f}^{(k)} + \frac{(g_i - \langle \vec{f}^{(k)}, \vec{h}_i \rangle)}{\|\vec{h}_i\|^2} \cdot \vec{h}_i$$

(2.17)

onde  $\vec{h}_i^T$  : corresponde a i-ésima linha de H ( $\vec{h}_i \in R^N$ )

$$\langle \vec{f}^k, \vec{h}_i \rangle = \sum_{j=1}^N H_{i,j} \cdot f_j^{(k)} : \text{reprojeção de } \vec{f}^{(k)} \text{ em } i \text{ (produto interno)}$$
$$\|\vec{h}_i\|^2 = \langle \vec{h}_i, \vec{h}_i \rangle$$

c)  $k=k+1$

No capítulo 5 são apresentados os resultados obtidos por este método para projeções sem e com ruído. Para efeito de comparação, foi considerado como iteração completa, um ciclo de iterações completo (M iterações).

Os métodos da expansão em séries apresentam as seguintes vantagens sobre os métodos da transformada (7):

- . podem incorporar facilmente conhecimentos a priori e as especificidades geométricas de novos métodos ou equipamentos;
- . fornecem imagens de contraste mais elevado;

As desvantagens são:

- . lentos quando comparados aos métodos da transformada. Nas condições da imagem da seção 2.2.1, tem-se, aproximadamente, em termos de operações de multiplicação,

$$m.n(\text{reprojeções}) + m.n(\text{estimativas}) = m.n.n + m.n.2n$$

Para  $m=2n$ , tem-se aproximadamente  $5n^3$  multiplicações.

- . à medida que a iteração prossegue, apresenta instabilidades devido ao ruído e/ou à inconsistência do problema;



**2.4 Método estatístico ML (Maximum Likelihood)**

Considerando a figura 3.4 e o modelo de formação das projeções discutido na seção 3.2, pode-se determinar a solução com a maior probabilidade condicional (verossimilhança) de se obter as projeções  $\vec{y}$ , ou seja,

$$\max_{\vec{f}} \text{Prob}[\vec{Y}=\vec{y}|\vec{f}] \tag{2.18}$$

No apêndice D, mostra-se que a solução é a obtida pela aplicação da pseudo-inversa de H em  $\vec{y}$ ,

$$\vec{f} = H^+ \cdot \vec{y} \tag{2.19}$$

Uma alternativa para a solução desta equação é dada pelo algoritmo iterativo EM (Expectation-Maximization) .

**2.4.1 Máxima Verossimilhança (ML) pelo algoritmo Expectation-Maximization (EM)**

O EM é um algoritmo iterativo para a estimativa da solução com máxima verossimilhança , e foi proposto inicialmente por Dempster et al. (10) . A sua aplicação em tomografia por emissão foi iniciada, independentemente, por Shepp e Vardi (50) e por Lange e Carson (32) .

Este algoritmo compreende duas fases: fase E onde se calcula o valor esperado condicional e a fase M quando se maximiza este valor esperado. O algoritmo reduz-se a (32):

$$f_j^{(k+1)} = f_j^{(k)} \cdot \frac{\sum_{i=1}^M H_{ij} \cdot y_i}{\sum_{i=1}^M H_{ij}} \quad j=1,N$$

onde

$$\hat{y}_i = \sum_{l=1}^N H_{il} \cdot f_l^{(k)} \quad i=1,M \quad : \text{reprojeção da imagem obtida na iteração atual}$$

As vantagens do ML via EM em reconstrução (emissão e transmissão) são:

- . trata a emissão e a transmissão de forma análoga;
- . incorpora um bom modelo físico e estatístico da contagem de fótons;
- . inclui implicitamente a informação da não-negatividade;
- . converge globalmente para a máxima verossimilhança;

- . cada iteração preserva a soma total das contagens;
- . a matriz de projeção H incorpora fatores físicos do equipamento e das medidas como o decaimento radioativo, eficiência dos detetores, variação da resolução espacial e fator de atenuação;

As desvantagens são:

- . como qualquer outro método iterativo, é necessário reconstruir toda a imagem;
- . é lento em comparação com técnicas como o CRP (pelo menos 1 ordem de grandeza). Para as mesmas condições anteriores (seção 2.2.1), tem-se

$$n^2(\text{estimativas}) + m.n(\text{reprojeções}) = n^2(m+m+2) + m.n.(n)$$

Para  $m=2n$ , tem-se, aproximadamente,  $6n^3$  multiplicações.

- . é assumido que o número de projeções seja maior ou igual ao de pixels, e que a matriz H tenha posto igual a sua menor dimensão (32), de modo a garantir a concavidade da função de verossimilhança;
- . as imagens reconstruídas usando a abordagem ML tendem a ser mais ruidosas à medida que a iteração prossegue (28);
- . nas regiões de alto contraste são produzidos "overshoots" devido ao ML (28);
- . a eficiência da correção a cada iteração não é uniforme sobre a imagem (54).
- . a solução de máxima verossimilhança não garante a melhor solução em termos de erro médio quadrático.

## 2.5 Conclusões

A reconstrução tomográfica e a restauração de imagens podem ser unificadas, dentro de uma visão mais abrangente, na recuperação de imagens (53).

Dentre os métodos de reconstrução discutidos neste capítulo, os métodos da transformada são os mais rápidos, mas pressupõem que as projeções são formadas a partir da transformada de Radon. Aproximações e correções permitem o emprego destes métodos em SPECT e PET.

A tabela abaixo mostra o número aproximado de operações de multiplicação necessários para reconstruir uma imagem  $n \times n$ , a partir de  $2n$  ângulos de projeção com  $n$  amostras por ângulo. Não foram consideradas as operações de adição, e nem as simplificações possíveis e dependentes da implementação, como no caso do CRP. Para o caso dos métodos iterativos, o tempo de processamento é bastante dependente da implementação. Não havendo disponibilidade de memória suficiente no computador, a matriz de coeficientes (matriz de projeção  $H$ ) deve ser calculada a partir de tabelas a cada iteração, implicando em um tempo adicional, representado por  $a$  e  $b$  na tabela 2.1. Nestas condições, na prática, o ART se torna mais lento do que o EM (ver tabela 5.1).

Tabela 2.1 Número aproximado de operações de multiplicação para reconstruir uma imagem  $n \times n$  a partir de  $2n$  ângulos de projeção com  $n$  amostras por ângulo

método	número de operações (multiplicação)
CRP	$2n^3$
FRP	$8n^2(\log_2 n + 2)$
DTF	$n^2(4\log_2 n + 5)$
ART	$(5n^3 + a)$ por iteração completa
EM	$(6n^3 + b)$ por iteração completa

### 3-MODELOS

- 3.1. Modelo para o ruído: Poisson
- 3.2. Formação das projeções
- 3.3. Modelo para as projeções sem ruído Poisson
- 3.4. Modelo estatístico do sistema
- 3.5. Modelo aditivo para ruído Poisson

A qualidade da reconstrução/restauração de imagens é bastante dependente dos modelos assumidos no processamento, os quais devem refletir, em maior ou menor grau, as informações "a priori" disponíveis sobre o processo, sejam elas de natureza estatística ou determinística. Estes modelos envolvem, essencialmente, um modelo para a imagem original, um modelo para o ruído presente nas medidas e um modelo de formação das observações (modelo do sistema). No caso da reconstrução, as observações medidas são as projeções. Discute-se, neste capítulo, o modelo de formação das projeções, donde se deriva a natureza do ruído presente nas mesmas, partindo-se da hipótese de que o modelo de ruído presente na imagem original é Poisson. Discute-se também um modelo para as projeções sem ruído, de modo a permitir estimadores com algumas informações a priori.

O presente trabalho limita-se à reconstrução de imagens bi-dimensionais (fatias), e que satisfaçam os modelos abaixo descritos, como o PET e o SPECT. Estruturas tri-dimensionais que podem ser decompostas em fatias, podem ser obtidas a partir do empilhamento da reconstrução das mesmas. Portanto, é suposto que a imagem original (imagem objeto) está no mesmo plano dos detetores, e que os detetores são colimados. Em SPECT, a colimação é obtida através de estruturas colimadoras (por construção) e em PET a colimação pode ser obtida pela imposição da quase simultaneidade na detecção de fótons, uma vez que cada pósitron dá origem a dois fótons simultâneos em direções praticamente opostas (60).

Assume-se que a imagem objeto está contida em uma região quadrada, subdividida igualmente em  $N=n^2$  pequenos quadrados (pixels) de mesmas dimensões com intensidade uniforme. As imagens serão tratadas no formato discretizado, onde cada amostra representa o valor médio do pixel.

Em tomografia por emissão (ECT), devido à natureza quântica dos fótons, o número efetivamente emitido por pixel, em determinado intervalo de tempo, segue o processo randômico de Poisson. As variáveis aleatórias (número de fótons emitidos) associadas a cada pixel podem ser consideradas independentes entre si a baixo nível de radiação (34). Desse modo, o processo de geração da imagem já apresenta uma flutuação caracterizada como ruído Poisson. Na seção 3.1 é descrito o modelo de ruído Poisson, independentes entre si, para os pixels da imagem a ser reconstruída. Na figura 3.1, o vetor  $\vec{f}_{N \times 1}$  representa, em notação vetorial, todos os pixels da imagem objeto sem ruído,

com  $N$  pixels e o vetor  $\vec{X}_{N \times 1}$  representa as variáveis aleatórias Poisson correspondentes. As projeções são formadas pela contribuição dos fótons emitidos por cada elemento (pixel/voxel) que pertença a uma determinada geometria em relação ao detector. Na seção 3.2 é mostrado que as projeções seguem também o processo Poisson, representado pelo vetor  $\vec{Y}$ . A matriz  $H$  representa o operador linear de projeção, que corresponde aos pesos na contribuição de cada pixel na formação das projeções. A projeção  $i$  está associada à linha  $i$  de  $H$ , que por sua vez contém os pesos relativos a cada pixel da imagem original. Devido à geometria, esta matriz contém muitos elementos zeros, ou seja, é esparsa e a sua dimensão é elevada. A título de exemplo, para uma imagem com  $100 \times 100$  pixels ( $N=10000$ ), com 100 ângulos de projeção com 100 amostras por ângulo de projeção, a dimensão de  $H$  é  $10000 \times 10000$ , ou seja, matriz de  $10^8$  elementos.

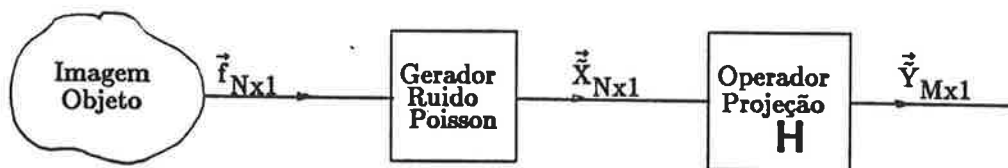


Figura 3.1. Modelo para a formação das projeções supondo ruído Poisson na imagem a ser reconstruída.

Na seção 3.4 é estabelecido o modelo do sistema utilizado para a estimativa das projeções e a reconstrução tomográfica. Na seção 3.3 discute-se um modelo de sinal para as projeções sem ruído e na seção 3.5 um modelo aditivo para ruído Poisson, de modo a justificar alguns procedimentos para estimativa ótima das projeções a partir das projeções medidas ( $\vec{y}_{M \times 1}$ ).

3.1 Modelo para o ruído: Poisson

Seja uma imagem objeto (figura 3.2) discretizada e representada pelo vetor  $\vec{f}^T = [ f_1 f_2 \dots f_N ]$  com N elementos (pixels) , onde o número de fótons efetivamente emitido em cada pixel ( $\tilde{X}_j$ ) é regido por um processo de Poisson, multivariado independente.

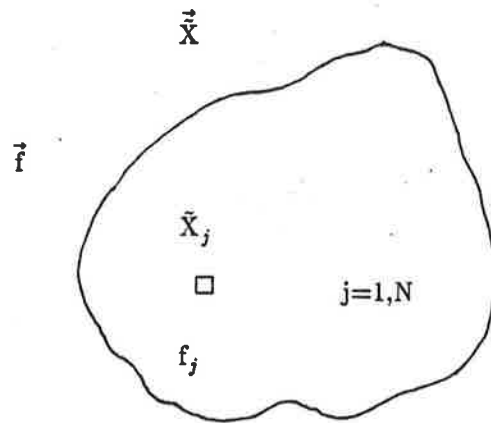


Figura 3.2. Imagem Objeto e o campo aleatório correspondente

Dessa forma, denotando por  $\vec{X}^T = [ \tilde{X}_1 \tilde{X}_2 \dots \tilde{X}_N ]$  o vetor de variáveis aleatórias independentes com taxa média de emissão  $\vec{f}$  , tem-se para o elemento  $j$  (34):

$$\text{Prob} [\tilde{X}_j = x_j | f_j] = \frac{e^{-f_j} (f_j)^{x_j}}{x_j!} \quad (\text{distribuição Poisson}) \quad (3.1)$$

E a probabilidade conjunta, uma vez que são independentes,

$$\text{Prob} [\vec{X} = \vec{x} | \vec{f}] = \prod_1^N \frac{e^{-f_j} (f_j)^{x_j}}{x_j!} \quad (3.2)$$

$$\text{onde } E[\tilde{X}_j | f_j] = f_j \quad j=1, N \quad (E[\ ] : \text{valor esperado}) \quad (3.3)$$

$$\text{var}[\tilde{X}_j | f_j] = f_j \quad j=1, N \quad (\text{var}[ \ ] : \text{variância}) \quad (3.4)$$

$$E[\tilde{X}_k \cdot \tilde{X}_l] = E[\tilde{X}_k] \cdot E[\tilde{X}_l] \quad k \neq l \quad (\text{independência}) \quad (3.5)$$

em notação vetorial,

$$E[\vec{\tilde{X}} | \vec{f}] = \vec{f} \quad (3.6)$$

$$\text{var}[\vec{\tilde{X}} | \vec{f}] = \vec{f} \quad (3.7)$$

Este modelo caracteriza adequadamente as imagens cintilográficas a baixas intensidades de radiação (35), e percebe-se a dependência do ruído com o sinal, uma vez que a variância do ruído é igual ao valor esperado do sinal (equação 3.4).

Pode-se notar que o ruído é mais crítico para baixas intensidades. De fato, considerando-se como relação sinal/ruído a expressão :

$$S/N = \frac{(E[\tilde{X}_j | f_j])^2}{\text{var}[\tilde{X}_j | f_j]} \quad (3.8)$$

resulta  $S/N = f_j$ , o que mostra que quanto maior o valor esperado do pixel, maior é a relação sinal/ruído.

### 3.2 Formação das projeções

Considerando-se, inicialmente, o caso de apenas um elemento emissor (figura 3.3), cujo número de emissões segue o processo de Poisson com taxa média  $f_j$  e um detetor ideal, pode-se provar (Apêndice A) que o número de fótons detetados também é uma variável aleatória Poisson com valor esperado  $h_{ij} \cdot f_j$ , onde  $h_{ij}$  corresponde à probabilidade de que um fóton efetivamente emitido em  $j$  seja detetado pelo sensor  $i$ , ou seja,

$$\tilde{X}'_j = h_{ij} \cdot \tilde{X}_j \quad \text{é variável aleatória (v.a.) Poisson} \quad (3.9)$$



Desse modo a contagem acumulada ao longo do feixe  $i$  (figura 3.3) corresponde à soma de variáveis aleatórias independentes Poisson. No entanto, esta somatória também tem distribuição de Poisson, com valor esperado igual à somatória dos valores esperados de cada componente. As v.a.  $Y_i$ ,  $i=1, M$  são também independentes entre si (60).

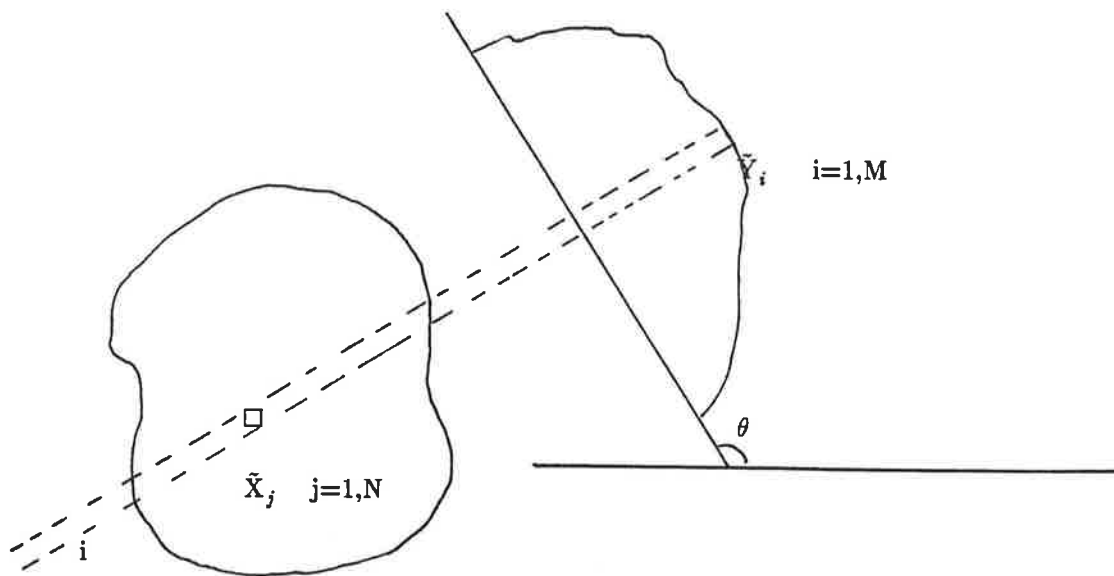


Figura 3.3. Formação das Projeções

Portanto,

$$\tilde{Y}_i = \sum_j^N \tilde{X}'_j = \sum_{j=1}^N h_{ij} \tilde{X}_j \quad i=1, M \text{ são v.a. independentes de Poisson} \quad (3.10)$$

Como  $E[\tilde{Y}_i | \vec{f}] = \sum_{j=1}^N h_{ij} E[\tilde{X}_j | \vec{f}]$  resulta (equação 3.3):

$$E[\tilde{Y}_i | \vec{f}] = \sum_{j=1}^N h_{ij} f_j \quad i=1, M \quad (3.11)$$

O fator  $h_{ij}$  pode englobar fenômenos como espalhamento, atenuação, decaimento radioativo, esquema da amostragem da projeção, eficiência dos detetores, resolução espacial, coincidências

acidentais e condições geométricas do instrumento (32) . Em notação vetorial, considerando-se  $M$  feixes de projeção ao redor da imagem objeto, tem-se (figura 3.3):

$$\vec{Y} = H_{M \times N} \vec{X} \tag{3.12}$$

$$H = \begin{bmatrix} h_{11} & h_{12} & \dots & h_{1N} \\ h_{21} & h_{22} & \dots & h_{2N} \\ \dots & \dots & \dots & \dots \\ h_{M1} & h_{M2} & \dots & h_{MN} \end{bmatrix} \tag{3.13}$$

(Matriz de Projeção)

$$E[\vec{Y} | \vec{f}] = H E[\vec{X} | \vec{f}] \tag{3.14}$$

Pelo fato de serem independentes , a probabilidade conjunta resulta em:

$$\text{Prob} [\vec{Y} | \vec{f}] = \prod_1^M \text{Prob}\{\tilde{Y}_i | \vec{f}\} \tag{3.15}$$

Da equação 3.6 tem-se

$$E[\vec{Y} | \vec{f}] = H \vec{f} \tag{3.16}$$

seja  $\vec{g} = H\vec{f}$

O diagrama da figura 3.4 ilustra a relação entre as projeções ruidosas e a imagem objeto.

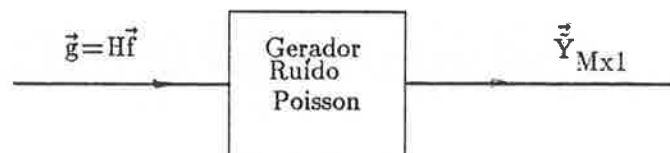


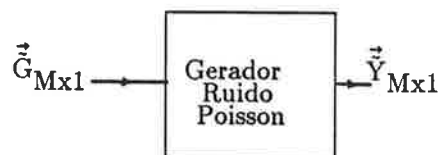
Figura 3.4. Formação das projeções ruidosas.

Conclui-se que o valor esperado das projeções, dada imagem  $\vec{f}$ , é uma combinação linear de  $\vec{f}$ . Portanto, dispondo-se da estimativa desses valores esperados, pode-se obter a imagem original pela

solução da equação 3.16, recaindo-se no clássico problema inverso (reconstrução) que, muitas vezes, é mal condicionado.

### 3.3 Modelo para as projecções sem ruído Poisson

Em tomografia, as medidas obtidas são as projecções. Com o intuito de possibilitar a incorporação de eventuais informações estatísticas "a priori" para a estimativa das projecções (estimadores Bayesianos), torna-se interessante estabelecer um modelo sobre a projecção sem ruído ( $\vec{G}$ ).



As projecções sem ruído como um todo, ou por ângulo de projecção, podem ser tratadas como campos aleatórios com média e variância não-estacionárias, além de correlacionados entre si, pelo menos para projecções vizinhas. Este modelo, apesar de preciso, é muito geral. Por outro lado, um modelo de sinal estacionário no sentido amplo é claramente inadequado, uma vez que em casos reais a média, em geral, é não-estacionária e a estrutura de correlação (segundo momento) não leva em consideração características locais. Trussel e Hunt (55) utilizaram um modelo com média não-estacionária e uma estrutura simplificada de covariância, assumindo que os sinais são não-correlacionados e as variâncias estacionárias (matriz diagonal com os elementos da diagonal iguais). Kuan (30) agregou mais informações neste modelo, considerando as variâncias não-estacionárias, resultando no modelo de sinal com média e variância não-estacionárias e não-correlacionados entre si (NMNV), onde a média não-estacionária representaria a estrutura mais grosseira do sinal, e as variâncias não-estacionárias caracterizariam, parcialmente, a estrutura local.

Em analogia com o modelo proposto por Kuan para a imagem, pode-se estabelecer um modelo Gaussiano multivariado, não-correlacionado, com média e variâncias não-estacionárias (NMNV) para as projeções sem ruído, ou seja,

$$\text{Prob}[\vec{G}=\vec{g}] = \frac{1}{\sqrt{(2\pi)^M |\text{Cov}(\vec{G})|}} e^{-0.5(\vec{g}-E[\vec{G}])^T \cdot \text{cov}^{-1}[\vec{G}] \cdot (\vec{g}-E[\vec{G}])} \quad (3.17)$$

onde

$$\text{cov}[\vec{G}] = E[(\vec{G}-E[\vec{G}])(\vec{G}-E[\vec{G}])^T] \quad (3.18)$$

$|\text{Cov}(\vec{G})|$  : determinante de  $\text{Cov}(\vec{G})$

$$\text{Cov}[\vec{G}] = \begin{bmatrix} \text{var}(\tilde{G}_1) & 0 & \dots & 0 \\ 0 & \text{var}(\tilde{G}_2) & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \text{var}(\tilde{G}_M) \end{bmatrix} \quad (3.19)$$

$$\text{onde } \text{var}[\tilde{G}_i] = E[(\tilde{G}_i - E[\tilde{G}_i])^2] \quad (3.20)$$

Assumindo-se a condição da equação 3.19 (não-correlacionamento), torna-se possível a derivação de estimadores pontuais como aqueles descritos no capítulo 4. Esta condição, em conjunto com a distribuição gaussiana (equação 3.17), implica, na realidade, em assumir que  $\tilde{G}_i$ ,  $i=1,M$  são independentes entre si.

### 3.4 Modelo estatístico do sistema

Considerando, para dada imagem objeto  $\vec{f}$ , que as projeções são v.a. independentes de Poisson (equação 3.10) com valores esperados

$$E[\vec{Y} | \vec{f}] = H \vec{f} \quad (\text{da eq.3.16})$$

pode-se construir o modelo estatístico equivalente para as projeções por (figura 3.5):

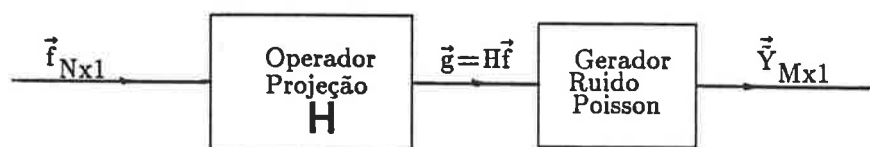


Figura 3.5. Modelo estatístico equivalente do sistema

Deste modo, uma alternativa para a reconstrução completa de  $\vec{f}$  consiste em estimar-se as projeções ( $\vec{g}$ ) a partir de  $\vec{y}$  e então aplicar-se algoritmos de reconstrução sobre  $\hat{\vec{g}}$ , dada a matriz de projeção  $H$ .

Este modelo para tomografia coincide com o modelo proposto por Lo (34) para a *restauração* de imagens com ruído Poisson, onde a matriz de projeção  $H$  corresponde à matriz de degradação (borramento).

### 3.5 Modelo aditivo para ruído Poisson

Cada v.a. independente de Poisson  $\tilde{Y}_i$  pode também ser vista como um processo com ruído aditivo, dependente do sinal porém não correlacionado com o sinal (24), como descrito na figura 3.6

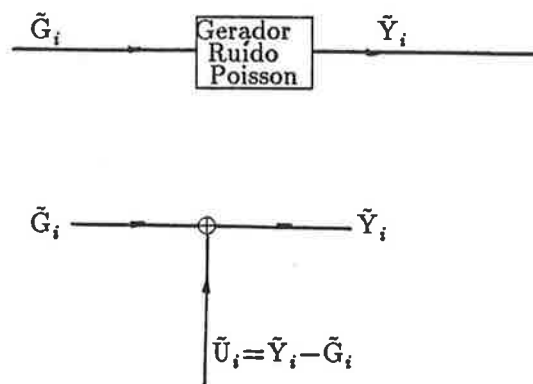


Figura 3.6. Transformação em ruído aditivo

onde  $\tilde{G}_i$  pode ser uma variável aleatória (no caso de modelos Bayesianos), cujo valor deseja-se estimar.

Os parâmetros da estatística das v.a.  $\tilde{U}_i$  e  $\tilde{Y}_i$  são (23):

$$E[\tilde{Y}_i | g_i] = g_i \quad (\text{v.a. Poisson}) \quad (3.21)$$

$$\text{var}[\tilde{Y}_i | g_i] = E_{\tilde{Y}_i}[(\tilde{Y}_i - E(\tilde{Y}_i | g_i))^2 | g_i] = g_i \quad (\text{v.a. Poisson}) \quad (3.22)$$

Da equação 3.21 e 3.22 tem-se as v.a.s (Apêndice B, equações A.20 e A.26)

$$E_{\tilde{Y}_i}[\tilde{Y}_i | \tilde{G}_i] = \tilde{G}_i \quad (3.23)$$

$$E_{\tilde{Y}_i}[(\tilde{Y}_i - E(\tilde{Y}_i | \tilde{G}_i))^2 | \tilde{G}_i] = \tilde{G}_i \quad (3.24)$$

$$\begin{aligned} E_{\tilde{U}_i}[\tilde{U}_i | \tilde{G}_i] &= E[(\tilde{Y}_i - \tilde{G}_i) | \tilde{G}_i] \\ &= E(\tilde{Y}_i | \tilde{G}_i) - \tilde{G}_i = \tilde{G}_i - \tilde{G}_i \end{aligned}$$

$$=0 \quad (3.25)$$

$$\begin{aligned} E[\tilde{G}_i \tilde{U}_i] &= E_{G_i} E_{U_i} [\tilde{G}_i \tilde{U}_i | \tilde{G}_i] && \text{(da equação A.23)} \\ &= E_{G_i} [\tilde{G}_i \cdot E_{U_i} (\tilde{U}_i | \tilde{G}_i)] && \text{(da equação 3.25)} \end{aligned}$$

$$=0 \quad (3.26)$$

$$\begin{aligned} E[\tilde{U}_i] &= E_{G_i} [E_{Y_i} [(\tilde{Y}_i - \tilde{G}_i) | \tilde{G}_i]] && \text{(da equação A. 23 )} \\ &= E_{G_i} [E_{Y_i} [\tilde{Y}_i | \tilde{G}_i] - \tilde{G}_i] = E[\tilde{G}_i - \tilde{G}_i] \end{aligned}$$

$$=0 \quad (3.27)$$

$$E[\tilde{Y}_i] = E_{G_i} [E_{Y_i} [\tilde{Y}_i | \tilde{G}_i]] = E[\tilde{G}_i] \quad (3.28)$$

$$\begin{aligned} \text{var}[\tilde{U}_i | \tilde{G}_i] &= E[\tilde{U}_i^2 | \tilde{G}_i] \\ &= E[(\tilde{Y}_i - \tilde{G}_i)^2 | \tilde{G}_i] \\ &= E[(\tilde{Y}_i - E(\tilde{Y}_i | \tilde{G}_i) + E(\tilde{Y}_i | \tilde{G}_i) - \tilde{G}_i)^2 | \tilde{G}_i] \\ &= E[(\tilde{Y}_i - E(\tilde{Y}_i | \tilde{G}_i))^2 | \tilde{G}_i] \\ &= \text{var}(\tilde{Y}_i | \tilde{G}_i) \\ &= \tilde{G}_i \end{aligned} \quad (3.29)$$

$$\begin{aligned} \text{var}[\tilde{U}_i] &= E[\tilde{U}_i^2] = E[(\tilde{Y}_i - \tilde{G}_i)^2] = E[(\tilde{Y}_i - E(\tilde{Y}_i | \tilde{G}_i) + E(\tilde{Y}_i | \tilde{G}_i) - \tilde{G}_i)^2] \\ &= E[(\tilde{Y}_i - E(\tilde{Y}_i | \tilde{G}_i))^2] \\ &= E_{G_i} E_{Y_i} [(\tilde{Y}_i - E(\tilde{Y}_i | \tilde{G}_i))^2 | \tilde{G}_i] \\ &= E_{G_i} (\tilde{G}_i) && \text{(da equação 3.29 )} \end{aligned} \quad (3.30)$$

$$\begin{aligned} \text{var}[\tilde{Y}_i] &= E[(\tilde{Y}_i - E(\tilde{Y}_i))^2] = E[(\tilde{G}_i + \tilde{U}_i - E(\tilde{G}_i))^2] && \text{(da equação 3.28)} \\ &= E[(\tilde{G}_i - E(\tilde{G}_i))^2 + E(\tilde{U}_i^2) - 2E[(\tilde{G}_i - E(\tilde{G}_i)) \cdot \tilde{U}_i]] \\ &= \text{var}(\tilde{G}_i) + \text{var}(\tilde{U}_i) - 2E[\tilde{G}_i \tilde{U}_i] + 2E(\tilde{G}_i) \cdot E(\tilde{U}_i) \\ &= \text{var}(\tilde{G}_i) + \text{var}(\tilde{U}_i) && \text{(das eq. 3.26 e 3.27 )} \\ &= \text{var}(\tilde{G}_i) + E[\tilde{G}_i] \end{aligned} \quad (3.31)$$

Para  $i \neq j$   $\tilde{Y}_i$  e  $\tilde{Y}_j$  são independentes, assim como  $\tilde{Y}_i$  e  $\tilde{G}_j$ . Então,

$$\begin{aligned} E[\tilde{U}_i \tilde{U}_j] &= E[(\tilde{Y}_i - \tilde{G}_i)(\tilde{Y}_j - \tilde{G}_j)] = E[\tilde{Y}_i \tilde{Y}_j] - E[\tilde{Y}_i \tilde{G}_j] - E[\tilde{Y}_j \tilde{G}_i] + E[\tilde{G}_i \tilde{G}_j] \\ &= E[\tilde{Y}_i] \cdot E[\tilde{Y}_j] - E[\tilde{Y}_i] E[\tilde{G}_j] - E[\tilde{G}_i] E[\tilde{Y}_j] + E[\tilde{G}_i \tilde{G}_j] \\ &= E[\tilde{G}_i \tilde{G}_j] - E[\tilde{G}_i] \cdot E[\tilde{G}_j] && \text{pois } E[\tilde{Y}_j] = E[\tilde{G}_j] \end{aligned} \quad (3.32)$$

ou seja,

$$\begin{aligned}
\text{cov}(\tilde{U}_i, \tilde{U}_j) &= \text{cov}(\tilde{G}_i, \tilde{G}_j) \\
\text{cov}(\tilde{U}_i, \tilde{G}_i) &= E[(\tilde{U}_i - E(\tilde{U}_i))(\tilde{G}_i - E(\tilde{G}_i))] \\
&= E(\tilde{U}_i \tilde{G}_i) - E(\tilde{U}_i) \cdot E(\tilde{G}_i) = E(\tilde{U}_i \tilde{G}_i) = 0 \quad (\text{da equação 3.26}) \quad (3.33)
\end{aligned}$$

Conclui-se que são válidas as seguintes afirmações:

- O ruído aditivo ( $\tilde{U}_i$ ) é não-correlacionado com o sinal ( $\tilde{G}_i$ ) como pode ser constatado pela equação 3.33 e tem valor esperado igual a zero;
- A variância do ruído aditivo ( $\tilde{U}_i$ ) é o valor esperado do sinal ( $\tilde{G}_i$ ) (equação 3.30);
- A variância do sinal observado ( $\tilde{Y}_i$ ) é a soma das variâncias do sinal ( $\tilde{G}_i$ ) mais a variância do ruído aditivo  $\tilde{U}_i$  (Equação 3.31);
- O valor esperado do sinal observado ( $\tilde{Y}_i$ ) é o valor esperado do sinal ( $\tilde{G}_i$ ) (equação 3.28);

Se as variáveis aleatórias  $\tilde{G}_i$  forem não-correlacionados ( $E[\tilde{G}_i \cdot \tilde{G}_j] = E[\tilde{G}_i] \cdot E[\tilde{G}_j]$ ), resulta

$$E[\tilde{U}_i \cdot \tilde{U}_j] = 0 \quad i \neq j \quad (3.34)$$

i.é, o ruído é branco. Nestas condições, o ruído aditivo  $\vec{\tilde{U}}$  caracteriza-se por ter a matriz de covariância diagonal:

$$\text{Cov}[\vec{\tilde{U}}] = \begin{bmatrix} E(\tilde{G}_1) & 0 & \dots & 0 \\ 0 & E(\tilde{G}_2) & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & E(\tilde{G}_M) \end{bmatrix} \quad (3.35)$$

$$E[\vec{\tilde{U}}] = \vec{0} \quad (3.36)$$

$$\text{var}[\vec{\tilde{U}}] = E[\vec{\tilde{G}}] \quad (3.37)$$



## 4-METODOLOGIA PROPOSTA

### 4.1. Descrição do método proposto

### 4.2. Estimadores para processo Poisson

#### 4.2.1. Maximum a Posteriori -MAP (Bayesiano)

#### 4.2.2. Estimador local linear com menor erro médio quadrático (Wiener)

### 4.3. Estimativa envolvendo transformação não linear de Anscombe

#### 4.3.1. Estimador Bayesiano MAP

#### 4.3.2. Estimador de Wiener local

#### 4.3.3. Estimador heurístico

#### 4.3.4. Minimização de função custo (rugosidade)

### 4.4. Reconstrução DTF: refinamentos

#### 4.4.1. Filtragem

#### 4.4.2. Interpolação no domínio da frequência

#### 4.4.3. Implementação do DTF

### 4.5. Conclusões

O maior fator limitante em tomografia por emissão em medicina é o ruído estatístico inerente ao processo (50). Desse modo, um modelo para a reconstrução deve considerar os aspectos estatísticos envolvidos.

Dentro do modelo estatístico do sistema descrito anteriormente, mostrou-se que o problema da reconstrução tomográfica de imagens, sujeitas a processo Poisson, pode ser resolvido estimando-se as projeções antes da reconstrução propriamente dita. Esta abordagem apresenta as seguintes vantagens:

- a) Utilização de vários tipos de estimadores, inclusive estimadores otimizados (Maximum a Posteriori, Wiener,...);
- b) Incorporação de informações a priori sobre as projeções (não negatividade, suavidade, ...);
- c) Uso de estatística local nos estimadores;
- d) Agilização na reconstrução, uma vez que a redução do ruído e a correção da atenuação permitem o emprego de técnicas de reconstrução rápidas, não-iterativas, como a Filtragem-Retroprojeção (FRP) e a aplicação direta da transformada de Fourier(DTF).

Métodos da retroprojeção filtrada apenas filtram as projeções com filtros passa-baixa clássicos como o Hann, Hamming e Parzen (43), assumindo modelo de sinal (projeção) estacionário. Outra abordagem, discutida anteriormente (seção 2.4.1), que maximiza a verossimilhança pelo algoritmo EM, incorpora o modelo de formação das projeções e a natureza estatística das projeções (Poisson). No entanto, este método é lento devido à iteratividade, além do fato de que a máxima verossimilhança não garante o menor erro médio quadrático.

O presente trabalho propõe a estimativa das projeções antes da reconstrução, considerando um modelo de ruído realístico, um modelo de formação das projeções, e um modelo estatístico do sistema. Os estimadores descritos a seguir, na sua maioria, utilizam informações locais (estatística local) na estimativa, podendo, portanto, serem empregados em sinais não-estacionários.

A seção 4.1 traz a descrição do método proposto, que envolve, essencialmente, a estimativa das

projeções considerando os modelos descritos no capítulo anterior. São utilizados estimadores diretamente sobre as projeções com ruído Poisson (seção 4.2) e estimadores após a transformação de Anscombe. A seção 4.2.1 traz o estimador Bayesiano Maximum a Posteriori (MAP) para as projeções e a seção 4.2.2 o estimador com menor erro médio quadrático (filtro de Wiener).

Na seção 4.3, o ruído Poisson é transformado em ruído aproximadamente gaussiano e independente do sinal. Nestas condições, são apresentados 4 estimadores : MAP, Wiener, heurístico e por minimização da função custo.

Na seção 4.4, são apresentados alguns refinamentos sobre o método da aplicação Direta da Transformada de Fourier, de modo a se conseguir reconstruções de melhor qualidade e mais rapidamente.

#### 4.1 Descrição do Método proposto

De acordo com o modelo estatístico da formação das projeções, descrito na seção 3.4, pode-se formular o problema da reconstrução em ECT conforme o diagrama ilustrado na figura 4.1, onde  $\vec{Y}$  é um vetor ( $M \times 1$ ) de variáveis aleatórias independentes, com distribuição de Poisson, e média  $E[\vec{Y}] = \vec{g} = H\vec{f}$ .

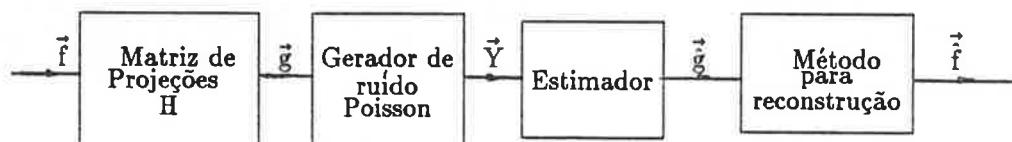


Figura 4.1. Diagrama em blocos do método proposto

A imagem objeto é representada pelo  $\vec{f}_{Nx1}$  (N pixels) a qual é projetada pelo operador  $H_{MxN}$  "formando" as projeções sem ruído  $\vec{g}_{Mx1}$ . Agregando-se o ruído Poisson, o qual é dependente do sinal, têm-se as projeções medidas (ruidosas)  $\vec{Y}_{Mx1}$ . A reconstrução da imagem  $\vec{f}_{Nx1}$  ocorre em dois passos. Primeiro estimando-se as projeções sem ruído ( $\vec{g}_{Mx1}$ ) e então reconstruindo-se ( $\vec{f}$ ) a partir destas projeções estimadas.

O bloco "estimador" fará a estimativa (uni-dimensional) do  $\vec{g}$  a partir das projeções medidas  $\vec{y}$ . Serão apresentados, na próxima seção, vários estimadores com características ótimas ou sub-ótimas. Cada estimador envolve, implícita ou explicitamente, três fases: a) um modelo de formação do sinal; b) um critério a ser otimizado; c) um algoritmo computacional.

No bloco de reconstrução, podem ser empregados vários métodos de reconstrução (capítulo 2). Objetivando-se a rapidez, e se as projeções corrigidas puderem ser consideradas provenientes da transformada de Radon os métodos da transformada (FRP ou DTF) são os mais adequados.

O desempenho de cada estimador será avaliado segundo o esquema mostrado na figura 4.2, onde a projeção sem ruído  $\vec{g}$  foi simulada, e a projeção medida  $\vec{y}$  corresponde à saída de um processo de Poisson com média  $\vec{g}$ .

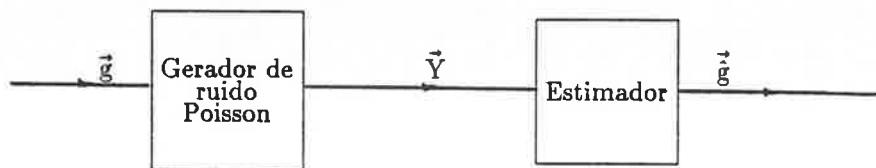


Figura 4.2. Esquema para avaliar o estimador .

A figura 4.3a mostra o sinal simulado ( $\vec{g}$ ), e a figura 4.3b o sinal com ruído Poisson ( $\vec{y}$ ). O número de amostras utilizado (M) foi de 512 pontos. O valor máximo do sinal sem ruído foi atribuído a 41, de modo a simular sinais com baixa relação sinal/ruído, uma vez que em processos de Poisson a relação sinal/ruído é dependente do valor esperado (equação 3.8). Consistente com o modelo de ruído Poisson, pode-se notar que a variância é dependente do sinal original. A figura 4.4 mostra o ruído aditivo ( $\vec{y}-\vec{g}$ ) e o histograma correspondente, com média e variância iguais a 0.086 e 18.91, respectivamente. Se o processo fosse estacionário, os valores teóricos previstos para as mesmas seriam (eq. 3.27 e 3.30), respectivamente, 0 e 20.705.

Como medida de desempenho do estimador será usada a Raiz quadrada do Erro Médio Quadrático Normalizado (REMQRN) definida por:

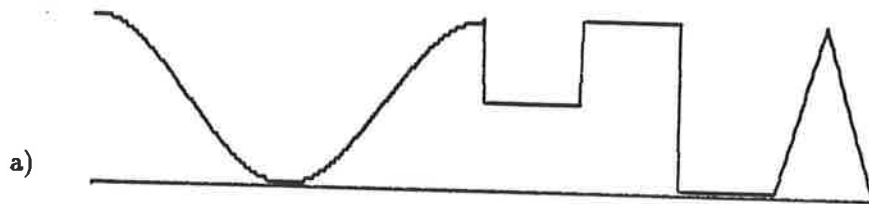
$$\text{REMQRN} = \sqrt{\frac{\sum_j^M (g_j - \hat{g}_j)^2}{\sum_j^M g_j^2}} \quad (4.1)$$

a qual fornecerá uma indicação do erro de estimativa. Considerando-se  $\vec{g} = \vec{y}$  para o caso da figura 4.3, o REMQRN obtido para a amostra da figura 4.3b, resulta em:

$$\text{REMQRN}_{\text{sem estimador}} = 0.1713$$

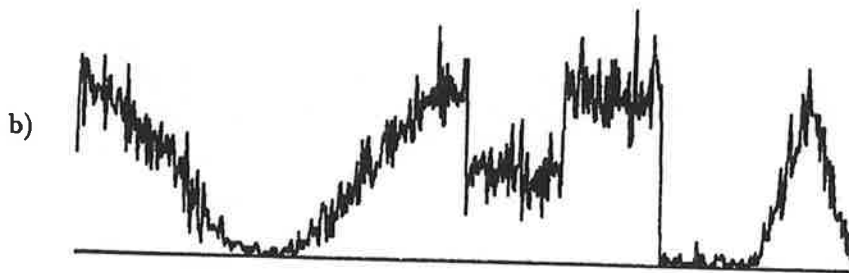
Pode-se medir a melhoria obtida pelo emprego do estimador por:

$$\text{melhoria do estimador} = 10 \log \frac{\sum (y_i - \hat{g}_i)^2}{\sum (\hat{g}_i - g_i)^2} = 20 \log \frac{\text{REMQRN}_{\text{sem estimador}}}{\text{REMQRN}_{\text{estimador}}} \quad (4.2)$$



Arq.: poissona.dat

(min= 1.000,med= 20.705,max= 41.0)



Arq.: poisson.dat

(min= 0.000,med= 20.791,max= 61.0)

Figura 4.3 Simulação com 512 pontos discretos . O traçado a) mostra um sinal sem ruído ( $\vec{g}$ ) com valores mínimo, médio e máximo (min,med,max=1.00, 20.705, 41.0) e em b) uma saída com ruído Poisson ( $\vec{y}$ ) , para o qual os valores foram (min,med,max= 0.00, 20.791, 61.0)



a)

Arq.: poissand

(min= -16.000,med= 0.086,max= 20.0)

### Histograma

b)

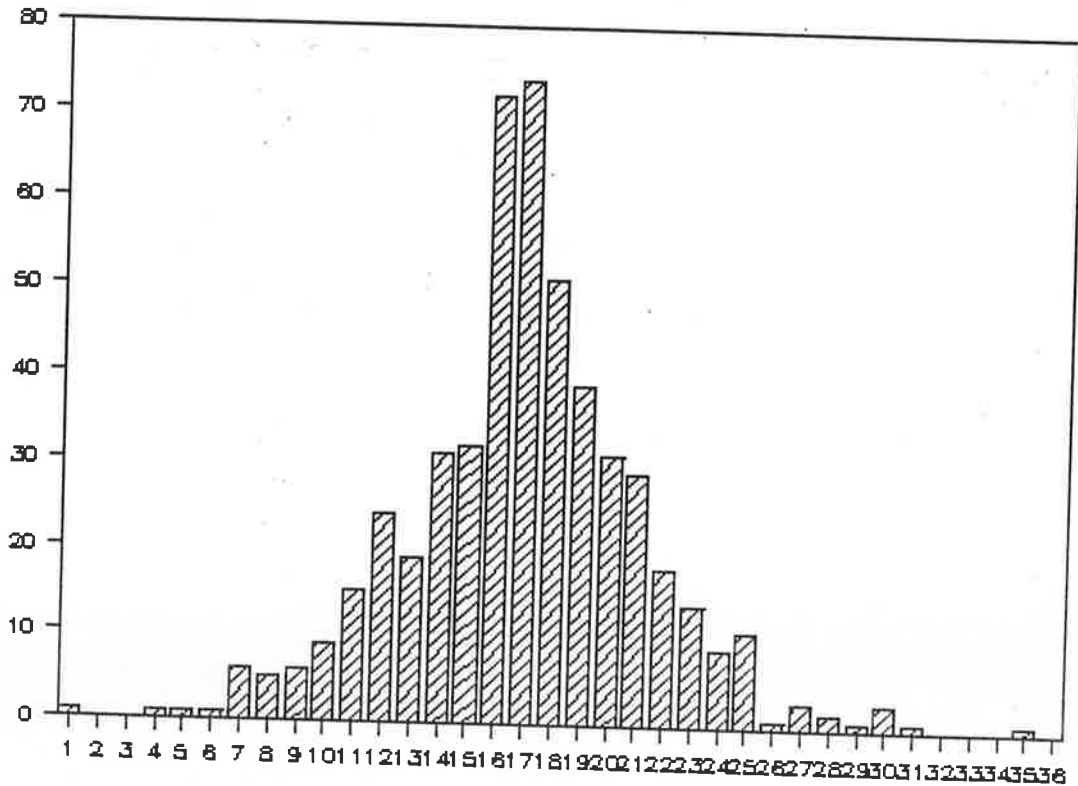


Figura 4.4 Simulação com 512 pontos discretos . O traçado a) mostra o ruído ( $\vec{y}-\vec{g}$ ) com valores mínimo, médio e máximo (min,med,max= -16.00, 0.086, 20.0) e variância=18.91; e em b) o histograma correspondente , em 36 classes

#### 4.2 Estimadores para processo Poisson

Dado que o ruído é Poisson e as v.a.  $Y_i$ ,  $i=1, M$  são independentes, pode-se aplicar alguns métodos ótimos para a estimativa das projeções  $g_i$  (figura 4.2). Serão considerados os métodos Bayesianos Maximum a Posteriori (MAP) e o de menor erro médio quadrático (filtro local de Wiener).

##### 4.2.1 Maximum a Posteriori - MAP (Bayesiano)

Este método é vantajoso quando se conhece a estatística do sinal  $\vec{G}$ . O critério a ser maximizado é a probabilidade de  $\vec{G}$  dadas as observações  $\vec{y}$ . Do teorema de Bayes tem-se:

$$\text{Prob}[\vec{G} | \vec{y}] = \frac{\text{Prob}[\vec{Y} | \vec{g}] \cdot \text{Prob}[\vec{G}]}{\text{Prob}[\vec{Y}]} \quad (4.3)$$

onde

$$\text{Prob}[\vec{Y} | \vec{g}] = \prod_1^M \text{Prob}[Y_i | g_i] = \prod_{i=1}^M \frac{e^{-g_i} \cdot (g_i)^{y_i}}{y_i!} \quad (\text{Poisson}) \quad (4.4)$$

Supondo-se que o sinal  $\vec{G}$  tem uma distribuição Gaussiana multivariada com média e covariância não-estacionária (NMNV), conforme descrito na seção 3.3, pode-se escrever:

$$\text{Prob}[\vec{G} = \vec{g}] = \frac{1}{\sqrt{(2\pi)^M |\text{Cov}(\vec{G})|}} e^{-0.5(\vec{g} - E[\vec{G}])^T \cdot \text{cov}^{-1}[\vec{G}] \cdot (\vec{g} - E[\vec{G}])} \quad (4.5)$$

Utilizando-se uma função monotônica para simplificar os cálculos na maximização, tem-se:

$$\begin{aligned} L(\vec{g}) &= \ln \text{Prob}[\vec{G} | \vec{y}] \\ &= \sum_{i=1}^M (-g_i + y_i \ln(g_i) - \ln(y_i!)) - 0.5(\vec{g} - E[\vec{G}])^T \text{Cov}^{-1}(\vec{G}) \cdot (\vec{g} - E[\vec{G}]) - K \end{aligned}$$

$$\text{onde } K = 0.5 \ln((2\pi)^M |\text{Cov}[\vec{G}]|) + \ln \text{Prob}[\vec{Y}]$$

Eliminando-se as constantes e as parcelas que não influirão na maximização em  $\vec{g}$ , e derivando-se em relação a  $g_k$  resulta:



$$\frac{\partial L(\vec{g})}{\partial g_k} = (y_k \cdot \frac{1}{g_k} - 1) - \sum_{i=1}^M C_{ki} (g_i - E[G_i]) \quad k=1, M \quad (4.6)$$

onde  $C_{ki} = [\text{Cov}^{-1}(\vec{G})]_{ki}$  (4.7)

Considerando ainda que o sinal  $\vec{G}$  é não correlacionado entre si (NMNV) , i.é,

$$\text{Cov}[\vec{G}] = \begin{bmatrix} \text{var}(G_1) & 0 & \dots & 0 \\ 0 & \text{var}(G_2) & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \text{var}(G_M) \end{bmatrix} \quad (4.8)$$

Substituindo na equação 4.6 e igualando a zero , resulta (35) no estimador puntual:

$$\left(\frac{y_k}{g_k} - 1\right) - \frac{g_k - E[G_k]}{\text{var}(G_k)} = 0 \quad k=1, M \quad (4.9)$$

Resolvendo esta equação do segundo grau, tem-se o estimador local (filtro adaptativo) não-linear:

$$g_k = \frac{E[G_k] - \text{var}(G_k) + \sqrt{(E[G_k] - \text{var}(G_k))^2 + 4 \text{var}(G_k) \cdot y_k}}{2} \quad (4.10)$$

A segunda solução desta equação é abandonada, pois é menor ou igual a zero.

Não dispondo das médias e variâncias para cada v.a. , as mesmas terão de ser estimadas a partir dos sinais ruidosos ( $\vec{y}$ ), supondo que são localmente ergódicas. Da equação 3.28, tem-se:

$$E(Y_k) = E[G_k] \cong \bar{y}_k = \frac{1}{2J+1} \cdot \sum_{j=-J}^{j=J} y_{k+j} \quad \text{para janela de } 2J+1 \text{ elementos} \quad (4.11)$$

$$\text{var}(Y_k) \cong s_k^2 = \frac{1}{2J} \cdot \sum_{j=-J}^{j=J} (y_{k+j} - \bar{y}_{k+j})^2 \quad k=1, M \quad (4.12)$$

$$\text{var}(G_k) = \text{var}(Y_k) - E(G_k) \quad (\text{da equação 3.31}) \quad (4.13)$$

$$\text{var}(G_k) \cong s_k^2 - \bar{y}_k$$



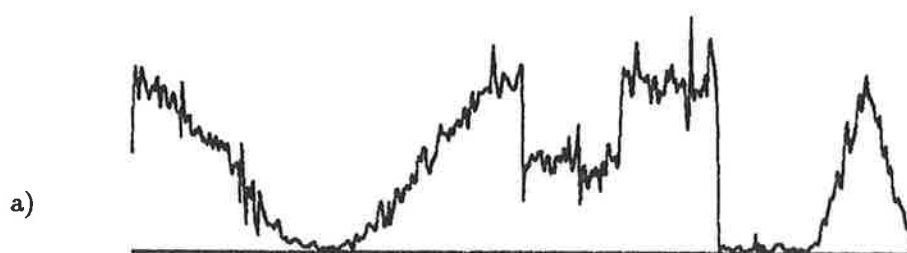
A estimativa destes parâmetros é crítica para a qualidade da restauração, pois este método assume que a estatística do sinal é conhecida. Na equação acima, caso a  $\text{var}(G_k)$  resulte em valor negativo, a mesma é colocada a zero.

Logo,

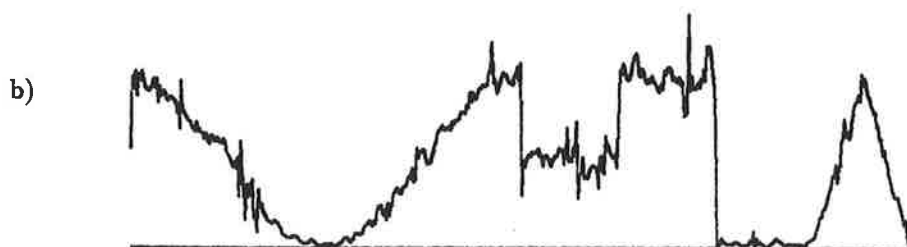
$$v_k^2 = \max(0, s_k^2 - \bar{y}_k)$$

$$g_k = \frac{\bar{y}_k - v_k^2 + \sqrt{(\bar{y}_k - v_k^2)^2 + 4 v_k^2 \cdot y_k}}{2}$$

A figura 4.5 e a tabela 4.2 mostram os resultados deste método aplicado aos sinais ruidosos da figura 4.3, utilizando janelas de 3 e 5 elementos na estimativa das médias e variâncias locais. A melhoria obtida foi de 3.67 e 4.75 dB respectivamente.



Arq.: po\_map\_3.dat  
(min= 0.000,med= 20.756,max= 56.4)



Arq.: po\_map\_5.dat  
(min= -0.000,med= 20.737,max= 55.7)

Figura 4.5. Resultado da aplicação do estimador Bayesiano MAP. a) com janela de 3 elementos para estimativa das médias e variâncias locais; b) com janela de 5 elementos. Melhoria de 3.67 e 4.75 dB respectivamente.

4.2.2 Estimador local e linear com menor erro médio quadrático (Wiener)

Na seção 3.5 foi discutido o modelo aditivo para processo de Poisson

$$\vec{Y} = \vec{G} + \vec{U} \quad \text{onde } \vec{U} \text{ é dependente de } \vec{G} \quad (4.14)$$

seja o estimador local e linear  $\vec{G} = E[\vec{G}] + B(\vec{Y} - E[\vec{Y}])$  onde matriz B será determinada de modo a minimizar o erro médio quadrático da estimativa:

$$\begin{aligned} \text{Erro}(B) &= E [|| \vec{G} - \vec{G} ||^2] \\ &= E[(\vec{G} - \vec{G})^T (\vec{G} - \vec{G})] \\ &= E[(E[\vec{G}] - \vec{G} + B(\vec{Y} - E[\vec{Y}]))^T \cdot (E[\vec{G}] - \vec{G} + B(\vec{Y} - E[\vec{Y}]))] \\ &= E[(E[\vec{G}] - \vec{G})^T \cdot (E[\vec{G}] - \vec{G}) + 2(\vec{Y} - E[\vec{Y}])^T B^T (E[\vec{G}] - \vec{G}) + \\ &\quad + (\vec{Y} - E[\vec{Y}])^T B^T B (\vec{Y} - E[\vec{Y}])] \end{aligned}$$

derivando em relação a cada componente de B e igualando a zero tem-se:

$$E\{ 2(E[\vec{G}] - \vec{G})(\vec{Y} - E[\vec{Y}])^T + 2B (\vec{Y} - E[\vec{Y}])(\vec{Y} - E[\vec{Y}])^T \} = [0]$$

substituindo a equação 4.14 e considerando que  $E[\vec{U}] = \vec{0}$ ,  $E[\vec{U}\vec{G}^T] = [0]$  (eq.3.26),

$$B (\text{cov}(\vec{G}) + \text{cov}(\vec{U})) = \text{cov}(\vec{G})$$

Donde o estimador (filtro de Wiener) :

$$\vec{G} = E(\vec{G}) + \text{cov}(\vec{G}) (\text{cov}(\vec{G}) + \text{cov}(\vec{U}))^{-1} (\vec{Y} - E(\vec{Y})) \quad (4.15)$$

Adotando o modelo descrito para as projeções sem o ruído Poisson (seção 3.5), como processo Gaussiano multivariado com média e variância não estacionárias, e não-correlacionados entre si (NMNV), tem-se que  $\vec{U}$  é também não correlacionado entre si (Equação 3.35),

$$\text{Cov}[\vec{U}] = \begin{bmatrix} E(G_1) & 0 & \dots & 0 \\ 0 & E(G_2) & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & E(G_M) \end{bmatrix} \quad (4.16)$$

Nestas condições, tem-se o estimador local, diretamente da equação 4.15, pois  $\text{cov}(\vec{U})$  e  $\text{cov}(\vec{G})$  são matrizes diagonais:

$$\hat{g}_i = E[G_i] + \frac{\text{var}(G_i)}{\text{var}(G_i) + \text{var}[U_i]} \cdot (y_i - E[Y_i]) \quad i=1, M \quad (4.17)$$

onde são válidas as equações abaixo (seção 3.5):

$$E[G_i] = E[Y_i]$$

$$\text{var}[G_i] = \text{var}[Y_i] - \text{var}[U_i]$$

$$\text{var}[U_i] = E[G_i]$$

Donde (30)

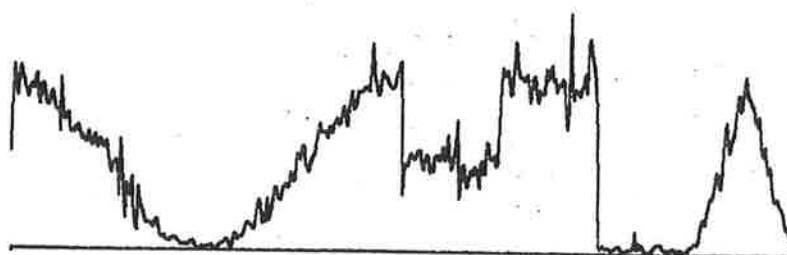
$$\hat{g}_i = E[G_i] + \frac{\text{var}(G_i)}{\text{var}(G_i) + E[G_i]} \cdot (y_i - E[G_i]) \quad (4.18)$$

Caso não se disponha das estatísticas das projeções não ruidosas, as mesmas ( $E[G_i]$  e  $\text{var}(G_i)$ ) terão de ser estimadas das projeções medidas (ruidosas), supondo que são localmente ergódicas, conforme descrito anteriormente (equações 4.11 e 4.12):

Logo

$$\hat{g}_i = \bar{y}_i + \frac{s_i^2 - \bar{y}_i}{s_i^2} \cdot (y_i - \bar{y}_i) \quad i=1, M \quad (4.19)$$

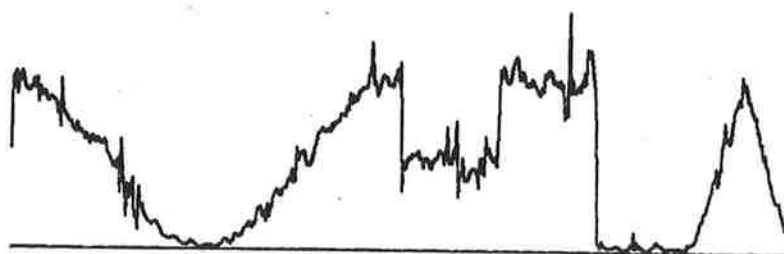
A figura 4.6 e a tabela 4.2 mostram os resultados deste estimador sobre o sinal da figura 4.3 onde as janelas consideradas para efeito de estimativa das médias e variâncias locais foram de 3 e 5 elementos. A melhoria foi de 3.68 e 4.74 dB, respectivamente. Nota-se que estes resultados são, qualitativa e quantitativamente, muito próximos aos obtidos para o estimador MAP (seção 4.2.1), pois ambos são estimadores Bayesianos ótimos, assumindo o mesmo modelo de sinal original, e porque o máximo da função  $\text{Prob}[\vec{G}|\vec{y}]$  (MAP) é próximo ao  $E[\vec{G}|\vec{y}]$  (estimador linear de menor erro médio quadrático) (29).



a)

Arq.: po\_wie\_3.dat

(min= 0.000,med= 20.805,max= 57.3)



b)

Arq.: po\_wie\_5.dat

(min= 0.000,med= 20.803,max= 57.0)

Figura 4.6. Resultado da estimativa pelo estimador do menor erro médio quadrático, aplicado diretamente sobre as projeções . a) com janela de 3 elementos para estimativa das médias e variâncias locais; b) com janela de 5 elementos. Melhoria de 3.68 e 4.74 dB respectivamente.

#### 4.3 Estimativa envolvendo transformação não-linear de Anscombe

O ruído Poisson é dependente do sinal como descrito na seção 3.1. No entanto, através de uma transformação não linear, o ruído pode tornar-se aproximadamente independente do sinal, aditivo (22) e mais próximo da distribuição gaussiana (3).

De fato, sendo  $Y$  a v.a com distribuição de Poisson com valor esperado  $g=E(Y)$ , tem-se pelo modelo aditivo (seção 3.5),

$$Y=g+U \quad \text{onde } E(U)=0$$

$$\text{var}(U)=g \quad (g \text{ não é v.a.})$$

Considerando a transformação da raiz quadrada,

$$Z=\sqrt{Y}=\sqrt{g+U}$$

Expandindo em série de Taylor, tem-se

$$Z=\sqrt{g}\left(1+\frac{U}{2g}-\frac{1}{8}\left(\frac{U}{g}\right)^2+\frac{1}{16}\left(\frac{U}{g}\right)^3-\dots\right)$$

donde (22),

$$E(Z)=\sqrt{g}\left(1+\frac{E(U)}{2g}-\frac{1}{8}\frac{E(U^2)}{g^2}+\frac{1}{16}\frac{E(U^3)}{g^3}-\dots\right)=\sqrt{g}\left(1-\frac{1}{8g}+\frac{1}{16g^2}-\dots\right)$$

$$\text{var}(Z)=E(Z^2)-(E[Z])^2=E(Y)-(E[Z])^2$$

$$=g-g\left(1+\frac{1}{64g^2}-\frac{1}{4g}+\dots\right)$$

$$=\frac{1}{4}+\frac{3}{32g}+\dots$$

Portanto  $\text{var}(Z)\cong 1/4$

O sinal  $Z$  pode ser escrito (22)

$$Z=\sqrt{g}+V \quad \text{onde } E(V)\cong 0 \quad \text{e } \text{var}(V)\cong 1/4$$

Nota-se que a relação sinal/ruído após a transformação,  $(S/N)_{\text{Transf.}}=(\sqrt{g})^2/\text{var}(V)=4g$ , é melhor do que sem a transformação ( $S/N=g$ ), da ordem de 4 vezes, devido à retirada da dependência do ruído em relação ao sinal.

Anscombe propôs uma transformação da raiz quadrada da v.a acrescida de uma constante no intuito de tornar a polarização (bias) da estimativa de  $g$  aproximadamente constante.



Deste modo, sendo  $Y_i$  a variável aleatória com distribuição Poisson, a transformação (37)

$$Z_i = 2 \sqrt{Y_i + \frac{3}{8}} \tag{4.20}$$

torna  $Z_i$  uma v.a. com média aproximadamente igual a  $2\sqrt{g_i + \frac{1}{8}}$  e variância unitária onde

$$g_i = E[Y_i]$$

Isto equivale ao modelo aditivo, como o mostrado na figura 4.7, onde para dado  $g_i$  tem-se,

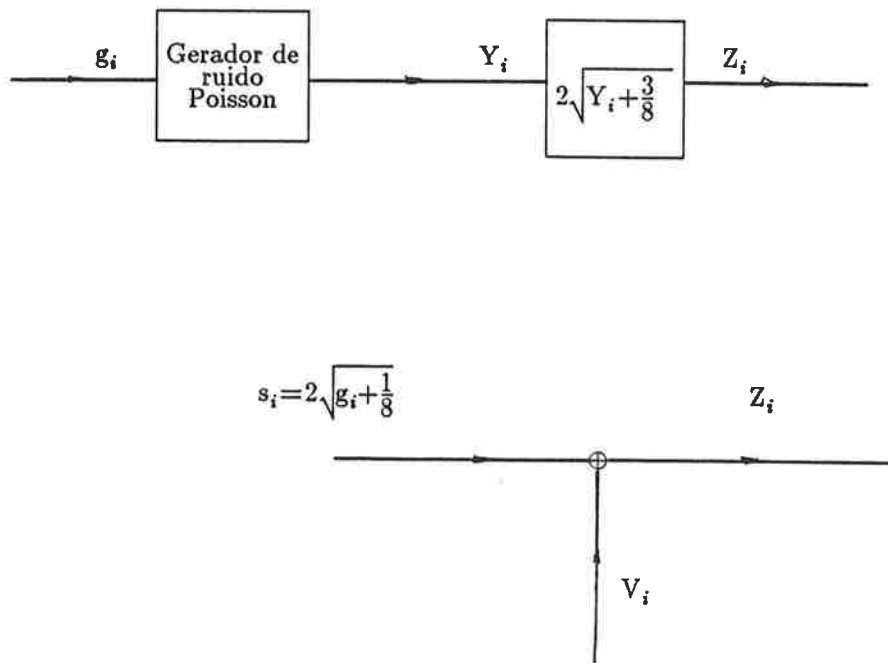


Figura 4.7. Transformação de Anscombe e modelo aditivo

onde  $E[V_i] \cong 0$  (4.21)

$\text{var}[V_i] \cong 1$  (independente do sinal) (4.22)

$V_i$  tem distribuição aproximadamente normal e não-correlacionados entre si.

$\text{cov}(\vec{V}) \cong I$  pois  $Y_i$   $i=1, M$  são independentes (4.23)

Nestas condições, o procedimento para a estimativa das projeções é local e envolve a estimativa

de  $s_i$  a partir de  $z_i$  conhecendo-se a estatística do ruído aditivo. Uma vez estimado o  $s_i$ , obtém-se a estimativa de  $g_i$  pela transformação inversa de Anscombe:

$$\hat{g}_i = \left(\frac{s_i}{2}\right)^2 - \frac{1}{8} \quad (4.24)$$

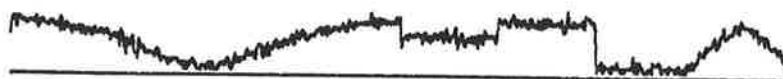
A figura 4.8 ilustra o sinal da figura 4.3 após a transformação de Anscombe sobre o sinal sem ruído (a) e sobre o sinal ruidoso (b), com valores mínimo, médio e máximo iguais a (2.345, 8.349 e 12.9) e (1.225, 8.297 e 15.7) respectivamente. Nota-se que a faixa dinâmica do sinal foi reduzida assim como a do ruído. A figura 4.9 mostra o ruído aditivo  $\vec{v} = \vec{z} - \vec{s}$  e o histograma deste sinal, onde a média e a variância obtidas foram -0.052 e 0.92, respectivamente. Como o ruído é, aproximadamente, independente do sinal, os valores teóricos previstos para a média e variância (equações 4.21 e 4.22) são 0 e 1, respectivamente. Percebe-se, como previsto, a distribuição aproximadamente gaussiana do ruído após a transformação do sinal com distribuição de Poisson.



a)

Arq.: anscom\_o

(min= 2.345,med= 8.349,max= 12.9)



b)

Arq.: anscombe.dat

(min= 1.225,med= 8.297,max= 15.7)

Figura 4.8 Resultado da aplicação da transformação de Anscombe a) sobre o sinal sem ruído da figura 4.3 em b) sobre sinal ruidoso.

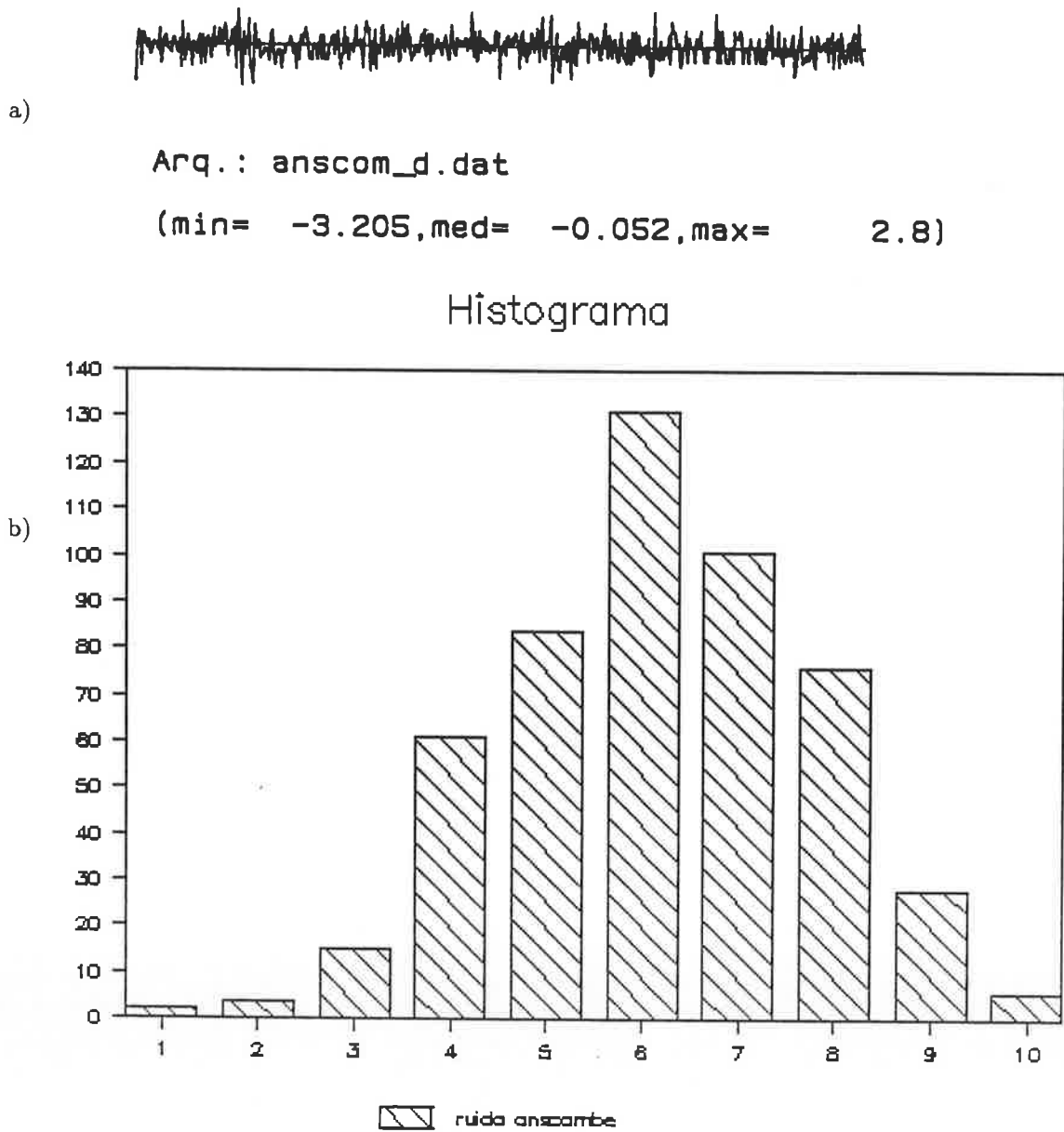


Figura 4.9 a) Ruído aditivo  $\vec{v}$  após a transformação de Anscombe do sinal com distribuição de Poisson e em b) o histograma correspondente. Variância medida do ruído=0.92

#### 4.3.1 Estimador Bayesiano MAP local

Analogamente ao ítem 4.2.1, maximiza-se a  $\text{Prob}[\vec{S} | \vec{z}]$  (ver figura 4.7), onde a distribuição de  $\vec{S}$  é assumida gaussiana multivariada, não-correlacionada, com média e variância não estacionárias (NMNV). No entanto, neste caso a  $\text{Prob}[\vec{Z} | \vec{s}]$  é gaussiana ao invés de Poisson, com variâncias unitárias.

Procedendo de forma similar ao descrito em 4.2.1, e assumindo que os elementos de  $\vec{S}$  são não correlacionados entre si, tem-se

$$\left( \frac{1}{\text{var}(V_i)} + \frac{1}{\text{var}(S_i)} \right) \cdot s_i = \frac{z_i}{\text{var}(V_i)} + \frac{1}{\text{var}(S_i)} \cdot E[S_i]$$

donde, o estimador local:

$$s_i = \beta_i \cdot z_i + (1 - \beta_i) \cdot E[S_i] \quad (4.25)$$

$$\beta_i = \frac{\text{var}(S_i)}{1 + \text{var}(S_i)} \quad (4.26)$$

onde  $E[S_i]$  e  $\text{var}(S_i)$  são estimados no sinal ruidoso ( $Z$ ), da mesma forma mencionada na seção 4.2.1 (Equações 4.11- 4.13).

A figura 4.10 e a tabela 4.2 mostram os resultados após a estimativa e a transformação inversa, usando janelas de 3 e 5 elementos para a estimativa das médias e variâncias locais. A melhoria obtida foi de 3.74 e 4.85 dB respectivamente.

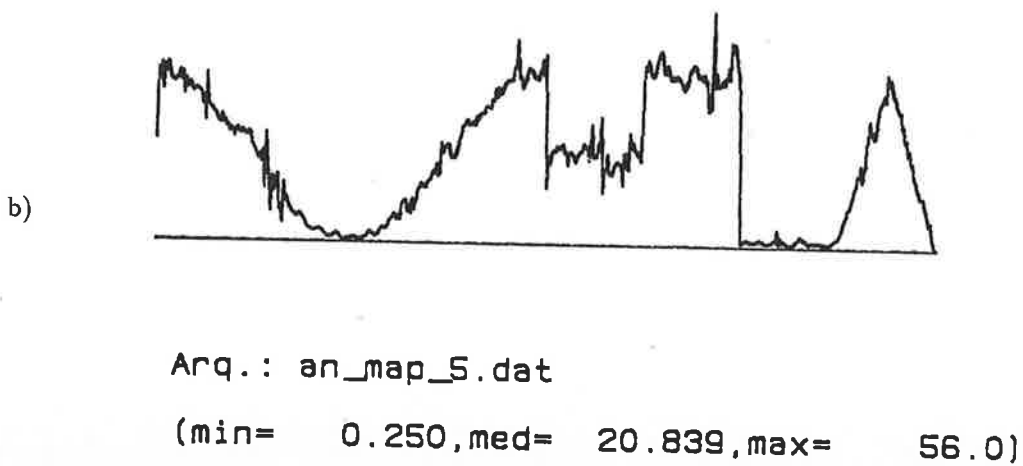
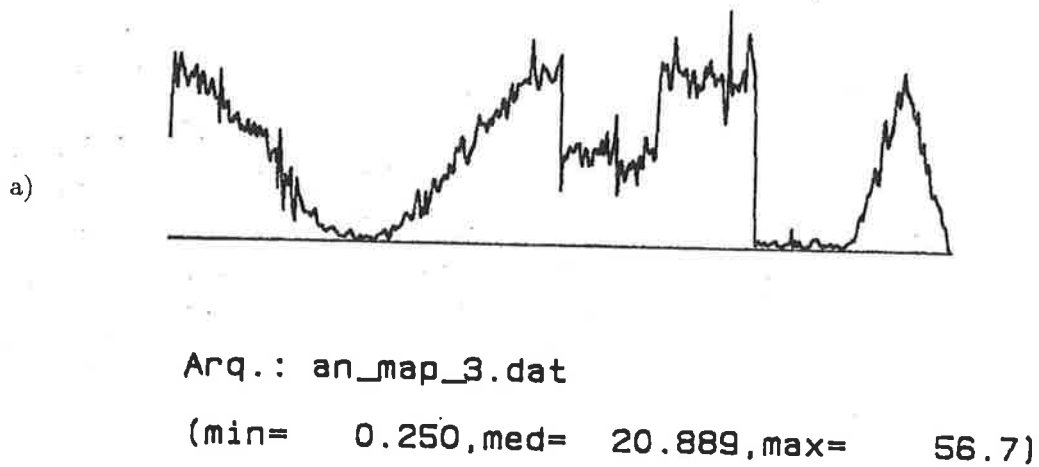


Figura 4.10. Resultado da estimativa através da transformação de Anscombe e o estimador Bayesiano MAP , o qual coincide, neste caso, com o estimador de Wiener. a) com janela de 3 elementos para a estimativa das médias e variâncias locais; b) com janela de 5 elementos . Melhoria de 3.74 e 4.85 dB respectivamente.

#### 4.3.2 Estimador linear com menor erro médio quadrático (Wiener local)

Utilizando-se procedimento similar ao da seção 4.2.2, porém após a transformação de Anscombe, ou seja, estimativa com menor erro médio quadrático em modelo aditivo, com ruído Gaussiano e independente do sinal, onde o sinal segue o modelo NMNV, tem-se:

$$\hat{s}_i = E[S_i] + \frac{\text{var}(S_i)}{\text{var}(S_i) + \text{var}(V_i)} \cdot (z_i - E[Z_i]) \quad (4.27)$$

$$\text{onde } E(V_i) = 0$$

$$\text{var}(V_i) = 1$$

$$\text{var}(Z_i) = \text{var}(S_i) + \text{var}(V_i)$$

$$E[Z_i] = E[S_i]$$

donde,

$$\hat{s}_i = E[S_i] + \frac{\text{var}(S_i)}{1 + \text{var}(S_i)} \cdot (z_i - E[S_i]) = \beta_i \cdot z_i + (1 - \beta_i) \cdot E[S_i] \quad (4.28)$$

$$\text{onde } \beta_i = \frac{\text{var}(S_i)}{1 + \text{var}(S_i)}$$

que coincide com o estimador Bayesiano, como era esperado, uma vez que as distribuições do sinal  $S$  e do ruído  $V$  são gaussianas (simétricas). Deste modo, o resultado está ilustrado na figura 4.10.

### 4.3.3 Estimador Heurístico

Nota-se que os estimadores anteriores (MAP e Wiener) correspondem à média ponderada entre a média local e o sinal medido. Maeda e Murata (37) propuseram um filtro não-linear envolvendo a mediana no lugar do sinal medido. Sendo  $\vec{z}$  o sinal após a transformação de Anscombe das projeções  $\vec{y}$  (figura 4.7), tem-se:

$$\hat{s}_i = \beta_i \cdot \tilde{z}_i + (1 - \beta_i) \cdot \bar{z}_i \quad (4.29)$$

onde  $\tilde{z}_i$  : mediana local

$\bar{z}_i$  : média local

$$\beta_i = \frac{\text{variância local}}{\text{máxima variância}} \quad (4.30)$$

Desse modo  $\hat{s}_i$  representa a média ponderada entre a média e a mediana locais. Se a região apresentar variações pequenas (variância pequena), o fator  $\beta_i$  será pequeno, prevalecendo a influência da média local (alisamento), enquanto que para elevadas variações (bordas),  $\beta_i$  se aproxima de 1, prevalecendo a mediana local. Como a mediana tende a preservar bordas, espera-se que este método não degrade substancialmente a resolução.

A figura 4.11 e a tabela 4.2 mostram os resultados onde as janelas utilizada para estimar média, variância e mediana locais foram de 3 (J=1) e 5 (J=2) elementos. A melhoria obtida foi de 5.05 e 6.64 dB respectivamente.





a) Arq.: an\_mae\_3.dat  
(min= 0.250,med= 20.854,max= 51.2)

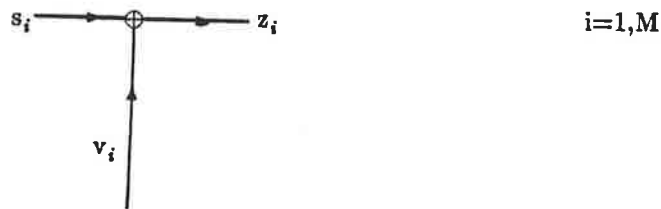


b) Arq.: an\_mae\_5.dat  
(min= 0.250,med= 20.798,max= 48.2)

Figura 4.11. Resultado obtido pelo estimador heurístico com o uso da transformação de Anscombe. a) com Janela de 3 elementos para estimar as médias, variâncias e medianas locais; b) com janela de 5 elementos . Melhoria de 5.05 e 6.64 dB, respectivamente.

4.3.4 Minimização de função custo (rugosidade)

É uma estimativa com informações determinísticas a priori (56). Considere o esquema abaixo, onde  $s_i$  representa o sinal a ser estimado,  $z_i$  o sinal medido e  $v_i$  o erro



Incorporando a informação a priori de que as projeções não têm variações bruscas de intensidade (suavidade) e que desejamos uma solução que minimize o erro médio quadrático, tem-se:

$$\text{custo}(\vec{s}) = \|\vec{v}\|_{\text{cov}(\vec{V})}^2 + \beta \cdot \|\text{L}\vec{s}\|^2 \quad (4.31)$$

onde  $\vec{v} = \vec{z} - \vec{s}$

L : operador Laplaciano definido por (Apêndice B):

$$(\text{L}\vec{s})_i = s_{i-1} - 2s_i + s_{i+1} \quad i=2, M-1 \quad (4.32)$$

$\beta$  : peso

$$\|\vec{v}\|_{\text{cov}(\vec{V})}^2 = \vec{v}^T \cdot \text{Cov}^{-1}(\vec{V}) \cdot \vec{v} \quad : \text{quadrado da norma do vetor erro normalizado} \quad (4.33)$$

Da equação 4.23,  $\text{cov}(\vec{V}) = \text{I}$ , logo

$$\|\vec{v}\|_{\text{cov}(\vec{V})}^2 = \vec{v}^T \vec{v} = (\vec{z} - \vec{s})^T (\vec{z} - \vec{s})$$

Nestas condições o custo é :

$$\text{custo}(\vec{s}) = \sum_{i=1}^M (z_i - s_i)^2 + \beta \cdot \sum_{i=2}^{M-1} (s_{i-1} - 2s_i + s_{i+1})^2 \quad (4.34)$$

derivando em relação a  $s_k$  e igualando a zero (minimização), tem-se:

$$2 \sum_{i=1}^M (z_i - s_i)(-\delta(i,k)) + 2\beta \sum_{i=2}^{M-1} (s_i - 2s_i + s_i) \cdot (\delta(i-1,k) - 2\delta(i,k) + \delta(i+1,k)) = 0$$

onde  $\delta(i,k)$ : função delta de Kronecker=1 para  $i=k$

$$= 0 \text{ para } i \neq k$$

$$s_k + \beta \cdot (s_{k-2} - 4s_{k-1} + 6s_k - 4s_{k+1} + s_{k+2}) = z_k \quad k=2, M-1 \quad (4.35)$$

Aplicando-se a transformada Z, onde T: distância amostral,  $w=2\pi f$  e f: variável no domínio da frequência, tem-se

$$S(w) + \beta \cdot S(w) \cdot (z^{-2} + z^2 - 4(z^{-1} + z^1) + 6) = Z(w)$$

$$\text{Como } z = e^{jwT} = \cos(wT) + j\text{sen}(wT)$$

$$z^{-1} + z^1 = 2\cos(wT)$$

$$z^{-2} + z^2 = 2\cos(2wT)$$

$$S(w) = \frac{Z(w)}{1 + \beta(2\cos 2wT - 8\cos(wT) + 6)} \quad (4.36)$$

o qual é um filtro recursivo de fase zero, passa-baixa. A figura 4.12a ilustra a resposta em frequência deste filtro para  $\beta=0.001, 0.01, 0.1, 0.2, 0.5, 1.0$  e a figura 4.12b para  $\beta=1, 5, 10, 20, 100, 1000$ . A solução deste estimador é dependente do peso ( $\beta$ ) dado à parcela referente à rugosidade. Para  $\beta=0$ , o sinal estimado é o próprio sinal medido, e para  $\beta \rightarrow \infty$  o sinal é extremamente alisado, tornando-se uma constante. Desta forma, a solução do menor custo percorre uma trajetória entre o sinal medido e o sinal fortemente alisado. A tabela 4.1 mostra o desempenho do estimador para diferentes  $\beta$ , obtendo o melhor índice para  $\beta \cong 5$ . A figura 4.13 mostra o resultado da estimativa para  $\beta=1$  e  $\beta=5$ , onde a melhoria foi de 4.63 e 4.85 dB respectivamente.

Para a implementação deste filtro, pode-se utilizar o Fast Fourier Transform (FFT unidimensional) para obter-se  $Z(w)$  e a FFT inversa. Uma alternativa ao emprego do FFT é utilizar a versão matricial para a minimização da função custo, possibilitando soluções iterativas. A equação 4.31 pode ser reescrita por,

$$\text{custo}(\vec{s}) = (\vec{z} - \vec{s})^T (\vec{z} - \vec{s}) + \beta (\mathbf{L}\vec{s})^T (\mathbf{L}\vec{s})$$

minimizando em relação a  $\vec{s}$ , tem-se

$$\frac{\partial \text{custo}(\vec{s})}{\partial \vec{s}} = 2(\vec{s} - \vec{z}) + 2\beta L^T L \vec{s} = \vec{0}$$

$$\vec{s} = (I + \beta L^T L)^{-1} \cdot \vec{z} \quad \text{o qual corresponde à eq.4.36 no domínio do espaço}$$

A solução iterativa decorre da equação anterior,

$$\begin{aligned} \vec{s} &= -\beta L^T L \vec{s} + \vec{z} \\ \vec{s}^{(n+1)} &= \vec{s}^{(n)} - \beta L^T L \vec{s}^{(n)} + (\vec{z} - \vec{s}^{(n)}) \\ \vec{s}^{(n+1)} &= (I - \beta L^T L) \vec{s}^{(n)} + (\vec{z} - \vec{s}^{(n)}) \end{aligned}$$

Nota-se que a solução acima se assemelha ao filtro de Wiener para o presente caso (56), onde  $E[\vec{V}] = \vec{0}$ , e  $\vec{S}$  e  $\vec{V}$  não-correlacionados:

$$\vec{s}_W = R_{SS} \cdot (I + R_{SS})^{-1} \cdot \vec{z}$$

onde  $R_{SS} = E[\vec{S}\vec{S}^T]$  : matriz de correlação do sinal S (abordagem Bayesiana).

Comparando-se o filtro de Wiener acima com as equações 4.15 e 4.27, pode-se constatar que são consistentes, pois como  $E[\vec{G}] = E[\vec{Y}]$ , pode-se re-escrever a equação 4.14 por:

$$\vec{Y} - E[\vec{Y}] = \vec{G} - E[\vec{G}] + \vec{U}$$

$$\text{colocando-se } \vec{s} = \vec{G} - E[\vec{G}]$$

$$\vec{z} = \vec{Y} - E[\vec{Y}]$$

$$\text{e como } R_{SS} = E[\vec{S}\vec{S}^T] = \text{cov}(\vec{G}) \quad \text{resulta,}$$

$$\vec{G} - E[\vec{G}] = \text{cov}(\vec{G}) \cdot (I + \text{cov}(\vec{G}))^{-1} \cdot (\vec{Y} - E[\vec{Y}]) \quad \text{o qual corresponde à}$$

equação 4.15.

Para a escolha do  $\beta$ , um critério alternativo (27) é impor que a contribuição das duas parcelas da equação 4.31 sejam equivalentes. Dessa forma,

$$\beta \cong \frac{\|\vec{V}\|^2}{\|L\vec{s}\|^2}$$

e o cálculo de  $\beta$  depende de conhecimentos a priori sobre o sinal original  $\vec{s}$ . Assumindo que a norma da rugosidade tolerada para  $\vec{s}$  é igual à norma do erro da estimativa, tem-se  $\beta=1$ .

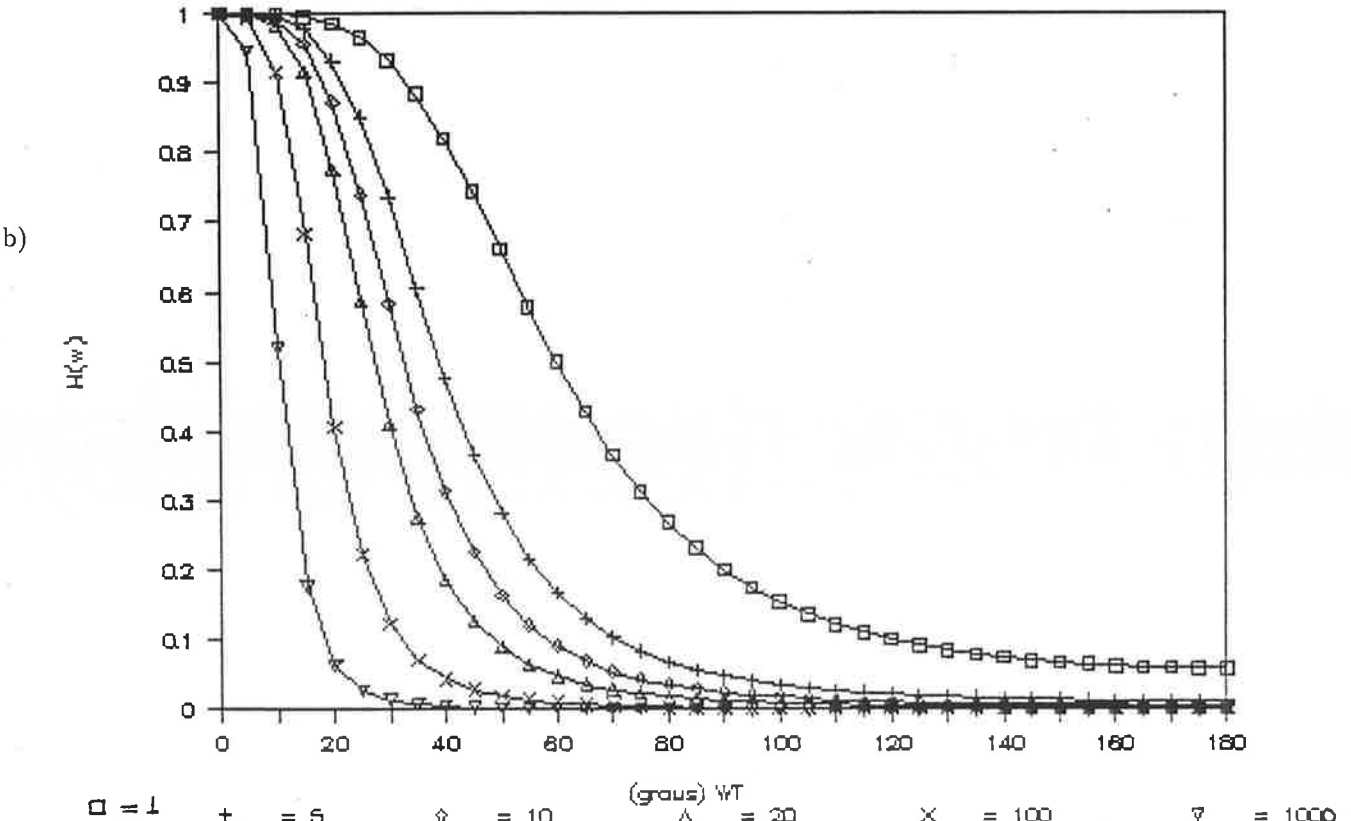
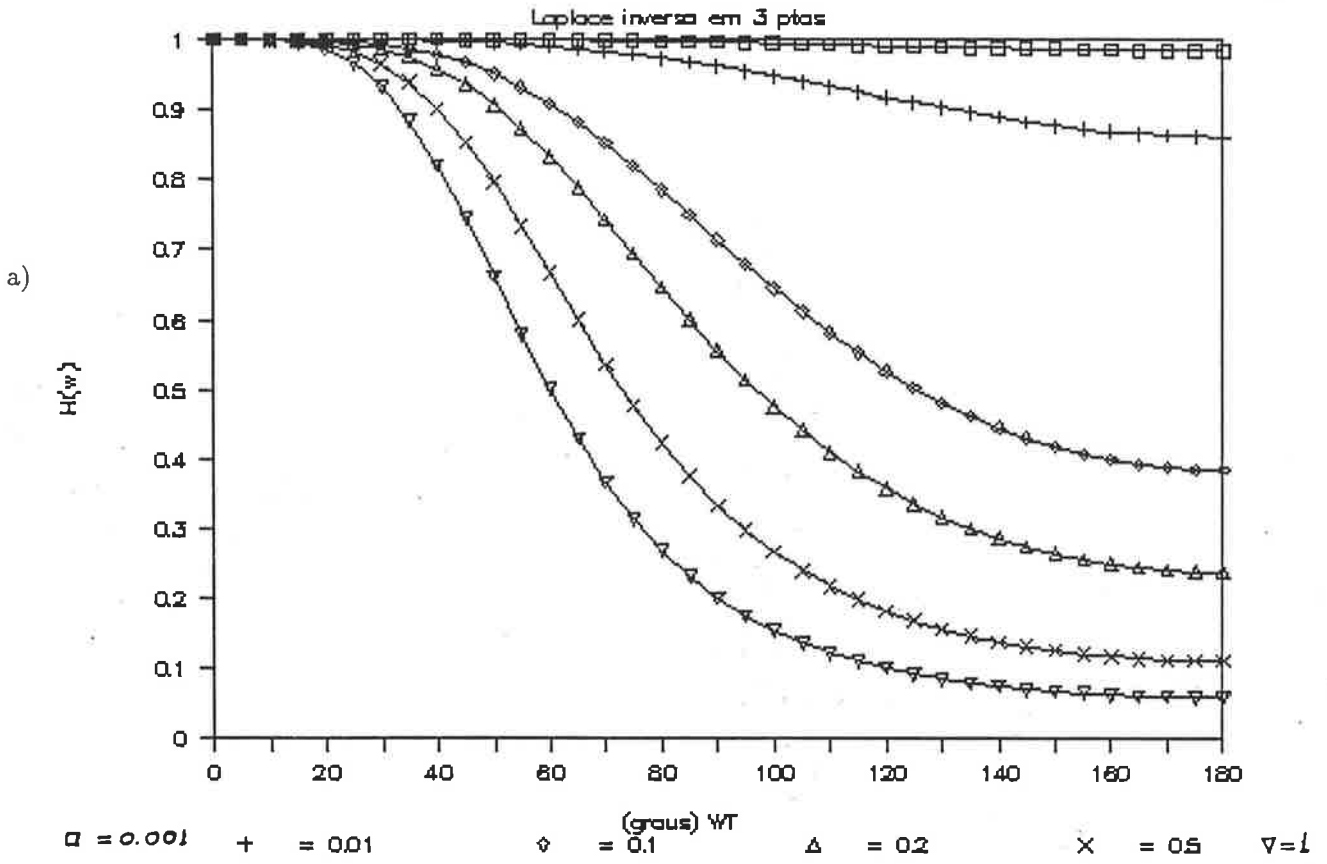


Figura 4.12. Resposta em frequencia do filtro do menor custo para diferentes  $\beta$ . a)  $\beta = 0.001, 0.01, 0.1, 0.2, 0.5$  e  $1.0$ ; b)  $\beta = 1, 5, 10, 20, 100, 1000$

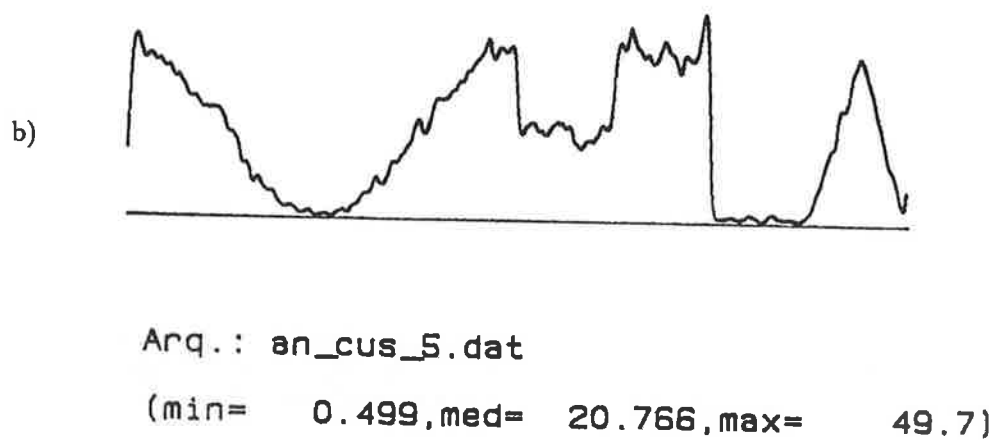
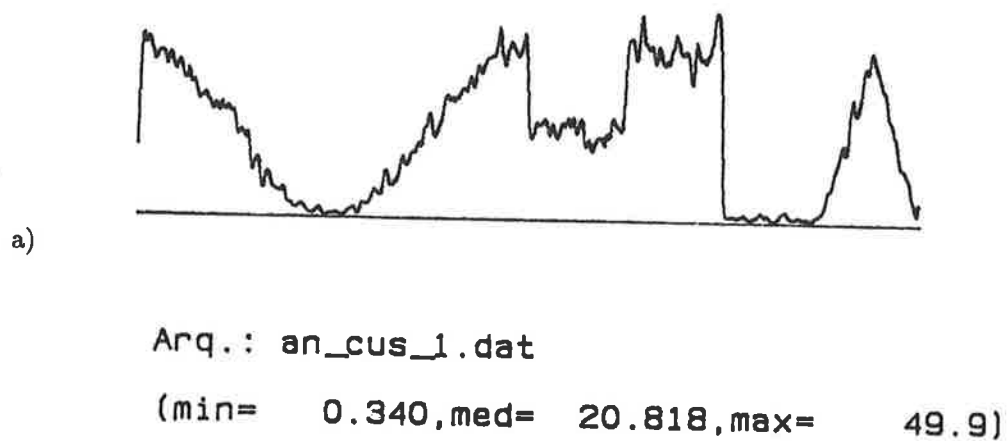


Figura 4.13 . Resultado da estimativa pela minimização da função custo, onde foi utilizado operador Laplace de 3 elementos. a)  $\beta=1$ ; b)  $\beta=5$  .Melhoria obtida de 4.63 e 4.85 dB respectivamente.

Tabela 4.1 Desempenho do estimador do menor custo em função do peso  $\beta$ .

$\beta$	REMQN	Melhoria (dB)
0.001	0.1708	0.03
0.01	0.1629	0.44
0.1	0.1294	2.44
0.2	0.1173	3.29
1	0.1005	4.63
2	0.0981	4.84
5	0.0980	4.85
6	0.0983	4.82
10	0.0996	4.71
20	0.1026	4.45
100	0.1147	3.48
1000	0.1446	1.47

Tabela 4.2 Tabela comparativa da Raiz quadrada do Erro Médio Quadrático Normalizado (REMQN), melhoria e tempo de execução<sup>(\*)</sup> entre os estimadores, aplicados diretamente sobre o sinal ruidoso (Poisson) e aplicados em conjunto com a transformada de Anscombe.

Estimador	Janela de 3 elementos			Janela de 5 elementos			
	REMQN	Melhoria (dB)	Tempo (s)	REMQN	Melhoria (dB)	Tempo (s)	
Poisson	MAP	0.1123	3.67	1.10	0.0991	4.75	1.48
	Wiener	0.1121	3.68	1.04	0.0992	4.74	1.43
Anscombe	MAP	0.1113	3.74	1.38	0.0980	4.85	1.76
	heurístico	0.0958	5.05	1.81	0.0798	6.64	2.75
	custo( $\beta=1$ )	0.1005	4.63	6.86			
	custo( $\beta=5$ )	0.0980	4.85	6.86			

(\*) baseado em um microcomputador compatível com IBM-PCAT, a 6 MHz e com co-processador matemático 80287.

#### 4.4 Reconstrução DTF: refinamentos

A aplicação direta da transformada de Fourier na reconstrução de imagens (CT) não tem sido amplamente utilizada devido a dificuldades na interpolação no domínio da frequência (52). Desse modo, é necessário criar condições mais adequadas para a interpolação.

Seja o domínio da imagem composto de  $n \times n$  pixels (quadrados), e  $m$  ângulos de projeção, com  $n$  amostras por ângulo de projeção.

Admitindo que a amostragem é adequada, i.é, a maior frequência espacial nas projeções ( $\rho_{\max}$ ) é menor ou igual à metade da frequência de amostragem (Critério de Nyquist):

$$\rho_{\max} \leq \frac{1}{2\Delta T} \quad \text{onde } \Delta T: \text{distância de amostragem}$$

Nestas condições, a banda de frequência está restrita a  $-\frac{1}{2\Delta T} \leq \rho \leq \frac{1}{2\Delta T}$

Considerando que no domínio da frequência (figura 4.15), deseja-se ter os  $n \times n$  coeficientes, em coordenadas cartesianas, para se poder utilizar o FFT bi-dimensional no cálculo da inversa, conclui-se que:

a) sendo  $\rho_{\max} = \frac{1}{2\Delta T}$ , tem-se  $\Delta\rho = \frac{1}{n\Delta T}$  (4.37)

- b) os coeficientes para os pontos fora do círculo de raio  $\rho_{\max}$  devem ser zerados;
- c) para se evitar o truncamento abrupto nas altas frequências para  $\rho$  próximo a  $\rho_{\max}$  (efeito de Gibbs), deve-se utilizar filtros que atenuem nas regiões de alta frequência;
- d) é necessário aumentar a densidade de pontos com transformadas de Fourier conhecidos (interpolação), de modo a melhorar a conversão para coordenadas cartesianas;

##### 4.4.1 Filtragem

Como as imagens originais são limitadas quanto à extensão no domínio do espaço, as projeções, por ângulo de projeção, também são limitadas. Portanto, a transformada de Fourier das projeções não é limitada em frequências. No caso discreto, pela teoria da amostragem, este fato gera uma distorção conhecida por "aliasing" (33). Esta distorção é minimizada por uma frequência de



amostragem elevada, e pelo fato da maioria dos dispositivos físicos de captação das projeções serem, inerentemente, filtros passa-baixa. Desse modo, um filtro digital deve apenas restringir os ruídos de alta frequência, sem prejudicar em demasia a resolução.

Propõe-se que o filtro tenha o formato Gaussiano, pois apresenta as seguintes vantagens:

- a) a transformada de Fourier de uma função do tipo de Gauss, também é Gaussiana (5), ou seja,

$$\text{se } h(r) = \frac{1}{\sqrt{2\pi}\sigma} \cdot e^{-0.5 r^2/\sigma^2} \quad (4.38)$$

$$H(\rho) = e^{-2\pi^2\sigma^2\rho^2} \quad (4.39)$$

- b) é um filtro de fase zero, portanto não acarreta distorções de fase, mas apenas atenuação em altas frequências;
- c) o filtro no domínio da frequência pode ser facilmente escrito em função do parâmetro FWHM (Full-Width at Half-Maximum) desejado, o qual está relacionado com a resolução no domínio do espaço;

$$h\left(\frac{\text{FWHM}}{2}\right) = h(0)/2 \text{ donde } \text{FWHM} = 2\sigma\sqrt{2\ln 2}$$

e o filtro em termos de FWHM,

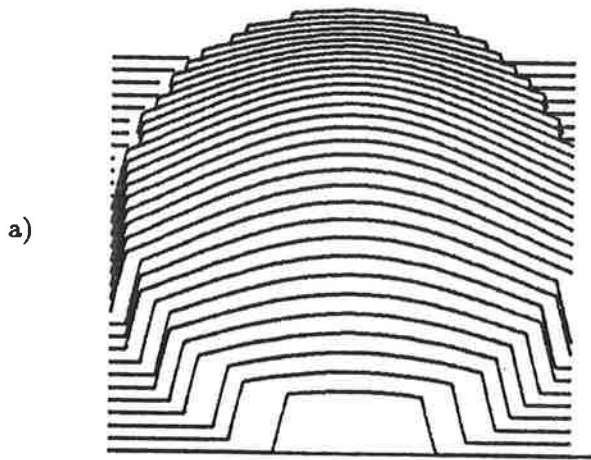
$$H(\rho) = e^{-\frac{\pi^2}{4\ln 2} \cdot \text{FWHM}^2 \cdot \rho^2} \quad (4.40)$$

Para projeções ruidosas, impondo atenuação de 20 dB na frequência  $\rho_{\max}$  em relação a  $\rho=0$ , tem-se,

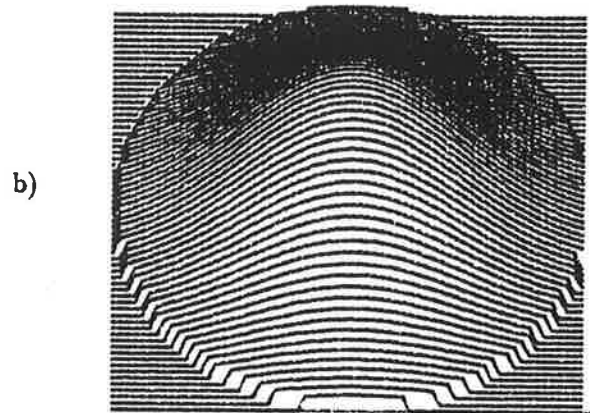
$$20 = 20 \log \frac{1}{H(\rho_{\max})} \quad \text{e pela equação 4.40}$$

$$\text{FWHM} \cong 1.61 \Delta T$$

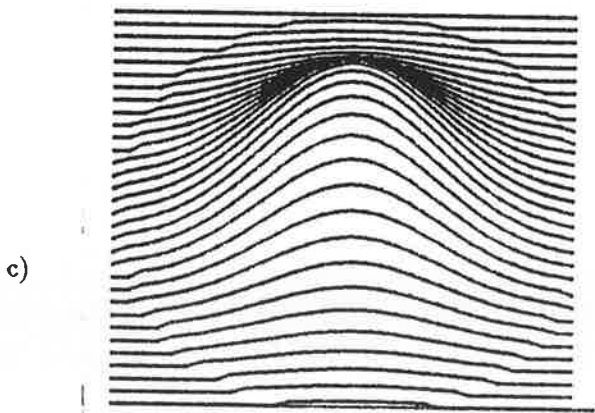
Portanto, para este valor de FWHM, a resolução estará pior do que 1.5 pixels devido à convolução com a função Gaussiana (equação 4.38). A figura 4.14 mostra a resposta em frequência (PSF) deste filtro para FWHM=1.0, 1.61, 2.0 e 2.59 respectivamente. Foi atribuído o valor zero para o filtro nas frequências externas ao círculo com raio igual a  $\rho_{\max}$ , pois as medidas permitem apenas o cálculo para  $\rho \leq \rho_{\max}$  (FFT das projeções).



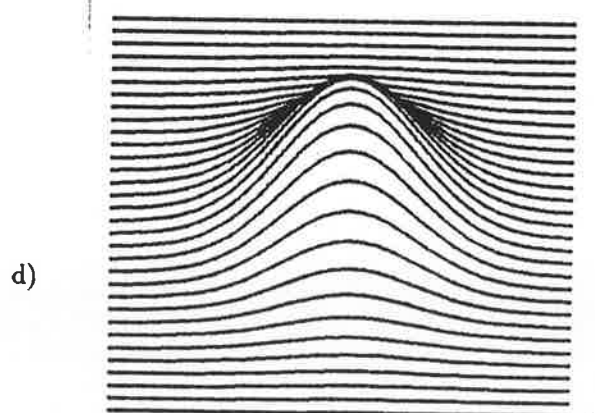
Arq.: gauss1\_0.fil



Arq.: gauss161.fil



Arq.: gauss2\_0.fil



Arq.: gauss2\_6.fil

Figura 4.14 Resposta em frequência dos filtros do tipo Gaussiano para a) FWHM=1.0 , b)1.61, c)2.0 e 2.59 respectivamente

4.4.2 Interpolação no domínio da frequência

Seja a figura 4.15, onde o ponto C representa um ponto discreto no sistema de coordenadas cartesianas, para o qual deseja-se estimar o valor a partir dos valores conhecidos em coordenadas polares. Sendo  $p_{11}$  e  $p_{12}$  as projeções (no domínio da frequência) na direção  $\theta_1$  mais próximos ao ponto C, e  $p_{21}$  e  $p_{22}$  na direção  $\theta_2$  a interpolação (de variáveis complexas) consiste em:

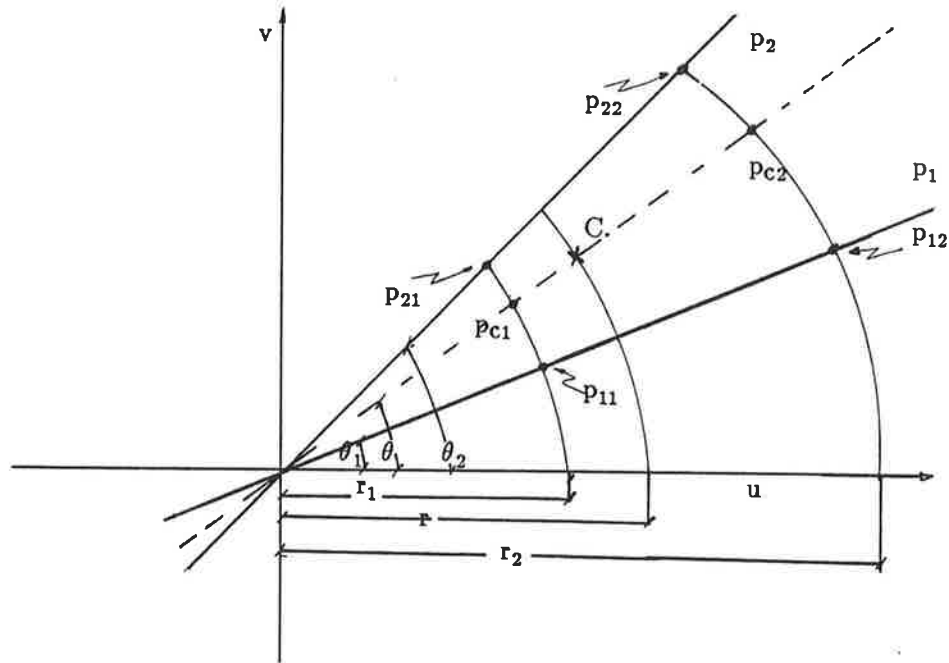


Figura 4.15 Interpolação bi-linear

a) estimar  $p_{c1}$  e  $p_{c2}$  ao longo da variação angular,

$$p_{c1} = p_{11} + \frac{\theta - \theta_1}{\theta_2 - \theta_1} (p_{21} - p_{11}) \tag{4.41}$$

$$p_{c2} = p_{12} + \frac{\theta - \theta_1}{\theta_2 - \theta_1} (p_{22} - p_{12}) \tag{4.42}$$

b) estimar  $p_c$  radialmente

$$p_c = p_{c1} + \frac{r - r_1}{r_2 - r_1} (p_{c2} - p_{c1}) \tag{4.43}$$

c) se  $\theta_2 = \pi$ , considere a propriedade  $g(t; \theta) = g(-t; \pi + \theta)$ , ou seja,

$$G(\rho; \pi + \theta) = G^*(\rho; \theta)$$

A interpolação angular descrita no ítem a) , se estendida ao longo da reta que passa por C e pela origem é uma combinação linear entre duas transformadas de projeções. Como a transformada de Fourier é um operador linear, a interpolação é equivalente a se estimar as projeções no domínio do espaço para o ângulo  $\theta$  e após realizar a transformada. A interpolação radial é mais problemática pois é efetivamente uma interpolação no domínio da frequência. Uma forma de diminuir o erro da interpolação radial é aumentar a densidade de pontos conhecidos. Isto pode ser obtido pela expansão com zeros para as projeções de cada ângulo de projeção (33). Considerando um sinal  $g(k\Delta T)$   $k=0, n-1$  periódica com período  $n\Delta T$ , a transformada de Fourier Discreta (DFT) é dada por (46),

$$G(l, \Delta\rho) = \sum_{k=0}^{n-1} g(k\Delta T) \cdot e^{-j2\pi \frac{k}{n} l} \quad l=0, n-1 \quad (4.44)$$

Construindo-se outro sinal  $g'(k\Delta T)$ ,  $k=0, n'-1$  com período  $n'\Delta T$  onde  $n' \geq n$ ,

$$\begin{aligned} g'(k) &= g(k) \quad 0 \leq k < n \\ &= 0 \quad n \leq k < n' \end{aligned} \quad (\text{expansão com zeros}) \quad (4.45)$$

A transformada de  $g'(k)$  é dada por:

$$\begin{aligned} G'(l) &= \sum_{k=0}^{n'-1} g'(k\Delta T) \cdot e^{-j2\pi \frac{k}{n'} l} = \sum_{k=0}^{n-1} g(k\Delta T) \cdot e^{-j2\pi \frac{k}{n} l \cdot \frac{n}{n'}} \\ &= G\left(\frac{n}{n'}, l\right) \end{aligned} \quad (4.46)$$

Portanto, se o fator de expansão  $\left(\frac{n}{n'}\right)$  for igual a 2, pode-se obter os valores intermediários de  $G(l)$  a partir de  $G'(l)$ . Fator de expansão igual a 4 é um bom valor levando-se em consideração a melhoria na qualidade da reconstrução e a degradação no tempo de processamento (15).

O número de operações de multiplicação para o método DTF com fator de expansão igual a 4 é, supondo que o número de ângulos de projeção  $m=2n$ ,

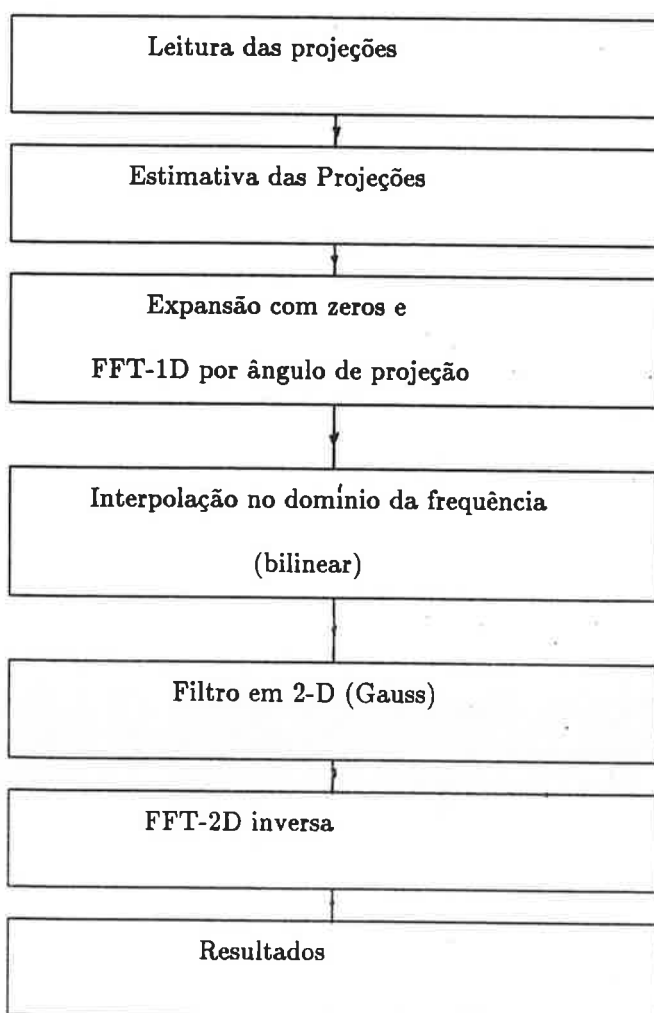
$$m \cdot 4n \log_2(4n) + 5n^2 + 2n (n \log_2 n) = 10n^2 \log_2 n + 21n^2$$

tornando-se competitivo, em termos de velocidade de processamento, a partir de  $n=64$ , se comparado ao FRP, devido principalmente à necessidade de retroprojeção para o FRP. A retroprojeção envolve, basicamente, operações de soma, mas para grandes imagens com elevado número de projeções, a retroprojeção demanda tempo apreciável..

A figura 5.6 e a tabela 5.1 mostram os resultados da reconstrução de um cilindro empregando este método.

#### 4.4.3 Implementação do DTF

O fluxograma da implementação computacional do DTF está representado pelo diagrama abaixo, onde no bloco 'Estimativa das projeções' estão incluídos todos os estimadores previamente discutidos.



#### 4.5 Conclusões

As projeções degradadas pelo ruído de Poisson podem ser estimadas efetivamente por estimadores que considerem a natureza estatística do ruído, obtendo-se melhorias da ordem de 5 dB.

A tabela 4.2 mostra, em termos do menor erro médio quadrático, que a janela de 5 elementos para o cálculo da estatística local é mais eficiente comparada à de 3 elementos, em detrimento do tempo de processamento.

Mostra-se também que a transformação de Anscombe melhora a qualidade da estimativa, com tempo maior de processamento.

Dentre os estimadores analisados, o de melhor desempenho, em termos do REMQN, é o estimador heurístico conjugado à transformação de Anscombe. Este desempenho se deve, fundamentalmente, ao filtro não-linear da mediana, que em diversos casos supera os filtros lineares ótimos (48).

É apresentado um estimador que minimiza a função custo, no caso, a rugosidade. Este estimador apresenta algumas características adicionais interessantes como a facilidade de se controlar o nível da filtragem nas projeções pelo parâmetro  $\beta$  (figura 4.12), e ter desempenho comparável a outros estimadores ótimos.

## 5-RESULTADOS

- 5.1. Simulações
- 5.2. Medida de qualidade da reconstrução do cilindro
- 5.3. Reconstrução para projeções sem ruído
- 5.4. Reconstrução do cilindro na presença de ruído Poisson
- 5.5. Reconstrução de estruturas mais complexas na presença de ruído Poisson
- 5.6. Reconstrução de fatia do cérebro a partir de projeções reais obtidas pelo PET
- 5.7. Conclusões

As diversas formas de se estimar as projeções a partir de projeções ruidosas, discutidas no capítulo 4, e a subsequente reconstrução da imagem são analisadas neste capítulo.

A reconstrução é realizada pelo método DTF (aplicação Direta da Transformada de Fourier) modificado, conforme descrito na seção 4.4, e a FRP (Filtragem-Retroprojeção). Os resultados são comparados com métodos clássicos como a CRP com filtro de Shepp-Logan (método da transformada), FRP com rampa, o ART (método determinístico de expansão em série), e o EM (método estatístico iterativo).

Para a comparação foi necessário criar um ambiente computacional para simulações, envolvendo:

- a) geração das projeções a partir da imagem original;
- b) geração de ruído Poisson nas projeções;
- c) implementação dos outros métodos de reconstrução : CRP, FRP, EM, e o ART;
- d) medida de qualidade da reconstrução;

### 5.1 Simulações

A imagem original de dimensões 32 x 32 pixels é constituída de um cilindro de raio correspondente a 8 pixels e cujo centro está posicionado em  $x=2$  e  $y=5$  pixels em um sistema de coordenadas cartesianas, cuja origem coincide com o centro do quadro, conforme mostrado na figura 5.1 . Foi utilizado um cilindro não-centrado para tornar a comparação mais efetiva, pois cilindros centrados, devido à simetria, apresentam projeções com variação no sentido angular nula (banda angular igual a zero).

A emissão total simulada do objeto foi de 10000 contagens, i.e,  $\sum_{i=1}^{1024} f_i = 10000$ , resultando no valor médio de 9.76 emissões por pixel na imagem e de 49.7 emissões por pixel do cilindro. Foi suposta a presença de 32 sensores igualmente espaçados para cada ângulo de projeção, de modo a cobrir todas as projeções do círculo de diâmetro igual a 32 pixels, interno ao quadro, ou seja, a distância amostral



nas projeções (feixes paralelos) é igual à dimensão de um dos lados do pixel. Foram utilizados 64 ângulos de projeção ao redor da imagem (figura 2.1), igualmente espaçados entre 90 e 270 graus, sendo a primeira projeção (traçado superior da figura 5.2) exatamente a 90 graus. A figura 5.2 ilustra o sinograma (transformada de Radon) obtido, onde cada linha corresponde às projeções de um ângulo de projeção. Os valores mínimo, médio e máximo deste sinograma resultaram, respectivamente em 0.0, 4.88 e 12.4.

As dimensões da imagem (32x32) foram escolhidas considerando o custo computacional envolvido, tanto no que se refere ao tempo de processamento quanto à memória disponível no microcomputador. O número de ângulos de projeção (64) e o número de amostras (32) por ângulo de projeção seguiram as seguintes considerações:

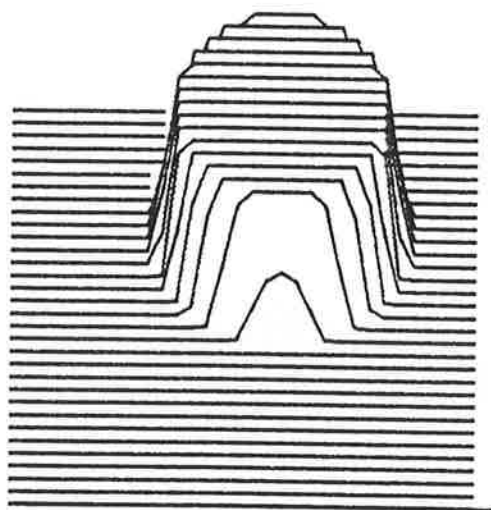
- a) para os métodos da transformada, que envolvem integral de linha, a distância amostral nas projeções deve ser igual à dimensão do pixel, donde o número de amostras ser igual a 32;
- b) para métodos que envolvem retroprojeção, a relação entre o número de ângulos de projeção ( $N_{ang}$ ) e o número de amostras por ângulo ( $N_{amostras}$ ) de projeção deve ser maior do que 1.57 para se evitar grandes erros numéricos na retroprojeção (33):

$$N_{ang} > \frac{\pi}{2} \cdot N_{amostras} + 1$$

- c) para métodos iterativos, deve-se considerar que a relação  $N_{ang}/N_{amostras}$  seja maior do que 1.5 para se evitar o efeito da propagação e amplificação de ruídos estatísticos na reconstrução (20);
- d) para o método DTF, as dimensões da imagem e das projeções devem ser potência de 2, para se utilizar o FFT com maior eficiência;

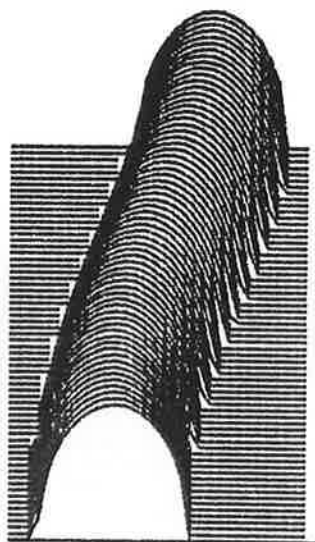
Os objetos presentes na imagem podem ter o formato de elipses. As projeções foram geradas, analiticamente, a partir das informações sobre a posição do centro, dimensões do eixo menor e maior, orientação e amplitude das elipses (21).

A geração do ruído Poisson foi implementada de acordo com a proposta descrita por Lo (34). Incorporando-se o ruído Poisson, conforme descrito no modelo estatístico do sistema (seção 3.4), resulta na figura 5.3 (amostra), com valores ( $min=0$ ,  $med=4.94$  e  $max=26.0$ ).



Arq.: c3232\_im.dat  
(min= 0.000,med= 9.766,max= 49.7)

Figura 5.1. Imagem objeto sem ruído (imagem original). Discretização em 32 x 32 pixels, total de 10000 contagens, com valor médio de 49.7 emissões/pixel do cilindro.



Arq.: c3232\_pr.dat  
(min= 0.000,med= 4.883,max= 12.4)

Figura 5.2. Sinograma (transformada de Radon) da figura 5.1. Cada linha corresponde à projeção para um determinado ângulo  $\theta$  (64 ângulos) com 32 amostras cada.

Os algoritmos para a reconstrução (CRP, FRP, ART e EM) foram implementados de acordo com o capítulo 2. O DTF foi implementado conforme a seção 4.4. Nos métodos que não envolvem a transformada de Fourier, i.é, ART e EM, é necessária a matriz de projeção H. Devido à elevada dimensionalidade (1024 x 1024) desta matriz, e ao fato da mesma ser esparsa, foi pré-calculada uma tabela, que fornece os valores desta matriz, para dado ângulo de projeção e distância do centro do feixe ao centro do pixel. Esta distância (figura 5.4) assume o seu valor máximo para (diagonal do pixel + distância amostral)/2, que no presente caso é  $(\sqrt{2} + 1)/2$ , supondo distância amostral igual à unidade. Para qualquer distância acima deste valor, a contribuição na projeção é nula. A tabela foi elaborada para 50 subdivisões da distância máxima e 64 ângulos de projeção, sendo, portanto, possível manter esta tabela em memória. Para dado ângulo de projeção e distância, a área do pixel interceptada pelo feixe foi obtida subdividindo-se o pixel em 2500 pequenos pixels, e verificando se os centros dos mesmos pertenciam ao feixe. A precisão obtida foi melhor do que 0.001 por ângulo de projeção, onde o valor esperado da soma de todas as projeções paralelas é 1 por pixel.

## 5.2 Medida de qualidade da reconstrução do cilindro

Para se avaliar a qualidade da imagem reconstruída, adotou-se o REMQN (Raiz quadrada do Erro Médio Quadrático Normalizado) global (equação 4.1), o REMQN nas bordas do cilindro e o REMQN na região central do cilindro, visando medir os erros em regiões planas da imagem e nas bordas.

Consideraram-se como pixels pertencentes à região central do cilindro todos aqueles cuja distância ao centro era menor do que 70% do raio. A região para medir o efeito das bordas foi considerada entre 70% e 130% do raio, conforme mostrado na figura 5.5.

Além destes três índices, mediu-se também o tempo para a reconstrução completa da imagem, incluindo-se acesso a disco, transformação e estimativa. O equipamento utilizado foi um microcomputador compatível com IBM-PCAT, a 6MHz, e com co-processador matemático 80287. As

linguagens utilizadas foram C e Fortran77.

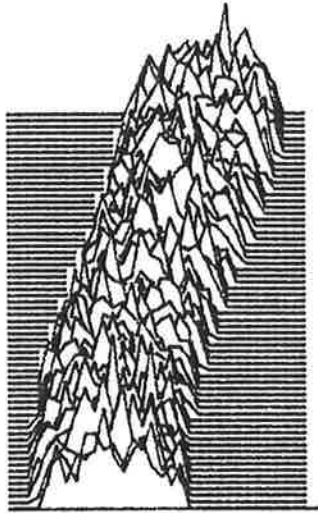
Nos métodos iterativos, para se avaliar a aproximação à máxima verossimilhança, mediu-se também o valor da função log likelihood, definida por:

$$L(\vec{f}^{(k+1)}) = \sum_{i=1}^M (y_i \cdot \ln z_i - z_i) \quad (5.1)$$

onde  $\vec{f}^{(k+1)}$  : imagem reconstruída na iteração (k+1)

$y_i$  ,  $i=1, M$  : projeções medidas

$z_i$  ,  $i=1, M$  :reprojeções obtidas com a imagem  $\vec{f}^{(k)}$



Arq.: c3232ppr.dat  
(min= 0.000,med= 4.937,max= 26.0)

Figura 5.3. Sinograma com ruído Poisson correspondente à figura 5.2 .

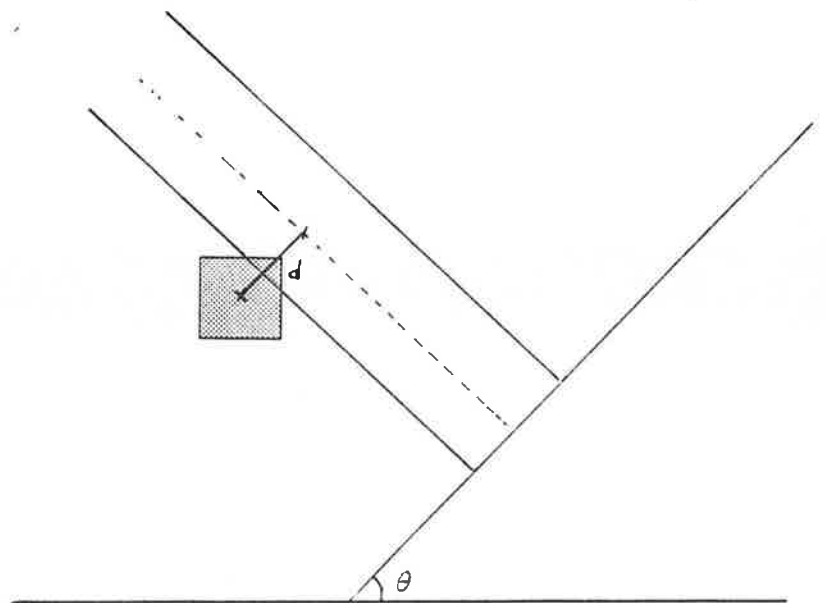


Figura 5.4 Esquema para cálculo da tabela para gerar a matriz de projeção H.

### 5.3 Reconstrução para projeções sem ruído

O objetivo desta seção é mostrar o desempenho dos diversos métodos de reconstrução em condições ideais, i.é, sem ruído, sem atenuação, sem espalhamento e com colimadores ideais. No caso dos métodos iterativos, mostra-se a evolução da reconstrução com o número de iterações.

A tabela 5.1 mostra os resultados, empregando-se os métodos da transformada DTF e FRP, EM (método estatístico), ART (método algébrico), conforme discutidos na seção 2. A figura 5.6 mostra o resultado, qualitativamente, para o DTF (com filtro tipo Gauss, FWHM=1 e 1.61) onde foi utilizada expansão com zeros de modo a completar 4 vezes o número de amostras por ângulo de projeção, o CRP (com filtro de Shepp-Logan) e o FRP com filtro rampa. A figura 5.7 ilustra para o caso iterativo EM em 2, 5, 10 e 30 iterações, e a figura 5.8 para o ART em 2, 10, 30 e 50 iterações. À medida que a iteração prossegue, o "overshoot" nas bordas é maior devido a erros numéricos computacionais, e a aproximações feitas tanto no cálculo das projeções, quanto no cálculo dos coeficientes da matriz de projeção H.

Pela tabela 5.1 pode-se concluir que o método DTF (FWHM=1.0), o CRP e o FRP reconstroem com qualidades equivalentes para projeções sem ruído, enquanto que os métodos iterativos apresentam um custo computacional muito mais elevado para atingir os mesmos índices de qualidade. O EM, como previsto teoricamente, apresenta valores de verossimilhança crescentes com a iteração, mas o erro não é monotonicamente decrescente com a iteração.

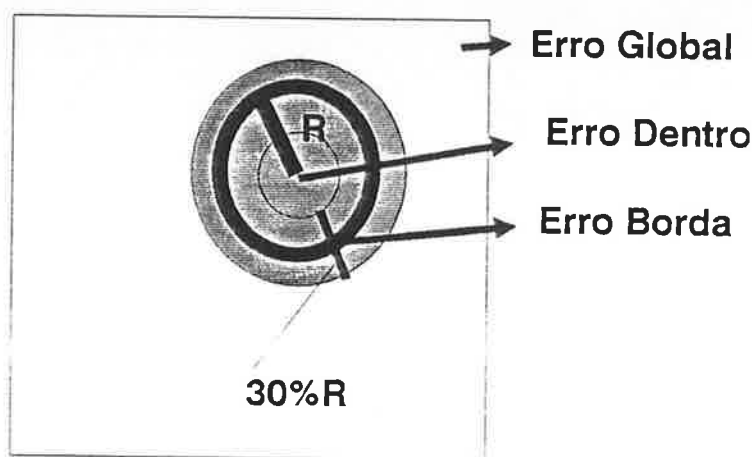


Figura 5.5. Definição das regiões : global, borda e central

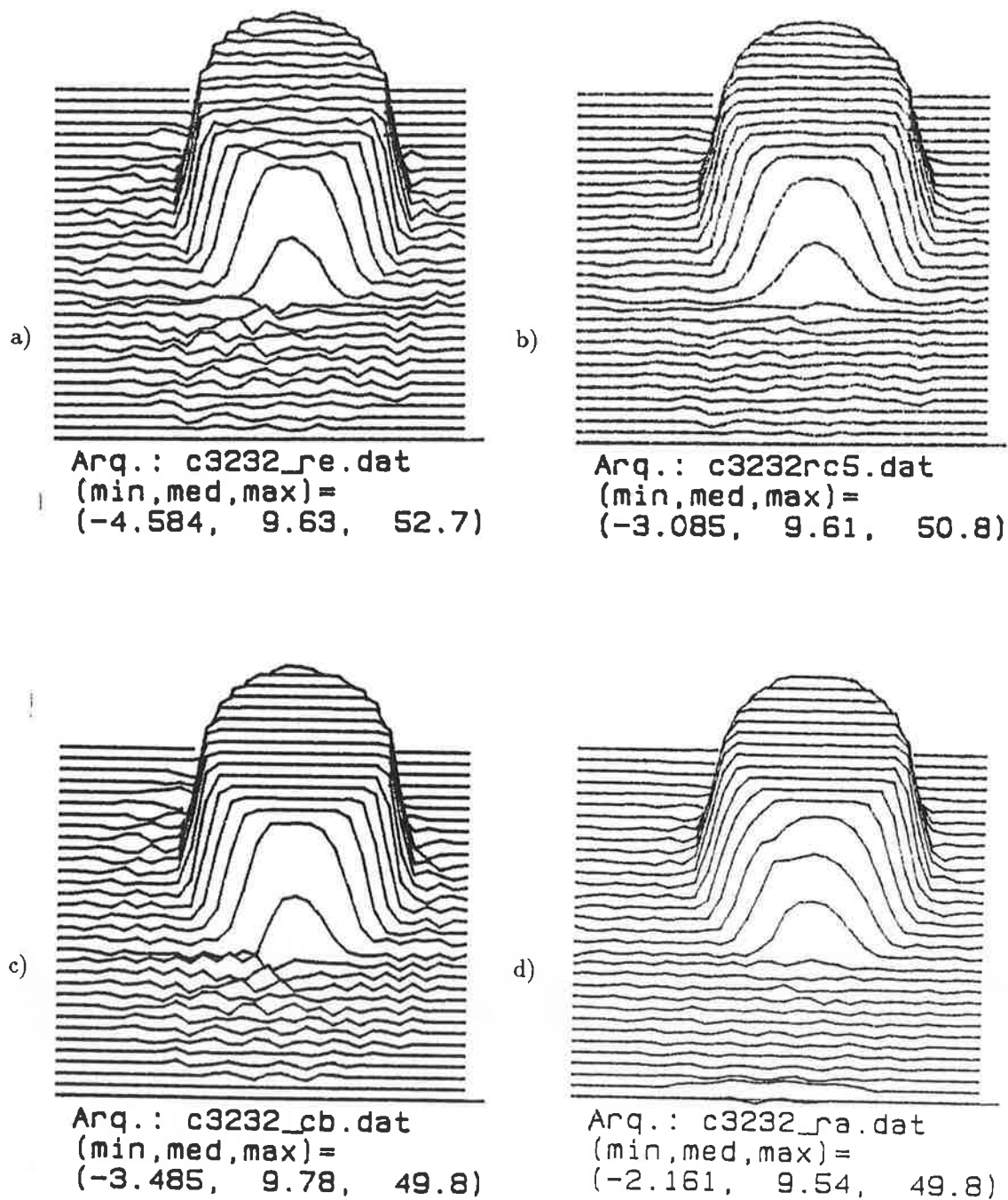
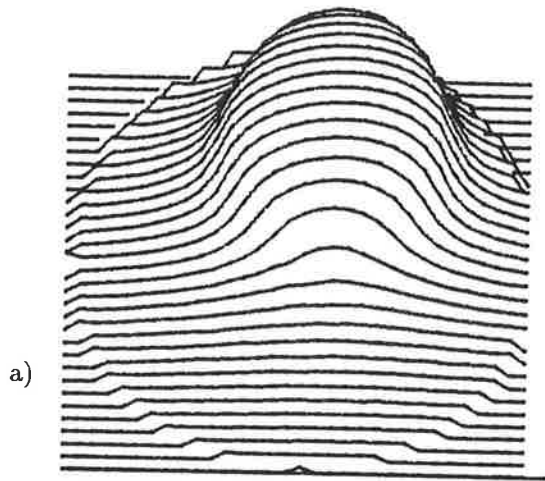
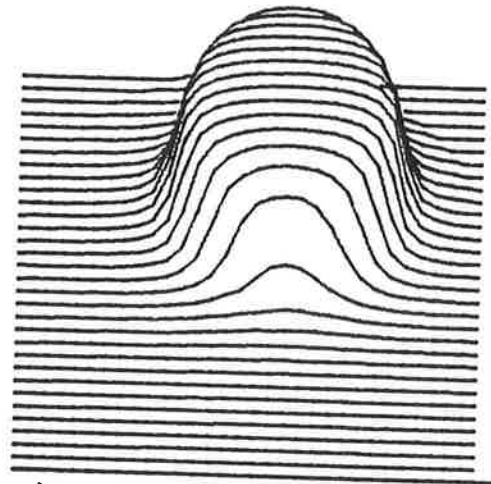


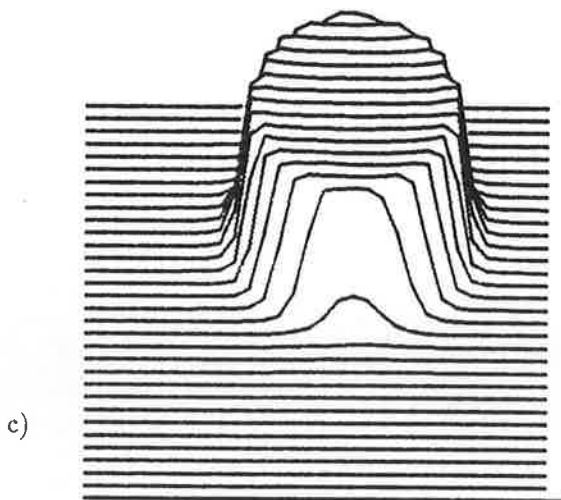
Figura 5.6. Reconstrução a partir de projeções sem ruído : a) DTF (filtro do tipo Gauss, FWHM=1); b) DTF com FWHM=1.61 ; c) CRP com filtro de Shepp-Logan ; d) FRP com rampa;



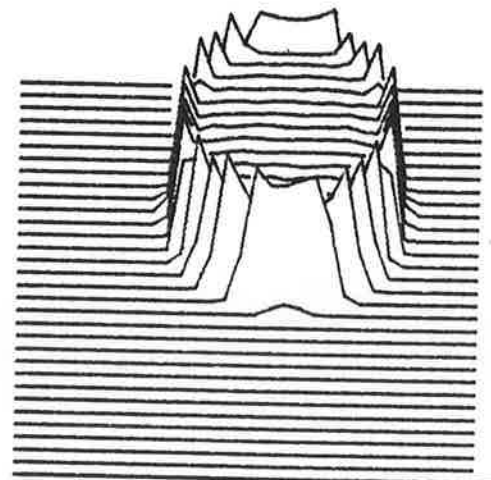
Arq.: c3232em1.dat  
(min,med,max)=  
( 0.000, 9.56, 38.3)



Arq.: c3232em2.dat  
(min,med,max)=  
( 0.000, 9.55, 50.9)



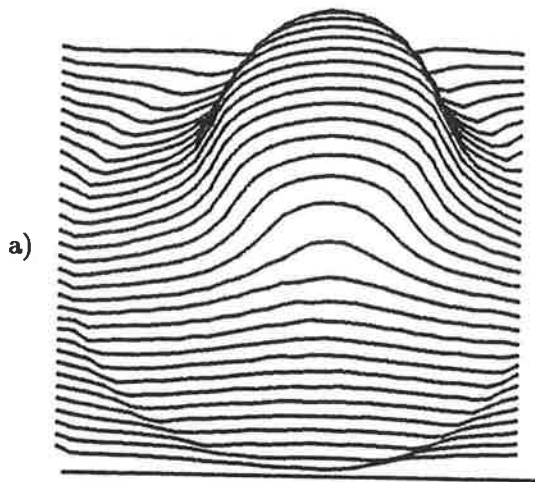
Arq.: c3232em3.dat  
(min,med,max)=  
( 0.000, 9.55, 51.6)



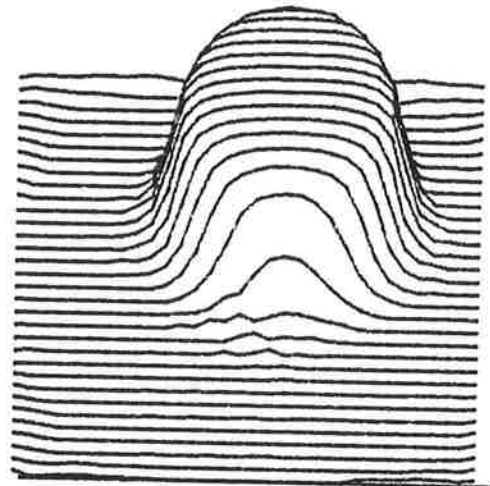
Arq.: c3232em4.dat  
(min,med,max)=  
( 0.000, 9.55, 75.7)

Figura 5.7 Reconstrução pelo método ML (EM) para projeções sem ruído. a)2 iterações;  
b)5; c)10; d)30

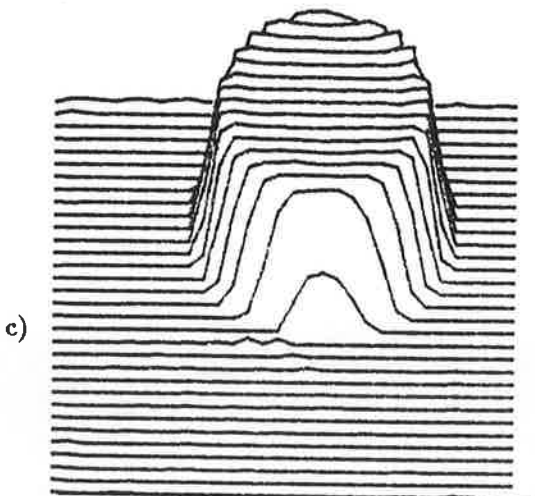




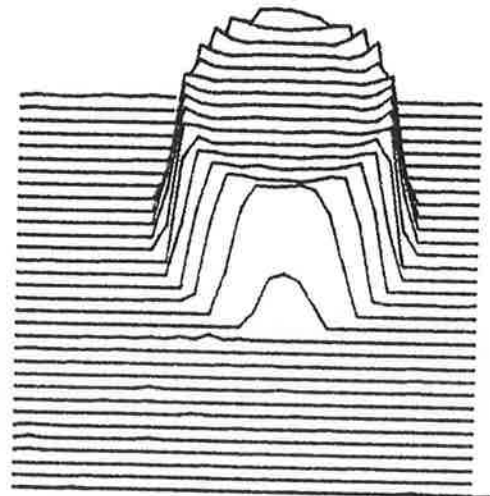
Arq.: c3232ar0.dat  
(min,med,max)=  
( 0.366, 9.94, 34.5)



Arq.: c3232ar2.dat  
(min,med,max)=  
( 0.000, 9.72, 49.5)



Arq.: c3232ar3.dat  
(min,med,max)=  
( 0.000, 9.62, 51.6)



Arq.: c3232ar6.dat

Figura 5.8 Reconstrução pelo método ART para projeções sem ruído. a)2 iterações; b)10;  
c)30; d)50

Tabela 5.1. Raiz quadrada do Erro Médio Quadrático Normalizado e do tempo<sup>(\*)</sup> de reconstrução de alguns métodos de reconstrução para o caso sem ruído

	REMQN			log	Tempo para reconstrução (s)
	central	bordas	global	likelihood	
DTF(FWHM=1.0)	0.0166	0.0849	0.0727		55.03
(FWHM=1.61)	0.0090	0.1413	0.1028		55.03
CRP(Shepp-Logan)	0.0006	0.0888	0.0708		39.38
FRP(rampa)	0.0169	0.1984	0.1428		79.92
EM 2	0.3042	0.4763	0.4236	9952.0	15.8 min
5	0.0381	0.2299	0.1659	12692.2	39.5 min
10	0.0092	0.1214	0.0857	13147.3	79.0 min
30	0.0368	0.2788	0.1979	13226.9	237.0 min
ART 2	0.3589	0.5051	0.4703	10450.4	72 min
5	0.1274	0.3218	0.2538	12346.2	181 min
10	0.0374	0.1960	0.1436	12927.1	361 min
30	0.0213	0.0590	0.0445	13159.2	1083 min
50	0.0256	0.0531	0.0418	13178.9	1810 min

(\*) baseado em um microcomputador compatível com IBM-PCAT a 6MHz, com co-processador matemático 80287

#### 5.4 Reconstrução na presença do ruído Poisson

A presença do ruído Poisson degrada substancialmente as projeções. A reconstrução ignorando o ruído é desastrosa conforme mostram as figuras 5.9 a) e b), e a tabela 5.2, onde os métodos empregados foram o DTF sem estimar as projeções, apenas com filtro tipo Gauss (FWHM=1 e 1.61), o CRP com filtro de Shepp-Logan, e FRP com rampa.

A figura 5.10 mostra os resultados obtidos pelo método estocástico e iterativo ML-EM, que incorpora o modelo de ruído Poisson em 2, 5, 10 e 30 iterações. Nota-se que a partir da décima iteração, a solução apresenta instabilidades devido ao ruído, aproximações e erros computacionais. A melhor solução para o presente caso está, aproximadamente, na quinta iteração (ver tabela 5.2).

A figura 5.11 e a tabela 5.2 mostram os resultados obtidos para o método ART em 2, 5, 10 e 30 iterações. Neste caso, a instabilidade se manifesta entre a décima e a trigésima iteração, sendo a melhor solução, aproximadamente, a 10 iterações.

Por outro lado, pre-processando as projeções de acordo com o descrito na seção 4.2 e 4.3, obtêm-se os resultados mostrados nas figuras 5.12 a 5.15. A janela empregada para o cálculo da estatística local (seção 4.2) foi de 5 elementos. A figura 5.12 corresponde à aplicação de estimadores Bayesianos MAP e Wiener diretamente sobre as projeções ruidosas, e reconstruídas pelo DTF com FWHM=1.0 e 1.61. Os resultados obtidos por estes dois estimadores Bayesianos são muito próximos, conforme previsto na seção 4.2. Quantitativamente, segundo a tabela 5.2, o melhor resultado (REMQN global=0.2521) corresponde ao MAP com FWHM=1.61.

As figuras 5.13 e 5.14 ilustram os resultados reconstruídos pelo DTF (FWHM=1.0 e 1.61), quando se aplica a transformação de Anscombe sobre as projeções antes da estimação. A figura 5.13 mostra os resultados obtidos com os estimadores MAP e heurístico, e a figura 5.14 utilizando o estimador do menor custo ( $\beta=1$ ) e ( $\beta=5$ ). Nota-se pela tabela 5.2, que utilizando-se os estimadores Bayesianos, a inclusão da transformação de Anscombe apenas melhora levemente a reconstrução na região central, piorando nas bordas. Nota-se também, que os estimadores que minimizam a função custo para  $\beta=1$ , produzem resultados comparáveis aos estimadores ótimos Bayesianos.

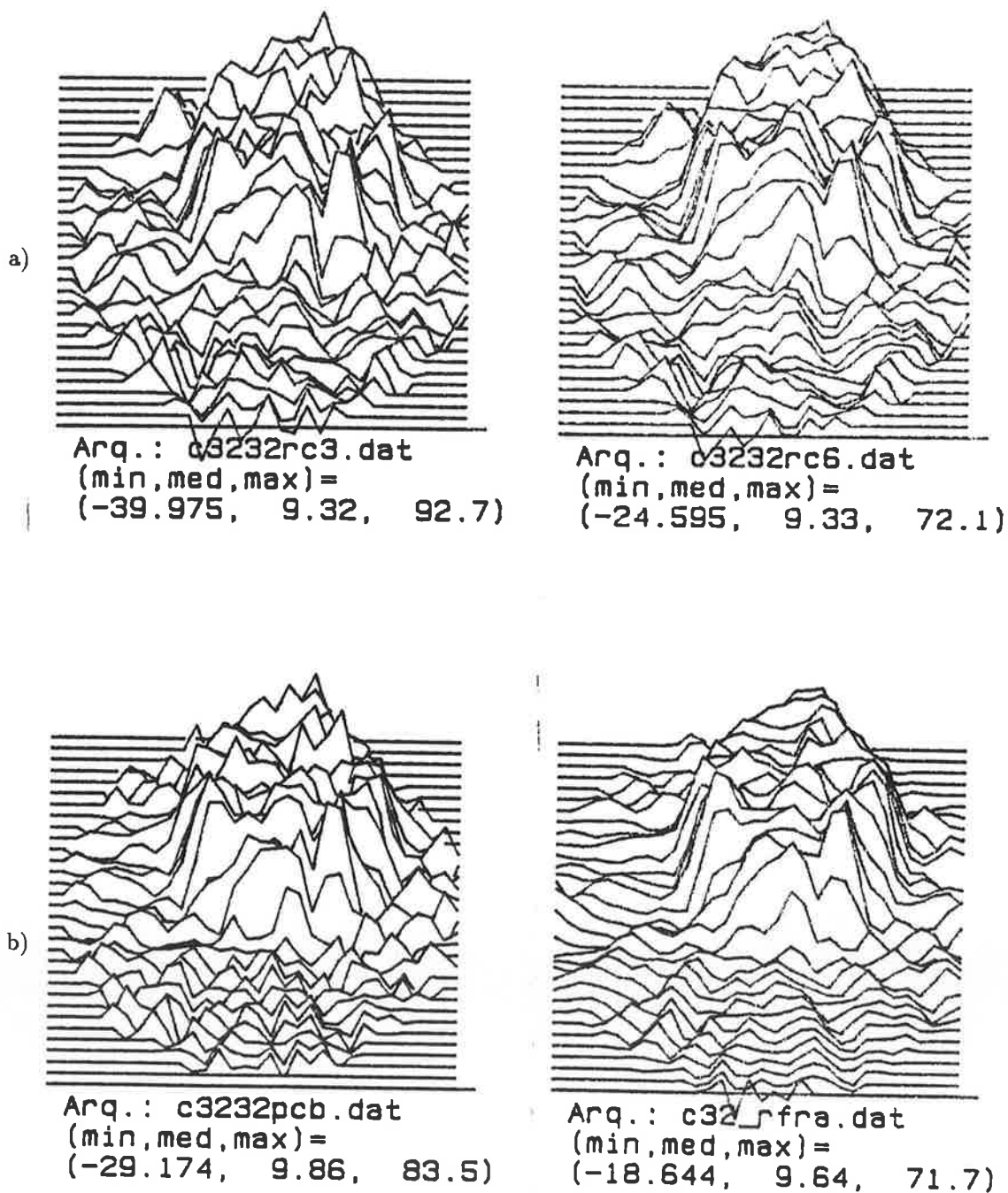


Figura 5.9 Reconstrução do cilindro a partir de projeções ruidosas , pelos métodos a)DTF sem estimar as projeções, com filtro do tipo Gauss com FWHM=1 e 1.61 ; b) CRP com filtro tipo Shepp-Logan e FRP com rampa respectivamente.

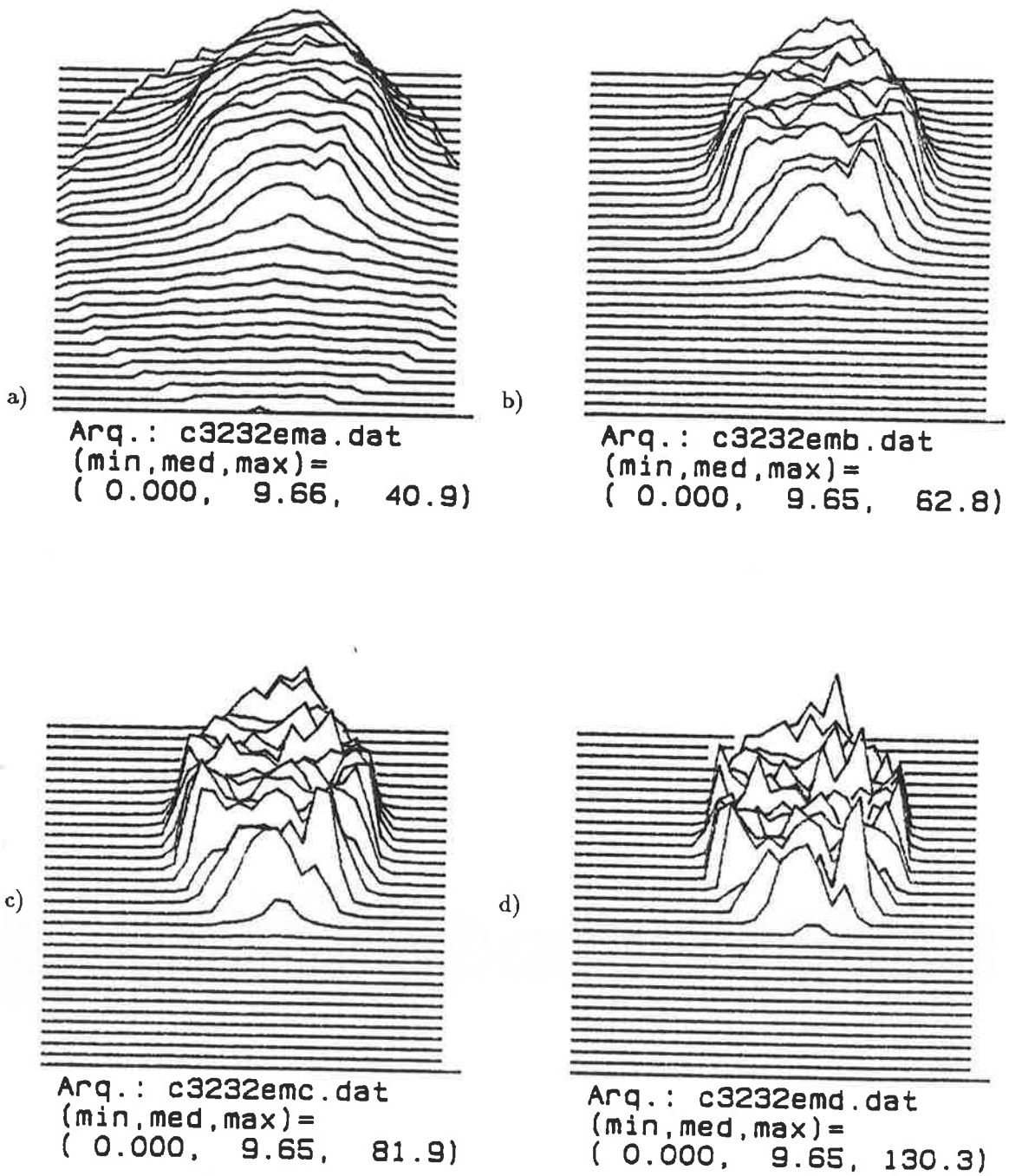


Figura 5.10 Reconstrução do cilindro a partir de projeções ruidosas , pelos método estocástico (ML) e implementado pelo algoritmo iterativo EM em a) 2 iterações; b) 5; c) 10 ; d) 30

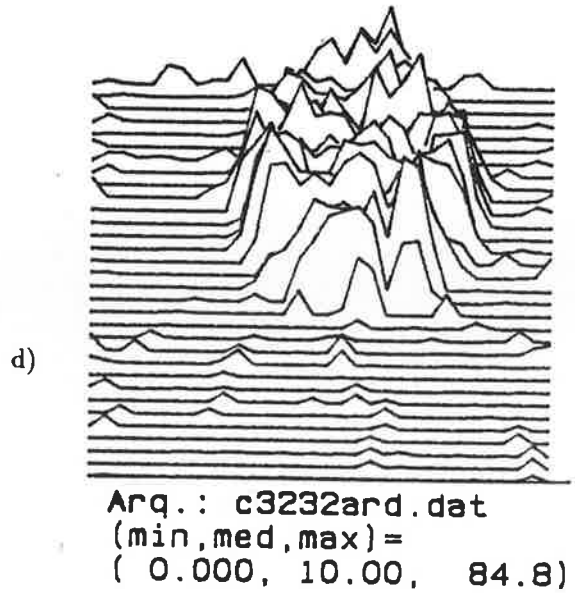
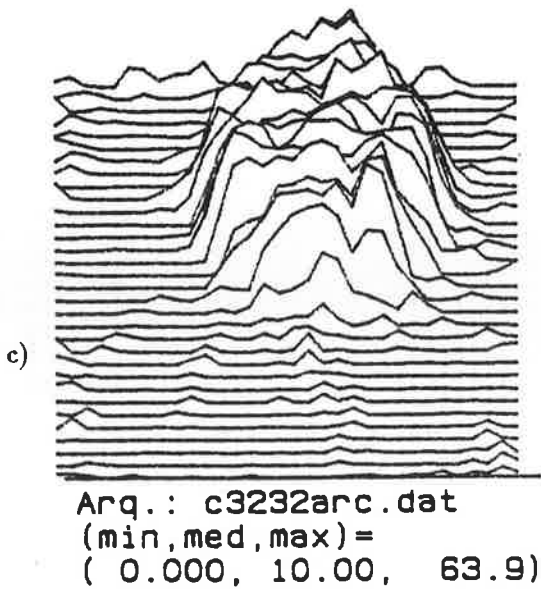
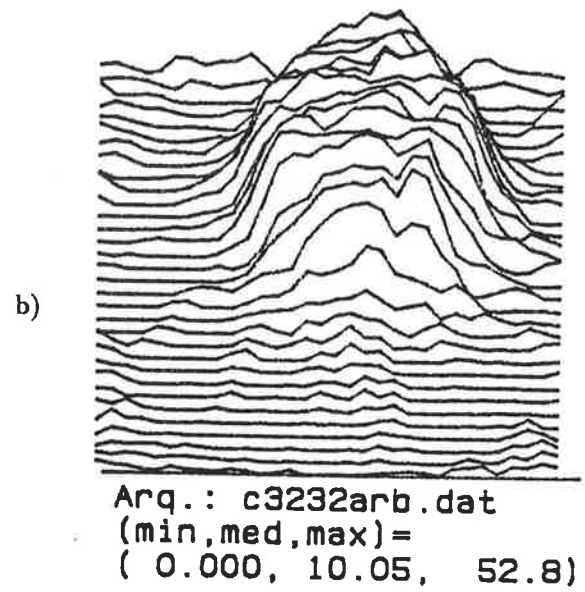
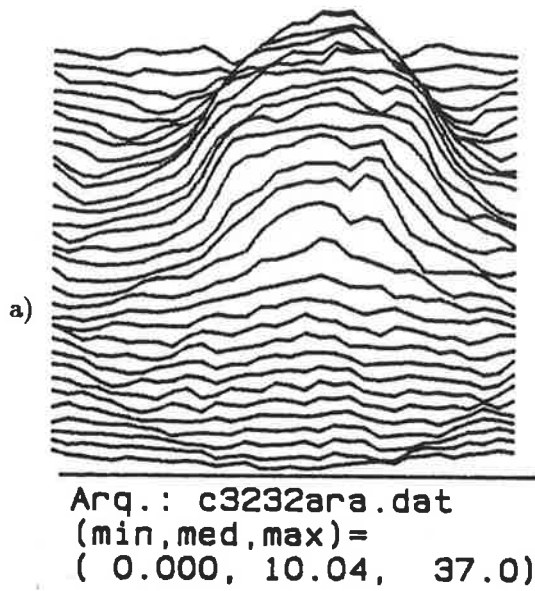


Figura 5.11 Reconstrução utilizando o método ART para projeções

c)10; d)30 iterações

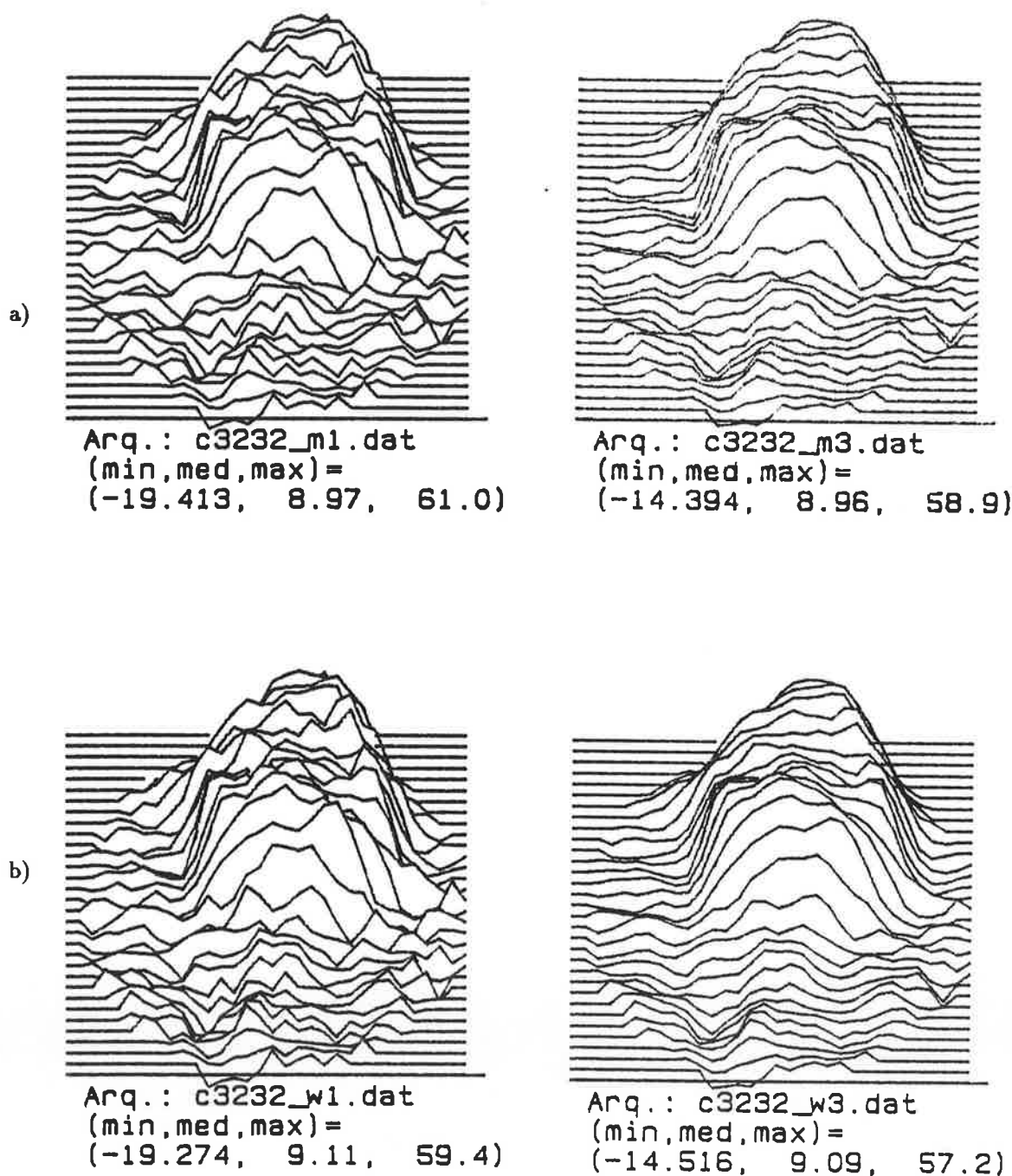


Figura 5.12 Resultado da reconstrução utilizando o método da Aplicação Direta da Transformada de Fourier e estimadores aplicados diretamente sobre as projeções: a) MAP com filtro gaussiano com FWHM=1 e 1.61, e b) Wiener (FWHM=1 e 1.61).

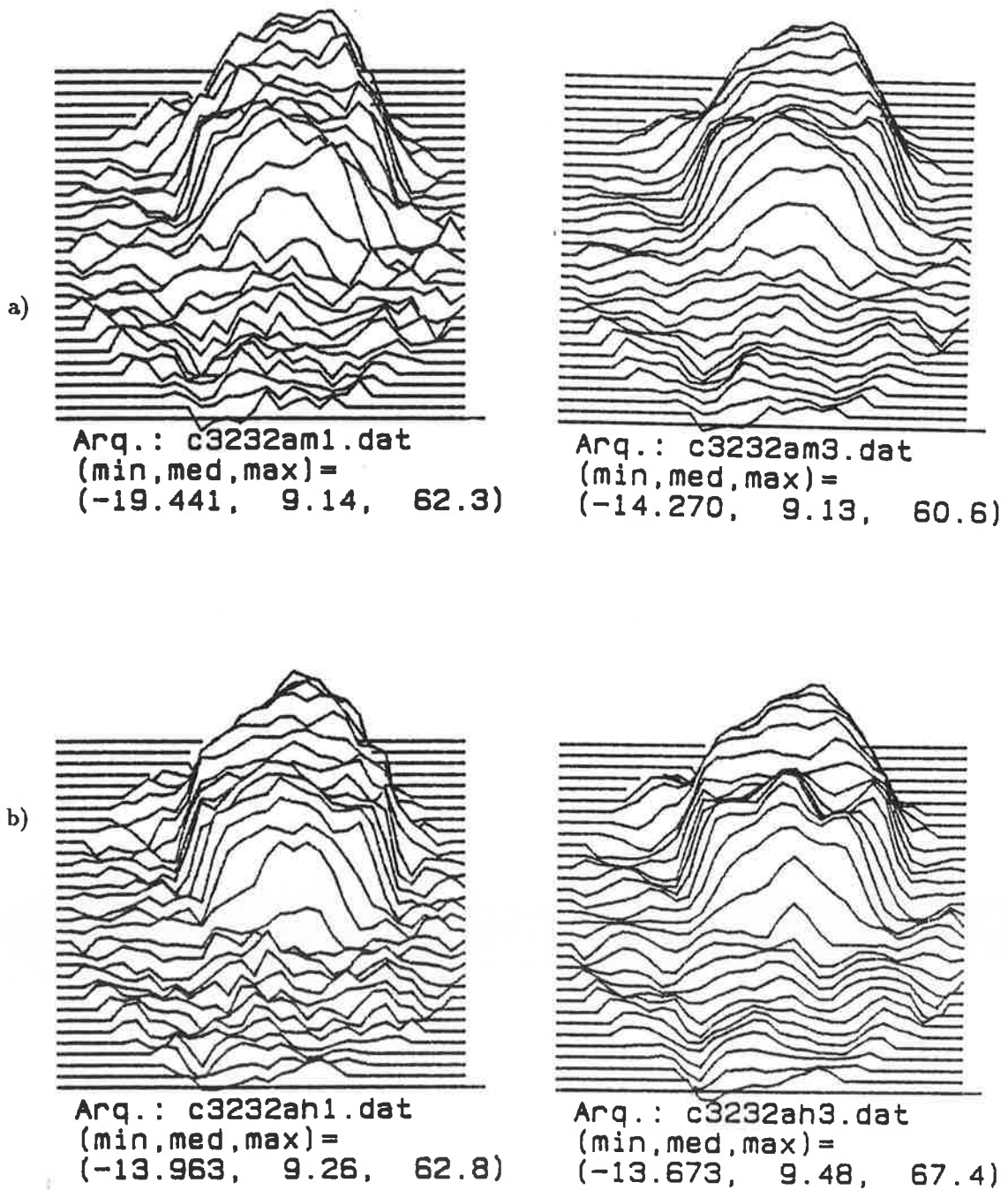


Figura 5.13 Resultado da reconstrução utilizando o método da Aplicação Direta da Transformada de Fourier , e estimadores sobre as projeções já alteradas pela Transformação de Anscombe: a) MAP (FWHM=1.0 e 1.61); b) heurístico (FWHM=1.0 e 1.61).



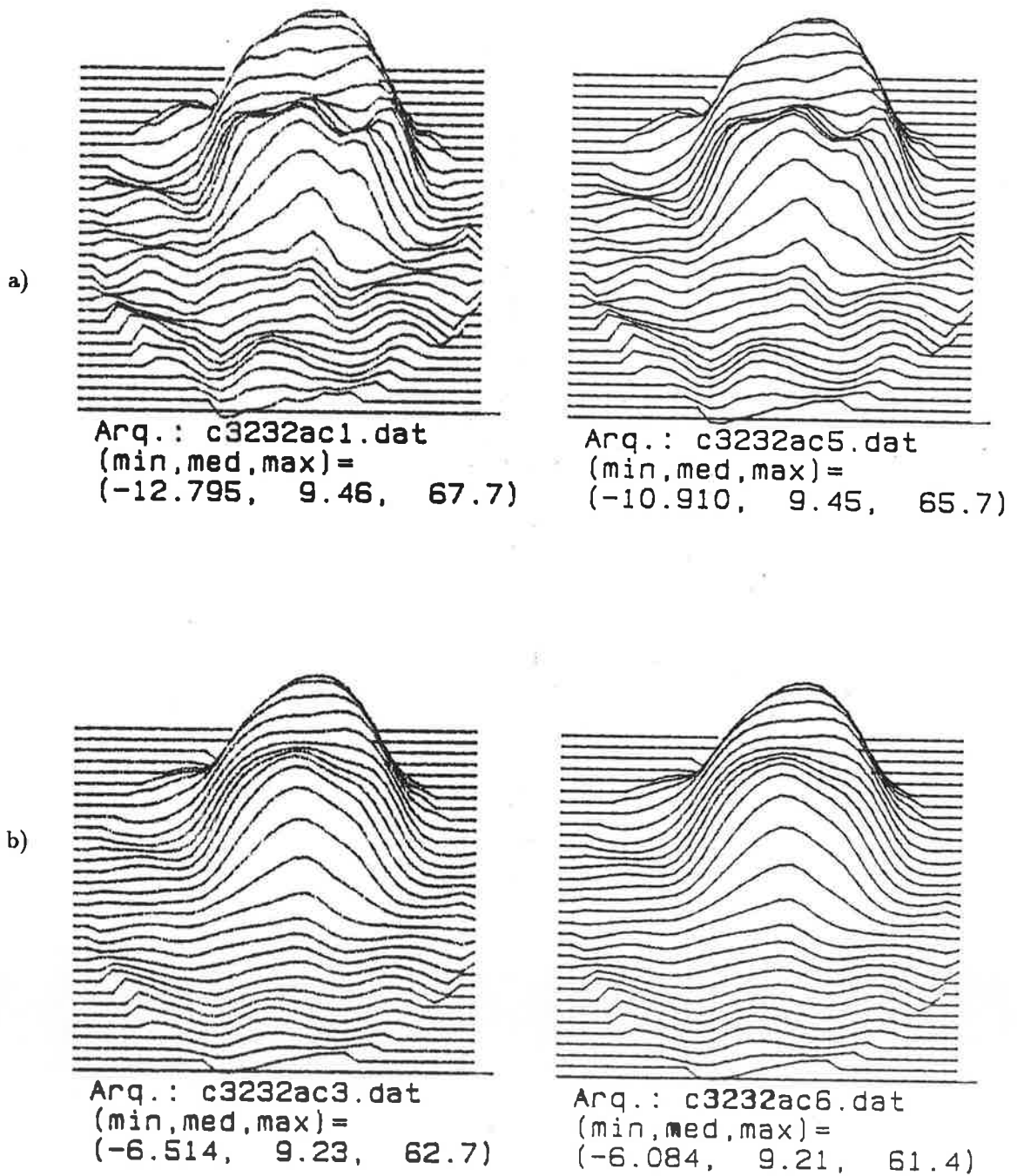


Figura 5.14 Resultado da reconstrução utilizando o método da Aplicação Direta da Transformada de Fourier , e estimadores sobre as projeções já alteradas pela Transformação de Anscombe: a) menor custo com  $\beta=1$  (FWHM=1.0 e 1.61) ; b) menor custo com  $\beta=5.0$  (FWHM=1.0 e 1.61) .

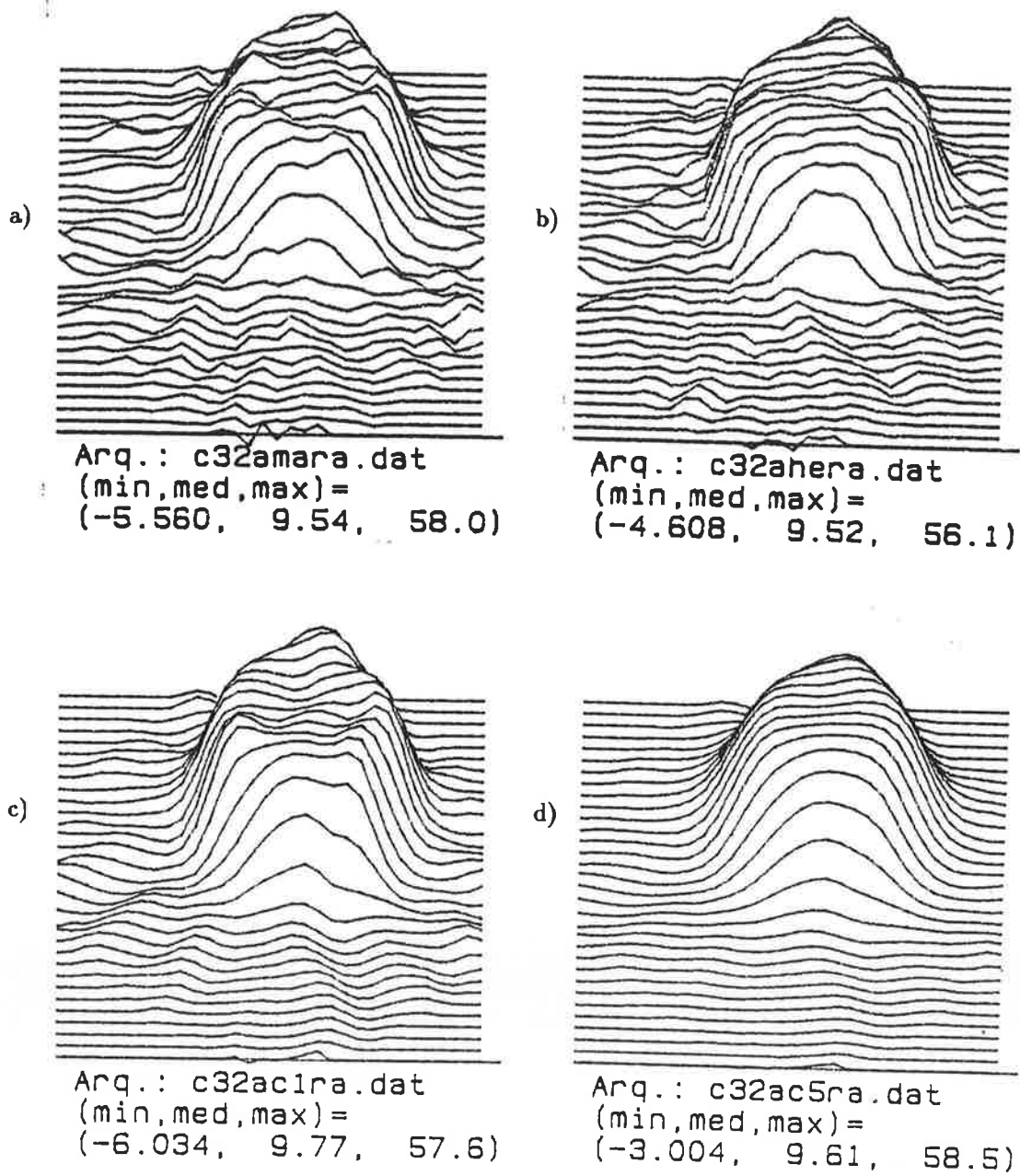


Figura 5.15 Resultado da reconstrução pelo método da Filtragem-Retroprojeção (rampa) sobre projeções já estimadas através da Transformação de Anscombe e : a) estimador MAP ; b) heurístico; c) custo ( $\beta=1$ ); d) custo ( $\beta=5$ ) .

Analisando-se (tabela 5.2) todos os estimadores empregados em conjunto com a técnica de reconstrução DTF, conclui-se que o estimador heurístico (filtro de Maeda) com FWHM=1.0 apresenta melhores resultados (REMQN global=0.2435).

A figura 5.15 mostra o resultado da reconstrução empregando a técnica da Filtragem-Retroprojeção (FRP) com rampa, sobre projeções estimadas através da transformação de Anscombe com estimadores do tipo MAP, heurístico, e custo mínimo para  $\beta=1$  e  $\beta=5$ . A tabela 5.2 confirma as conclusões anteriores relativas à técnica de reconstrução DTF, ou seja, os resultados dos estimadores que minimizam a função custo são próximos aos produzidos pelos estimadores Bayesianos, e o estimador heurístico apresenta o melhor desempenho (REMQN global = 0.1875).

Quantitativamente, as técnicas que produziram os melhores resultados foram o ML-EM a 5 iterações (REMQN global=0.1905) e a Filtragem-Retroprojeção (FRP) associada ao estimador heurístico (filtro de Maeda) sobre as projeções modificadas pela transformação de Anscombe (REMQN global=0.1875). A figura 5.16 mostra, graficamente, o desempenho de alguns métodos, em especial da Filtragem-Retroprojeção simples com rampa, da Filtragem-Retroprojeção com transformação de Anscombe e estimação heurística, do EM a 5 iterações e do ART a 10 iterações. Neste gráfico estão representados os erros (REMQN) na região central, na borda e global, além do tempo de reconstrução. Nota-se que a melhor reconstrução na região central é obtida pelo FRP associado ao estimador. As quatro técnicas apresentam índices semelhantes para a reconstrução nas bordas. O índice global é praticamente o mesmo para a FRP associado ao estimador, ML-EM e ART. No entanto, o tempo de processamento para o FRP é significativamente menor quando comparado a estes dois últimos métodos.

Tabela 5.2 Tabela comparativa de desempenho dos métodos de reconstrução quanto ao erro e tempo de reconstrução, para o caso ruidoso

		REMQN			Tempo p/ log		
		central	bordas	global	reconstrução	likelihood	
		(s)					
DTF	FWHM=1.0	0.3349	0.4559	0.5922	55.03		
CRP	Shepp-Logan	0.2394	0.3582	0.3692	39.38		
FRP	Rampa	0.1383	0.2692	0.2660	73.0		
EM	2 iterações	0.2962	0.4756	0.4211	15.8 min	10179.0	
	5 "	0.0954	0.2504	0.1905	39.5 min	12957.5	
	10 "	0.1651	0.2453	0.2087	79 min	13431.0	
	30 "	0.3417	0.4590	0.4040	237 min	13525.3	
ART	2 iterações	0.3516	0.5056	0.4696	72 min	10679.7	
	5 "	0.1317	0.3346	0.2696	177 min	12537.7	
	10 "	0.1154	0.2557	0.2078	361 min	13057.3	
	30 "	0.2608	0.3325	0.3075	1083 min	13220.5	
DTF	Pois_MAP(FWHM=1.0)	0.1424	0.3048	0.2737	59.93		
	(FWHM=1.61)	0.1190	0.3038	0.2521	59.93		
	Pois_Wiener(FW=1.0)	0.1377	0.3300	0.2868	59.87		
	(FWHM=1.61)	0.1155	0.3266	0.2662	59.87		
	Ansc_MAP (FW=1.0)	0.1268	0.3512	0.3000	61.28		
	(FWHM=1.61)	0.1017	0.3448	0.2763	61.28		
	Ansc_heur.(FW=1.0)	0.1253	0.2697	0.2435	64.65		
	(FWHM=1.61)	0.1685	0.2678	0.2659	64.65		
	Ansc_custo. $\beta=1$ (FW=1.0)	0.1533	0.3196	0.2819	76.07		
	$\beta=1$ (FWHM=1.61)	0.1373	0.3244	0.2748	76.07		
	$\beta=5$ (FWHM=1.0)	0.1219	0.3836	0.2937	76.07		
	$\beta=5$ (FWHM=1.61)	0.1180	0.3893	0.2955	76.07		
	FRP (Rampa)	Pois_MAP	0.0558	0.2674	0.2069	75.0	
		Pois_Wiener	0.0555	0.2971	0.2260	75.0	
		Ansc_MAP	0.0618	0.3097	0.2353	75.0	
		Ansc_Heur.	0.0493	0.2474	0.1875	75.0	
Ansc_custo( $\beta=1$ )		0.0773	0.2806	0.2155	75.0		
Ansc_custo( $\beta=5$ )		0.1179	0.3600	0.2703	75.0		

Conclui-se que, para projeções ruidosas, os métodos da transformada com transformação de Anscombe e estimação das projeções se mostram promissores. Em especial, comparando-se o método EM com a Filtragem-Retroprojeção (rampa) com transformação de Anscombe e estimação heurística, nota-se que este último apresenta índices de erro equivalentes, porém com tempo de reconstrução bastante reduzido em face do EM (figura 5.16).

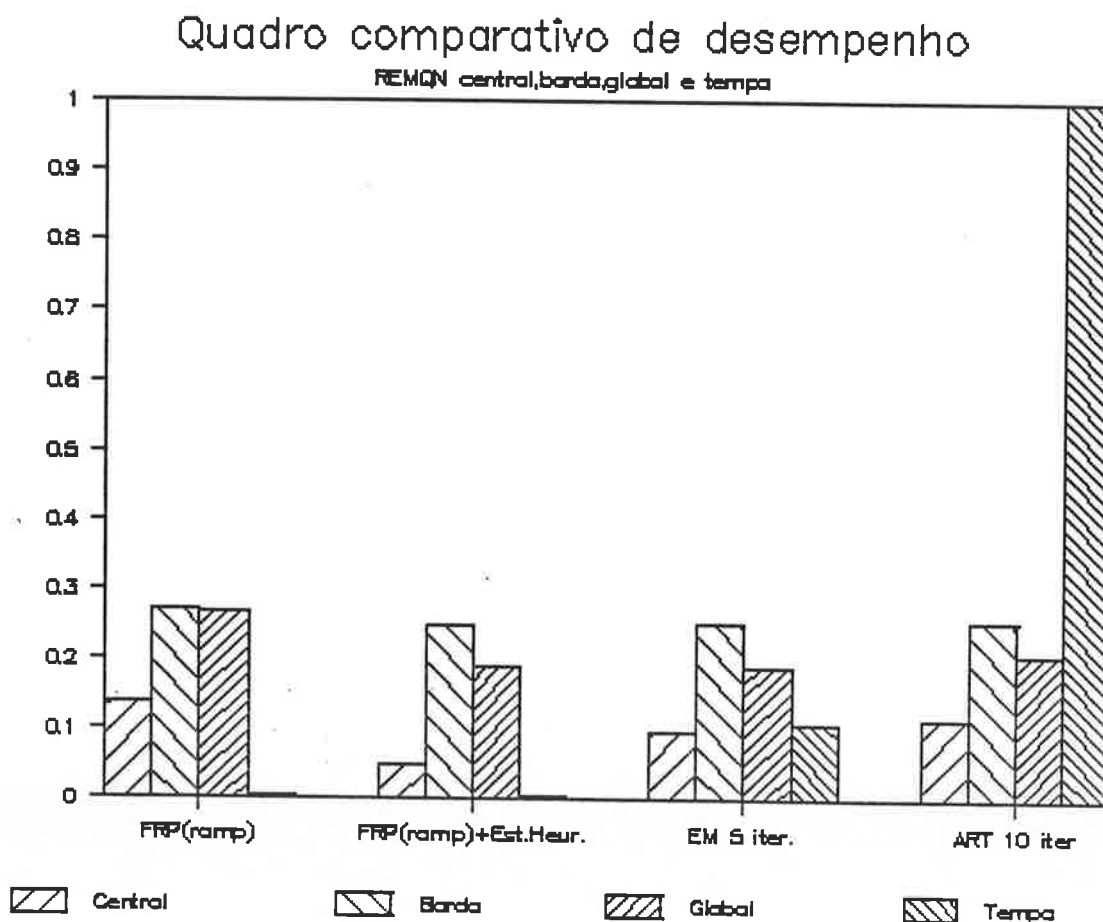


Figura 5.16 Gráfico comparativo do desempenho quanto ao erro (REMQR) central, na borda e global, e ao tempo de reconstrução, conforme tabela 5.2 . Os grupos correspondem a, respectivamente, Filtragem-Retroprojeção com rampa (FRP); Filtragem-retroprojeção após transformação de Anscombe e estimação heurística; EM a 5 iterações; ART a 10 iterações completas.

### 5.5 Reconstrução de estruturas mais complexas na presença de ruído Poisson

Para se avaliar melhor as conclusões da seção anterior, foram geradas as projeções para estruturas mais complexas, com resolução maior. A figura 5.17a mostra o objeto original (54) com 64x64 pixels, sem ruído, com contagem total igual a 200000, e a figura 5.17b mostra o sinograma correspondente, em 100 ângulos de projeção, com 64 amostras por ângulo. A figura 5.17c e 5.17d ilustram a reconstrução, para projeções sem ruído, pelo método das transformadas, respectivamente, pelo DTF com FWHM=1.0 e pela Filtragem-Retroprojeção com rampa. A tabela 5.3 resume as medidas obtidas. Nota-se que o tempo de reconstrução pelo DTF é menor e corresponde a pouco mais de 50% do tempo pela Filtragem-Retroprojeção, para a resolução acima citada.

Para o caso de projeções ruidosas (Poisson), a figura 5.18a mostra a reconstrução empregando-se a Filtragem-Retroprojeção com rampa e a figura 5.18b mostra o resultado utilizando-se este mesmo método após a estimação das projeções pelo estimador heurístico (transformação de Anscombe e filtro de Maeda). Nota-se uma melhoria significativa na reconstrução devido à estimação. As figura 5.18c e d ilustram, respectivamente, os resultados da aplicação do método EM após 5 e 10 iterações. Nota-se, qualitativamente, que os resultados são comparáveis, porém o EM é muito mais lento (Tabela 5.3). O resultado da figura 5.18d corresponde, aproximadamente, ao obtido por Tanaka (54) nas mesmas condições de ruído e resolução, pelo método modificado do EM (Filtered Iterative Reconstruction - FIR algorithm), na terceira iteração (Tanaka, fig.8). A figura 5.19a) mostra o resultado obtido pelo EM a 30 iterações, e as figuras 5.18b, c e d, os resultados empregando-se o ART a 5, 10 e 20 iterações.

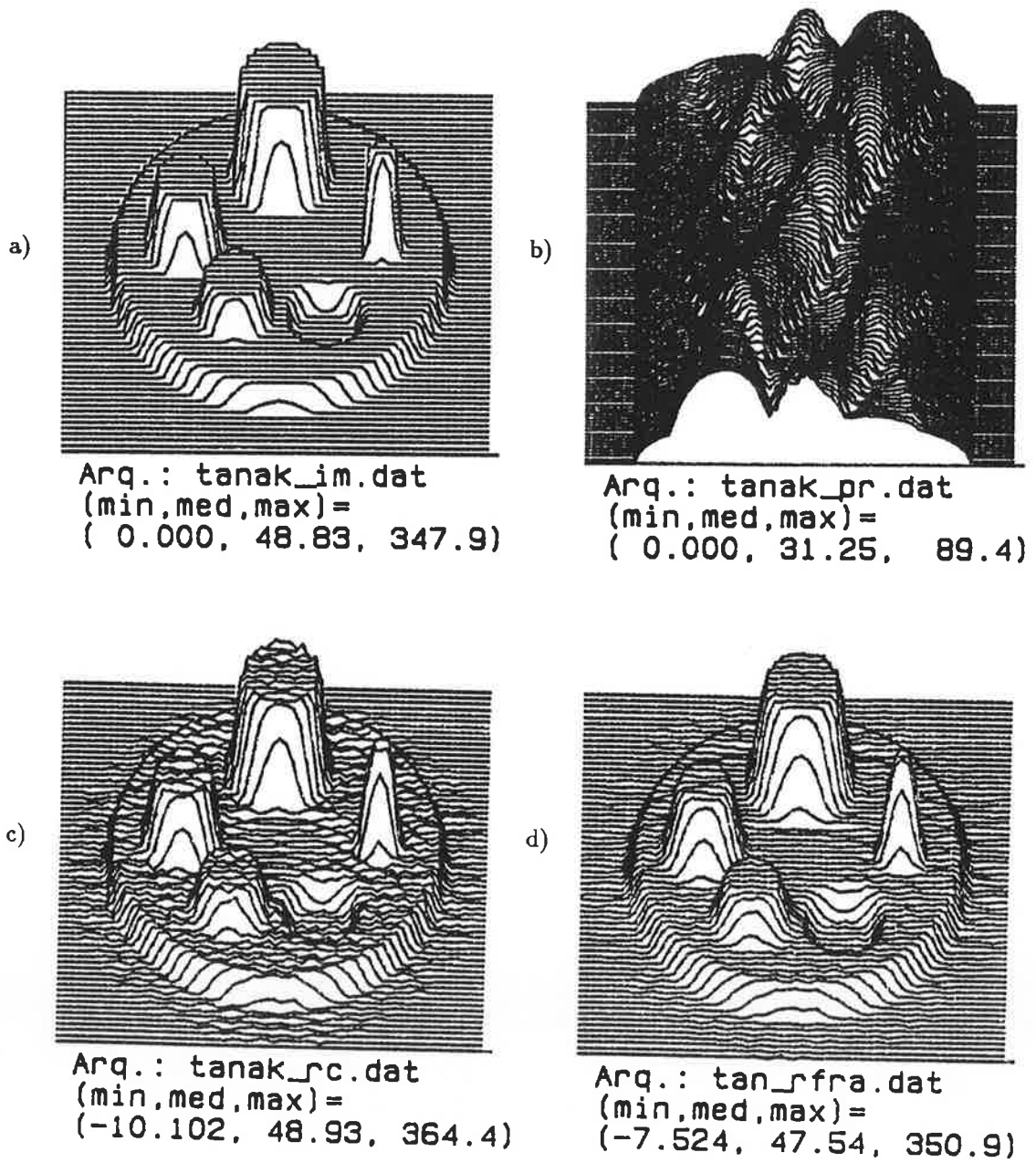


Figura 5.17. Reconstrução para estruturas mais complexas sem ruído. Imagem 64x64, a partir de 100 ângulos de projeção com 64 amostras cada. a) Imagem original; b) sinograma correspondente; c) reconstrução pelo DTF (FWHM=1) ; d) reconstrução pela Filtragem-Retroprojeção (rampa)

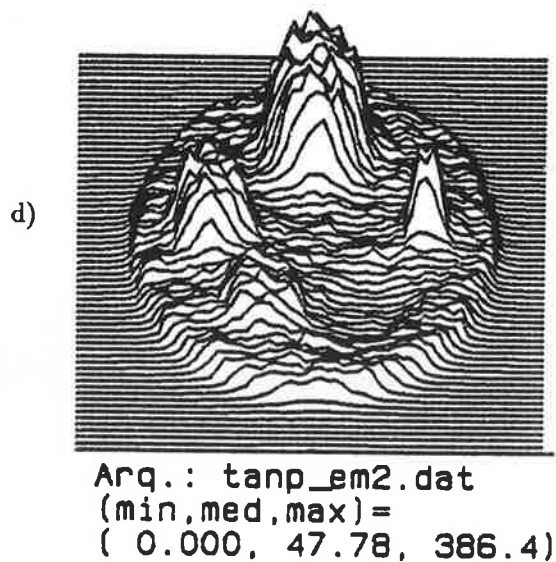
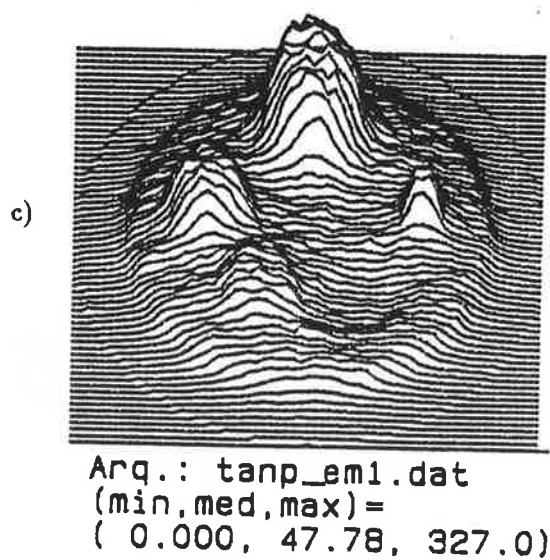
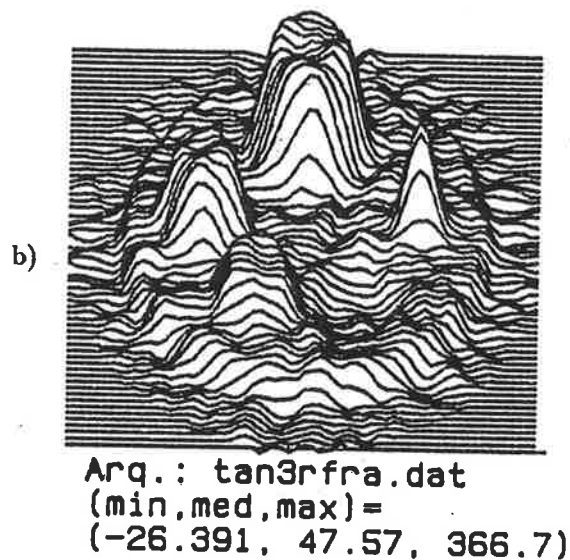
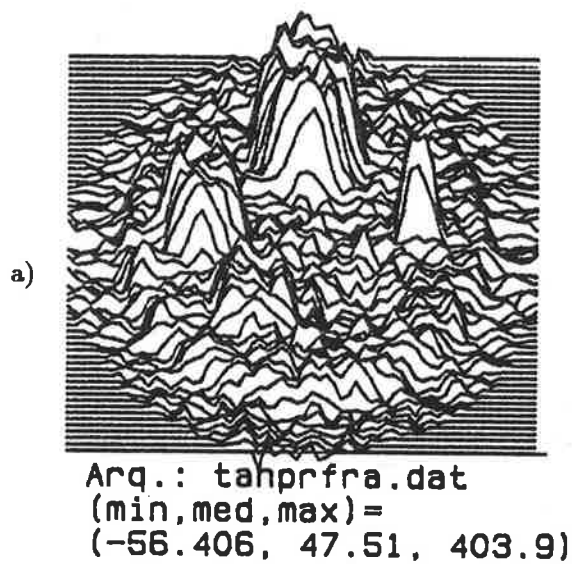
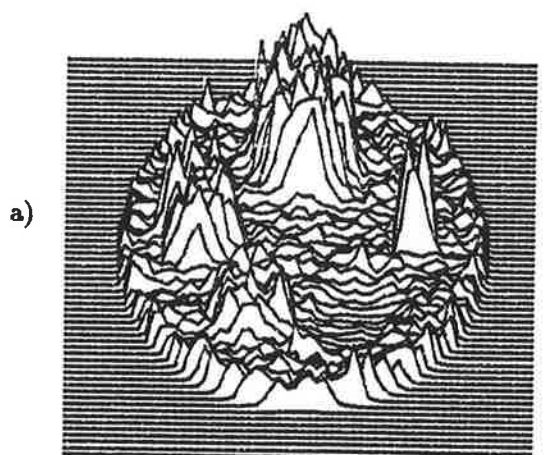
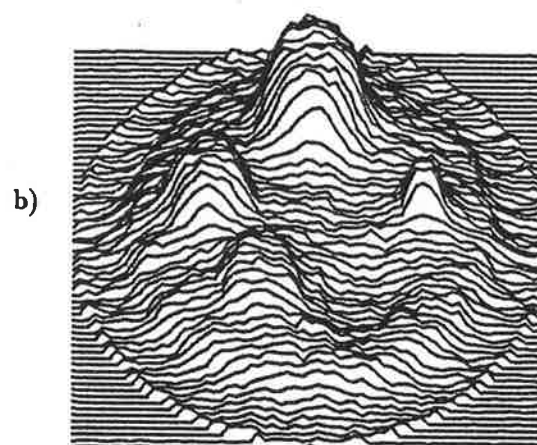


Figura 5.18. Reconstrução para estruturas mais complexas (64x64) com ruído .  
a) Filtragem-Retroprojeção (rampa); b) Filtragem-Retroprojeção com rampa após estimador heurístico ; c) EM a 5 iterações ; d) EM a 10 iterações

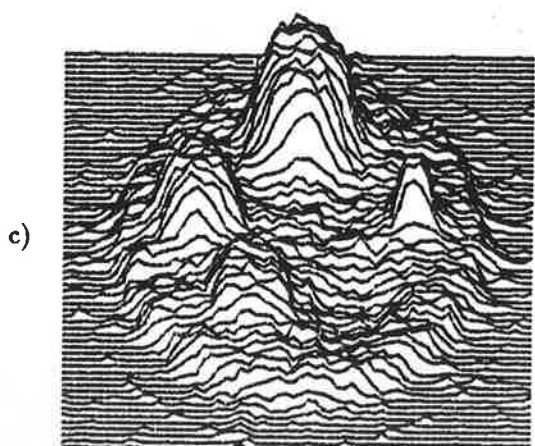




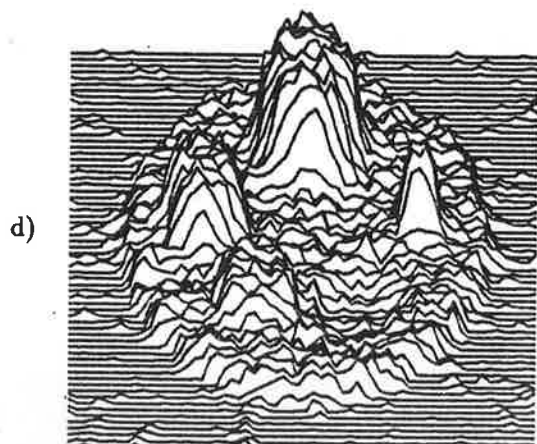
Arq.: tanp\_em3.dat  
(min,med,max)=  
( 0.000, 47.78, 465.0)



Arq.: TANP\_AR4.DAT  
(min,med,max)=  
( 0.000, 48.07, 260.6)



Arq.: tanp\_ar1.dat  
(min,med,max)=  
( 0.000, 48.05, 325.6)



Arq.: tanp\_ar5.dat  
(min,med,max)=  
( 0.000, 48.10, 373.0)

Figura 5.19. Reconstrução para estruturas mais complexas (64x64), com ruído . a) EM a 30 iterações; b) ART a 5 iterações; c) ART 10 iterações ; d) ART a 20 iterações

Tabela 5.3 Valores comparativos da reconstrução de imagem 64x64, a partir de 100 ângulos de projeção, com 64 amostras cada

	mínimo	médio	máximo	contagem total	REMQN global	Tempo p/ reconst(min)
<b>Sem ruído</b>						
Ideal	0	48.8	347.9	200000	0	—
DTF(FWHM=1)	-10.1	48.9	364.4	203077	0.0714	3min19s
Ret.Filt(Rampa)	-7.5	47.5	350.9	199107	0.0982	5min54s
<b>Com ruído</b>						
RFR* s/estim.	-56.4	47.5	403.9	213272	0.2132	5min54s
RFR c/estim.heur	-26.4	47.6	366.7	204092	0.1820	6min25s
EM 5 iter.	0	47.8	327.0	195719	0.2964	240min38s
EM 10 iter.	0	47.8	386.4	195705	0.1885	481min16s
EM 30 iter.	0	47.8	465.0	195705	0.2210	1440min
ART 5 iter.	0	48.1	260.6	196901	0.3634	1740min
ART 10 iter.	0	48.0	325.6	196819	0.2529	3499min
ART 20 iter.	0	48.1	373.0	197010.3	0.1950	6960min

\* RFR: Retroprojeção Filtrada com Rampa

## 5.6 Conclusões

Para a reconstrução tomográfica de imagens sujeitas a ruído de Poisson, onde as projeções possam ser consideradas transformadas de Radon da imagem a ser reconstruída, os métodos da transformada conjugados à estimação das projeções são os mais efetivos, pois apresentam resultados comparáveis ao EM, em uma fração do tempo de processamento requerido pelo mesmo. A tabela 5.2 mostra que, para a reconstrução do cilindro com ruído, a melhor técnica, dentre as estudadas, corresponde ao método da transformada (Filtragem-Retroprojeção com rampa) conjugado à estimação heurística (filtro de Maeda) com o emprego da transformada de Anscombe. Este resultado é praticamente equivalente ao melhor caso do EM (5 iterações), porém muito mais rápido. O desempenho da técnica DTF conjugado à estimação tem desempenho inferior quando comparado com a Filtragem-Retroprojeção, principalmente devido a erros na interpolação no domínio da frequência, agravado com a presença do ruído. No entanto, para imagens com maior número de pixels, o método DTF é mais rápido (tabela 5.3). A tabela 5.3 confirma, para imagens ruidosas mais complexas, a conclusão anterior, i.é, a técnica da Filtragem-Retroprojeção com estimação das projeções produz resultados comparáveis ao do EM, com tempo de processamento consideravelmente menor.

Utilizando-se a mesma técnica de reconstrução (Filtragem-Retroprojeção com rampa), o emprego da estimativa das projeções (estimador heurístico) produz resultados melhores (melhoria adicional de 1.37 dB) quando comparado à reconstrução sem estimar as projeções, com relativamente pequeno custo adicional de processamento (figuras 5.18a e b).

A utilização do estimador que minimiza a função custo (rugosidade), para  $\beta=1$ , produz resultados comparáveis aos outros estimadores ótimos (tabela 5.2, e figuras 5.14, 5.15)

Conclui-se, portanto, que a técnica proposta é uma alternativa viável para a reconstrução de imagens com ruído de Poisson.

## 6-CONCLUSÕES E DISCUSSÕES

O ruído Poisson é o fator limitante na reconstrução de imagens em ECT com baixa relação sinal/ruído. O método EM considera a natureza estocástica das projeções na reconstrução, mas o tempo de processamento limita a sua utilização em aplicações clínicas.

Neste trabalho, mostrou-se que os métodos da transformada aliados ao pré-processamento adequado das projeções têm bom desempenho, na reconstrução de imagens, cujas projeções possam ser consideradas aproximações da transformada de Radon da imagem a ser reconstruída. Especificamente, o emprego da transformação de Anscombe e a estimativa das projeções pelo método heurístico, antes da reconstrução pela Filtragem-Retroprojeção (rampa), apresenta resultados comparáveis ao do EM, porém com tempo de processamento bem menor. Dentre os métodos da transformada, procurou-se resgatar o DTF, pois este apresenta desempenho, em termos de velocidade de reconstrução, muito bom para imagens de elevadas dimensões. Este método torna-se ainda mais adequado com a utilização de "array processors", pois grande parte das operações envolvidas no DTF são operações entre vetores.

Foi apresentado um estimador que minimiza a função custo, no caso , a rugosidade, com desempenho comparável aos outros estimadores ótimos (tabela 5.2). Este estimador possui a vantagem adicional de poder incorporar, parcialmente, informações a priori disponíveis sobre a rugosidade das projeções no parâmetro do filtro.

As projeções podem ser estimadas, facilmente, utilizando filtros de Wiener invariantes no espaço, pois no modelo de ruído Poisson, a densidade espectral de potência do sinal original tem relação com a do sinal observado e a contagem total (16,40). No entanto, é uma abordagem global. Vários dos estimadores discutidos neste trabalho são estimadores locais (seções 4.2 e 4.3), respeitando, parcialmente, estruturas locais, e permitindo que cada ponto da projeção seja processado em separado, sem esperar pelos resultados dos seus vizinhos. Esta característica permite processamento em paralelo, agilizando a estimativa das projeções. Conforme mostrado na seção 3.2, os estimadores podem ser associados a outras técnicas de reconstrução, além dos métodos da transformada, permitindo correções (deconvoluções) em relação à resposta dos detetores-colimadores e ao espalhamento.

O presente trabalho limitou-se à reconstrução bi-dimensional de imagens com ruído Poisson, cujas projeções possam ser descritas pelo modelo da seção 3.2. Os resultados podem ser diretamente estendidos à reconstrução tri-dimensional. O SPECT, em especial, representa um grande problema, principalmente devido ao desconhecimento do coeficiente de atenuação, que certamente não é constante em estruturas médicas. Métodos alternativos, porém com demanda acentuada de processamento, têm sido propostos, onde se estimam conjuntamente a atenuação e a intensidade na reconstrução (42), ou mais recentemente, utilizando a técnica iterativa "simulated annealing" (61) com a informação sobre a anatomia do objeto a ser reconstruído, obtida, por exemplo, com raio-X.

Uma alternativa interessante para agilizar todo o processo de reconstrução de imagens ruidosas, é utilizar métodos recursivos (38), pois uma vez disponíveis as projeções para cada ângulo de projeção, as mesmas podem ser estimadas e utilizadas para a reconstrução. Esta abordagem permite ainda que o operador controle a aquisição dos dados até a obtenção da qualidade desejada, mas pressupõe processamento em paralelo com a aquisição e a estimação, além da capacidade de manipular matrizes de elevadas dimensões.

No método DTF, torna-se ainda necessário melhorar o desempenho em relação ao ruído de baixa frequência presente em toda a imagem reconstruída. Um fator de expansão igual a 8 deve minorar este efeito.

Outro ponto a ser pesquisado são os estimadores não-lineares de sinais com ruído Poisson. No presente trabalho mostrou-se que o estimador heurístico (filtro de Maeda) apresenta melhor desempenho do que vários estimadores ótimos, principalmente devido ao filtro não-linear envolvido para obter a mediana.

APÊNDICES

A) Probabilidade na detecção de fótons de um emissor que segue processo Poisson

Seja X: variável aleatória com distribuição de Poisson associada ao emissor com taxa de emissão média f

$$\text{Logo, } \text{Prob}[X=n] = \frac{e^{-f} (f)^n}{n!} \quad (\text{A.1})$$

Y: v.a. associada ao número de fótons detetados por um sensor

h: a probabilidade de que um fóton efetivamente emitido seja detetado pelo sensor. Este fator pode englobar o efeito de fenômenos como espalhamento, absorção, resposta do sensor, etc. Em um caso mais simples (emissão homogênea) pode ser  $h = \frac{\delta}{4\pi}$ , onde  $\delta$  é o ângulo sólido que envolve o sensor visto do emissor.

Portanto, dado que foram emitidos n fótons, a probabilidade de que sejam detetados m é (18):

$$\text{Prob}[Y=m | X=n] = \binom{n}{m} \cdot h^m \cdot (1-h)^{n-m} \quad m \leq n \quad (\text{distribuição Binomial}) \quad (\text{A.2})$$

Como

$$\text{Prob}[Y=m] = \sum_{n=m}^{\infty} \text{Prob}[Y=m | X=n] \cdot \text{Prob}[X=n] \quad (\text{A.3})$$

$$\begin{aligned} &= \sum_{n=m}^{\infty} \binom{n}{m} \cdot h^m \cdot (1-h)^{n-m} \cdot \frac{e^{-f} (f)^n}{n!} \\ &= e^{-f} \cdot \sum_{n=m}^{\infty} \frac{n!}{m! (n-m)!} h^m (1-h)^{n-m} \cdot \frac{f^n}{n!} \end{aligned} \quad (\text{A.4})$$

$$= \frac{e^{-f} h^m}{m!} \cdot f^m \cdot \sum_{i=0}^{\infty} \frac{(1-h)^i f^i}{i!}$$

como esta última somatória é a expansão em série de  $e^{(1-h)f}$ , tem-se

$$\text{Prob}[Y=m] = \frac{e^{-hf} (hf)^m}{m!} \quad \text{o qual corresponde à distribuição de Poisson com média hf} \quad (\text{A.5})$$

**B. Algumas propriedades do operador E[.] em variáveis aleatórias multivariadas**

Inicialmente considere-se apenas uma variável aleatória X com função densidade de probabilidade f(x) para o caso contínuo, e probabilidade p(x) para o caso discreto. Defina-se o valor esperado de X (25):

$$E[X] = \int_{-\infty}^{\infty} x \cdot f(x) \cdot dx \quad (\text{v.a. contínua}) \quad (\text{A.6})$$

$$\sum_{\text{todo } x} x \cdot p(x) \quad (\text{v.a. discreta})$$

Este operador é linear, i.é, sendo  $\alpha$  e  $\beta$  duas constantes,  $g(X)$  e  $h(X)$  funções de X, tem-se:

$$E[\alpha \cdot g(X) + \beta \cdot h(X)] = \alpha \cdot E[g(X)] + \beta \cdot E[h(X)] \quad (\text{A.7})$$

Considerando-se agora um vetor de v.a. (multivariado)  $\vec{X} = [X_1 \ X_2 \ \dots \ X_N]^T$ , define-se a função distribuição de probabilidade conjunta (44) por:

$$F_{X_1 X_2 \dots X_N}(x_1, x_2, \dots, x_N) = \text{Prob}[X_1 \leq x_1, X_2 \leq x_2, \dots, X_N \leq x_N] \quad (\text{A.8})$$

A função densidade de probabilidade conjunta (para o caso contínuo):

$$f_{\vec{X}}(\vec{x}) = f_{X_1 X_2 \dots X_N}(x_1, x_2, \dots, x_N) = \frac{\partial^N F_{X_1 X_2 \dots X_N}(x_1, x_2, \dots, x_N)}{\partial x_1 \partial x_2 \dots \partial x_N} \quad (\text{A.9})$$

A função densidade de probabilidade marginal de  $X_i$ ,

$$f_{X_i}(x_i) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} f_{\vec{X}}(\vec{x}) \cdot dx_1 \cdot dx_2 \dots dx_{i-1} \cdot dx_{i+1} \dots dx_N \quad (\text{A.10})$$

Neste contexto,

$$E[X_i] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} x_i \cdot f_{\vec{X}}(\vec{x}) \cdot dx_1 \cdot dx_2 \dots dx_N$$

$$= \int_{-\infty}^{\infty} x_i \cdot dx_i \cdot \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} f_{\vec{X}}(\vec{x}) \cdot dx_1 \cdot dx_2 \dots dx_{i-1} \cdot dx_{i+1} \dots dx_N$$

$$= \int_{-\infty}^{\infty} x_i \cdot f_{X_i}(x_i) \cdot dx_i \quad (\text{A.12})$$

onde ressalte-se que  $f_{X_i}(x_i)$  é a função densidade de probabilidade (f.d.p.) marginal de  $X_i$  e não a f.d.p. de  $X_i$ .

Desse modo, pode-se definir o valor esperado do vetor  $\vec{X}$  por:

$$E[\vec{X}] = E[X_1 \ X_2 \ \dots \ X_N]^T = [E(X_1) \ E(X_2) \ \dots \ E(X_N)]^T \tag{A.13}$$

onde  $E[X_i]$  é dado pela equação acima( A.12).

Considerando-se duas funções  $g(\vec{X})$  e  $h(\vec{X})$ , onde  $g : \mathbb{R}^N \rightarrow \mathbb{R}$  e  $h : \mathbb{R}^N \rightarrow \mathbb{R}$  tem-se,

$$E[g(\vec{X})] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} g(x_1, x_2, \dots, x_N) \cdot f_{\vec{X}}(\vec{x}) \cdot dx_1 \cdot dx_2 \dots dx_N \tag{A.14}$$

pela linearidade, onde  $\alpha$  e  $\beta$  constantes,

$$E[\alpha \cdot g(\vec{X}) + \beta \cdot h(\vec{X})] = \alpha \cdot E[g(\vec{X})] + \beta \cdot E[h(\vec{X})] \tag{A.15}$$

### B.1 Condicional

Por facilidade de notação será apenas desenvolvido para o caso de 2 v.a's. Contudo, pode ser diretamente estendido para multivariado geral. Define-se a função densidade de probabilidade condicional de  $X_1$ , dado  $X_2 = x_2$  por (44):

$$f_{X_1|X_2=x_2}(x_1) = \frac{f_{X_1 X_2}(x_1, x_2)}{\int_{-\infty}^{\infty} f_{X_1 X_2}(x_1, x_2) \, dx_1} = \frac{f_{X_1 X_2}(x_1, x_2)}{f_{X_2}(x_2)} \tag{A.16}$$

onde  $f_{X_2}(x_2)$  é f.d.p. marginal de  $X_2$

As v.a.  $X_1$  e  $X_2$  são independentes quando (25)

$$f_{X_1 X_2}(x_1, x_2) = f_{X_1}(x_1) \cdot f_{X_2}(x_2) \tag{A.17}$$

onde  $f_{X_1}(x_1)$  e  $f_{X_2}(x_2)$  são f.d.p marginais de  $X_1$  e  $X_2$  respectivamente.

Portanto, sendo independentes,

$$f_{X_1|X_2=x_2}(x_1) = f_{X_1}(x_1) \quad \text{ou seja} \tag{A.18}$$

a f.d.p. condicional, marginal e isolada coincidem.

O valor esperado condicional,

$$E_{X_1}[X_1 | X_2 = x_2] = \int_{-\infty}^{\infty} x_1 \cdot f_{X_1|X_2=x_2}(x_1) \cdot dx_1 = \text{função}(x_2) \tag{A.19}$$



Substituindo-se  $x_2$  na função acima (equação A.19) pela v.a  $X_2$ , tem-se a v.a

$$E_{X_1}[X_1 | X_2] = \text{função}(X_2) \tag{A.20}$$

Desse modo,

$$E[X_1] = E_{X_2}\{E_{X_1}[X_1|X_2]\} \quad \text{pois} \tag{A.21}$$

$$\begin{aligned} E[X_1] &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x_1 \cdot f_{X_1 X_2}(x_1, x_2) dx_1 \cdot dx_2 \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x_1 \cdot f_{X_1|X_2=x_2}(x_1) \cdot f_{X_2}(x_2) \cdot dx_1 \cdot dx_2 \quad (\text{da eq. A.16}) \\ &= \int_{-\infty}^{\infty} f_{X_2}(x_2) \cdot dx_2 \int_{-\infty}^{\infty} x_1 \cdot f_{X_1|X_2=x_2}(x_1) \cdot dx_1 \\ &= \int_{-\infty}^{\infty} f_{X_2}(x_2) \cdot E_{X_1}[X_1 | X_2=x_2] \cdot dx_2 \\ &= E_{X_2}\{E_{X_1}[X_1|X_2]\} \end{aligned}$$

Sendo  $g : \mathbb{R} \rightarrow \mathbb{R}$  e  $h : \mathbb{R} \rightarrow \mathbb{R}$  duas funções, tem-se

$$E_{X_1}[g(X_1) | X_2=x_2] = \int_{-\infty}^{\infty} g(x_1) \cdot f_{X_1|X_2=x_2}(x_1) \cdot dx_1 \tag{A.22}$$

$$E[g(X_1)] = E_{X_2}\{E_{X_1}[g(X_1)|X_2]\} \tag{A.23}$$

$$E_{X_1}\{E_{X_2}[g(X_1) \cdot h(X_2) | X_1=x_1]\} = E_{X_1}\{g(X_1) \cdot E_{X_2}[h(X_2)|X_1=x_1]\} \tag{A.24}$$

Por outro lado, se  $g : \mathbb{R}^2 \rightarrow \mathbb{R}$  tem-se:

$$E[g(X_1, X_2)] = E_{X_1}\{E_{X_2}[g(X_1, X_2)|X_1]\} \tag{A.25}$$

Das equações A.22 e A.25,

$$\text{var}(X_1|X_2=x_2) = E_{X_1}[(X_1 - E(X_1|X_2=x_2))^2 | X_2=x_2] \tag{A.26}$$

$$\text{var}(X_1|X_2) = E_{X_1}[(X_1 - E(X_1|X_2))^2 | X_2]$$

Sendo  $\vec{a}$  : um vetor de constantes  $N \times 1$ ,

$$E[\vec{a}' \cdot \vec{X}] = \vec{a}' \cdot E[\vec{X}] \quad ( '=transposta) \tag{A.27}$$

$$\text{var}(\vec{a}' \cdot \vec{X}) = \vec{a}' \cdot \text{cov}(\vec{X}) \cdot \vec{a} \quad (\text{A.28})$$

$$\text{cov}(\vec{X}) = E[(\vec{X} - E(\vec{X})) \cdot (\vec{X} - E(\vec{X}))'] \quad (\text{A.29})$$

Se A: matriz M x N de constantes, e

$$\vec{Y} = A \cdot \vec{X} \quad (\text{A.30})$$

$$E[\vec{Y}] = A \cdot E[\vec{X}] \quad (\text{A.31})$$

$$\text{cov}(\vec{Y}) = A \cdot \text{cov}(\vec{X}) \cdot A' \quad (\text{A.32})$$

## B.2 Multivariado Independente

$$E[\vec{X}] = [E(X_1) \ E(X_2) \ \dots \ E(X_N)]' \quad (\text{A.33})$$

onde  $E(X_i)$  é o valor esperado considerando-se cada v.a isoladamente (independente)

$$f_{X_1 X_2 \dots X_N}(x_1, x_2, \dots, x_N) = f_{X_1}(x_1) \cdot \dots \cdot f_{X_N}(x_N) \quad (\text{A.34})$$

### C. Operador Laplaciano 1-dimensional

Seja o sinal contínuo  $x(t)$ . O operador Laplaciano unidimensional em  $t$  é definido por:

$$L(x(t)) = \frac{\partial^2 x(t)}{\partial t^2} = \frac{\partial}{\partial t} \left( \frac{\partial x(t)}{\partial t} \right) \quad (\text{A.35})$$

De forma análoga pode-se definir o Laplaciano para sinais discretos  $s(i)$ ,  $i=1, M$ , no ponto  $i$

por:

$$(L\vec{s})_i = (D\vec{s})_{i+\frac{1}{2}} - (D\vec{s})_{i-\frac{1}{2}} \quad \text{onde } D: \text{ operador derivada} \quad (\text{A.36})$$

$$(D\vec{s})_j = s_{j+\frac{1}{2}} - s_{j-\frac{1}{2}} \quad (\text{A.37})$$

$$(L\vec{s})_i = (s_{i+1} - s_i) - (s_i - s_{i-1}) = s_{i+1} - 2s_i + s_{i-1} \quad (\text{A.38})$$

D. Estimação por Máxima Verossimilhança em ECT e sua relação com Pseudo inversa

Considerando o modelo da figura 3.4, e considerando a independência das projeções (variáveis aleatórias Poisson), podemos maximizar a equação abaixo em relação a  $\vec{f}$ :

$$\max_{\vec{f}} \text{Prob}[\vec{Y} = \vec{y} \mid \vec{f}]$$

$$\vec{g} = H \cdot \vec{f}$$

Seja  $\vec{h}_i$ ' a i-ésima linha de 'H', portanto

$$g_i = \vec{h}_i' \cdot \vec{f} \quad (\text{onde ' corresponde a transposta}) \quad (\text{A.39})$$

Como a função logarítmica é monotônica, podemos maximizar a função  $L(\vec{f})$ ,

$$L(\vec{f}) = \ln(\text{Prob}[\vec{Y} = \vec{y} \mid \vec{f}]) = \sum_{i=1}^m (-g_i + y_i \ln g_i - \ln(y_i!)) \quad (\text{A.40})$$

$$\partial L(\vec{f}) / \partial \vec{f} = \sum_{i=1}^m [-\partial g_i / \partial \vec{f} + y_i \cdot \partial \ln g_i / \partial \vec{f}] \quad (\text{A.41})$$

$$= \sum [-\partial(\vec{h}_i' \cdot \vec{f}) / \partial \vec{f} + y_i \cdot \partial \ln(\vec{h}_i' \cdot \vec{f}) / \partial \vec{f}] \quad (\text{A.42})$$

$$= \sum [-\vec{h}_i + y_i \cdot \vec{h}_i / (\vec{h}_i' \cdot \vec{f})]$$

$$= \sum [(y_i / (\vec{h}_i' \cdot \vec{f}) - 1) \cdot \vec{h}_i] \quad (\text{A.43})$$

$$\text{Portanto } \sum_{i=1}^m [(y_i / (\vec{h}_i' \cdot \vec{f}) - 1) \cdot \vec{h}_i] = 0 \quad (\text{A.44})$$

Uma das soluções da equação anterior (47), é:

$$y_i / (\vec{h}_i' \cdot \vec{f}) - 1 = 0 \quad i=1, m$$

Donde,

$$y_i = \vec{h}_i' \cdot \vec{f} \quad (\text{A.45})$$

$$\text{ou seja } \vec{y} = H \cdot \vec{f} \quad (\text{A.46})$$

Portanto a solução ML em ECT é nada mais que a clássica obtenção da pseudo inversa

$$\vec{f} = H^+ \cdot \vec{y} \quad (\text{A.47})$$

E. Listagem de parte dos programas desenvolvidos

Os programas para as simulações e avaliações do presente trabalho foram desenvolvidos em FORTRAN 77 e C, com total de, aproximadamente, 6600 linhas de código (200 kBytes). O FORTRAN foi utilizado para manter compatibilidade com as rotinas já desenvolvidas anteriormente, e para facilitar a interconexão com subrotinas disponíveis (21).

A parte desenvolvida em FORTRAN ( ~2850 linhas, 90kBytes) compreende:

- a) geração da imagem original, simulação do ruído Poisson, e adaptação nas rotinas para geração das projeções (21) (PHANLSF.FOR, 317 linhas, 12 kBytes);
- b) estimação das projeções, medida da qualidade e reconstrução pela técnica DTF (aplicação Direta da Transformada de Fourier) (RC\_FT.FOR, 694 linhas, 22 kBytes);
- c) adaptação nas rotinas(21) de reconstrução pelo método da Filtragem-Retroprojeção (XBKFILSF.FOR, 211 linhas, 13 kBytes) e Convolução-Retroprojeção (IMAGFS.FOR, 80 linhas, 2.2 kBytes) ;
- d) conjunto de subrotinas comuns, incluindo filtros, estimadores, interação com usuário, acesso a arquivos, etc (1551 linhas, 41 kBytes);

Na linguagem C foram desenvolvidos, aproximadamente, 3800 linhas de código (109 kBytes), compreendendo:

- a) reconstrução pelos métodos iterativos ART, ML-EM (TOMOSF.C, 1313 linhas, 45 kBytes);
- b) visualização dos resultados em monitor de alta resolução, e saída gráfica em papel (MATRDISP.C, 525 linhas, 12 kBytes);
- c) conjunto de subrotinas comuns, incluindo acesso a arquivos e a dispositivos gráficos, e interação com usuário, (1985 linhas, 52 kBytes);

Em face do número total de linhas , serão listados apenas os módulos principais, a saber, RC\_FT.FOR e TOMOSF.C, excluindo-se dos mesmos as rotinas de menor interesse, tais como interação com o usuário, e acesso aos periféricos.

```

C      file :RC_CTE.FOR
C
C      CONSTANTS for subunits of 'RC_FT'
C
C      PARAMETER (NL=32,NA=64,NPAD=4,NX=32,NY=32)
C      PARAMETER (NL=64,NA=100,NPAD=4,NX=64,NY=64)
C      PARAMETER (PI=3.141593, EPS=1.0E-10)

C(   file: RC_FT.FOR

      PROGRAM RC_FT

C
C      Filter=gauss, Interpolation=angular-linear
C
C      *** Furuie S.S      INCOR/NIRS - sep/88
C      TOMITANI T.      NIRS

C      Reconstruction using FOURIER direct method
C      - 1D -FFT of each projection
C      - interpolation on freq. domain
C      - filtering
C      - 2D-IFFT
C      * ASSUMED:
C      -NL,NPAD,NX,NY as 2**_
C      -reconstruction involves NA views in PI
C      -proj(i,j) -> proj.'j' at detector 'i'
C      -ang(j)=PI/NA*(j-1) crescent as trigonometric conventions,but starting at
C      Y-axis (see RECLBL)
C      -center of image projects into bin (NL/2)+1
C      -image(i,1) = upper left pixel
C
C      LIB files: RC_SUB, NOISE_BLUR, INQ, INDATA, OTDATA, MSGTIM
C-----
C
C      parameters definition
C
C      $INCLUDE:'RC_CTE.FOR'

C      .....local declarations
C      CHARACTER INFILE*30, OTFILE*30,ARQINF*30, OBS*80
C      CHARACTER TP_SOURCE(3)*20,TP_PROC(0:2)*20, tp_noise(0:12)*20,
C      1      TP_FILTER(5)*20,TP_INTERP(3)*20
C      INTEGER*2 HD(NX)
C      $LARGE: CR_TEMP,CI_TEMP,CR_FREQ,CI_FREQ
C      REAL*4 CR_TEMP[HUGE],CI_TEMP[HUGE]
C      REAL*4 CR_TX[HUGE],CI_TX[HUGE] !p/ FFTS2 (apenas notacional)
C      REAL*4 PROJ[HUGE],CR_FREQ[HUGE],CI_FREQ[HUGE]
C      DIMENSION CR_TEMP(NL*NPAD,NA+1),CI_TEMP(NL*NPAD,NA+1)
C      DIMENSION CR_TX(NX,NY),CI_TX(NX,NY) !p/ FFTS2 (apenas notacional)
C      DIMENSION PROJ(NL,NA),CR_FREQ(NX,NY),CI_FREQ(NX,NY)
C      REAL*4 CR_XIN(NL*NPAD),CI_XIN(NL*NPAD),
C      &      CR_XOUT(NL*NPAD),CI_XOUT(NL*NPAD)
C
C      REAL*4 WST(NL*NPAD),WCT(NL*NPAD),WLBR(NL*NPAD),
C      &      W2ST(NX+NY),W2CT(NX+NY),W2LBR(NX+NY)
C
C      EQUIVALENCE (CR_TEMP(1,1),CR_TX(1,1)),(CI_TEMP(1,1),CI_TX(1,1)) !
C      EQUIVALENCE (PROJ(1,1),CR_FREQ(1,1)) !P/ economizar mem.
C      EQUIVALENCE (WST(1),W2ST(1)),(WCT(1),W2CT(1)),
C      &      (WLBR(1),W2LBR(1))

```

```

C-----
IASF=161254
NH=NX
C CALL IWKIN(35000) !FOR IMSL
INFILE(1:20)=
TP_SOURCE(1)(1:20)='File:
TP_SOURCE(2)(1:20)='Delta function
TP_SOURCE(3)(1:20)='Plateau
TP_FILTER(1)(1:20)='None
TP_FILTER(2)(1:20)='Hanning
TP_FILTER(3)(1:20)='Shepp_Logan
TP_FILTER(4)(1:20)='Gaussian(FWHM= )
TP_FILTER(5)(1:20)='Hamming
TP_INTERP(1)(1:20)='Angular (bilinear)
TP_INTERP(2)(1:20)='Conventional bilinear
TP_INTERP(3)(1:20)='Distance
TP_PROC(0)(1:20)='PC IBM
TP_PROC(1)(1:20)='Mini-Map(Maxim-32)
TP_PROC(2)(1:20)='Mini-Map Emulator

tp_noise(0) = 'None
tp_noise(1) = 'Band limited
tp_noise(2) = 'Adap.MP/Pr( Blur)
TP_NOISE(3) = 'Rabbani(LMAP) (J=5)
tp_noise(4) = 'Kuan(LLMSE) (J=5)
TP_NOISE(5) = 'Ansc.Maeda (J=3)
TP_NOISE(6) = 'Ansc.Maeda_FS
TP_NOISE(7) = 'Ansc.Hann
tp_noise(8) = 'Ansc.Kuan(LMSE J=5)
TP_NOISE(9) = 'Ansc.Rabb (LMAP) XXX
TP_NOISE(10) = 'Ansc.Custo(L=3)
TP_NOISE(11) = 'Ansc.Custo5
TP_NOISE(12) = 'Ansc.Legend.

WRITE(*, 5)
5 format('1',///,
1 1X,' ### Programs for RECONSTRUCTION of images ###',/,
2 1X,' S.S.Furuie INCOR/BRAZIL',/,
3 1X,' T.Tomitani (Supervisor) NIRS /JAPAN',/,
3 1X,' 02/Nov/88',/,
3 //1X,' Method : .Noise suppression (Max.a post) ',/,
4 1X,' .Deblurring ',/,
2 1X,' .Direct Fourier with zero-padding ',/,
5 1X,' ',/))

WRITE(*, 10)NX,NY,NPAD
10 FORMAT(1H, 'Image reconstr.(opt.vers./RECLBL notation):',/,
1 1H, ' 1.physical radius = 1',/,
2 1H, ' 2.image (upper left)(1,1).. (',I5,I5,')',/,
3 1H, ' 3.FFT-ID of each PADDED projection (X',I4,')',/,
4 1H, ' 4.interpolation on freq. domain',/,
5 1H, ' 5.filtering',/,
6 1H, ' 6.IFFT-2D')

CALL BEEP
NNX=NX
NNY=NY
call GET_OPTIONS(TP_noise,12,Isorce,Itest,Ifilter,Interp,
1 Iproc,Iquality,Inoise_tec)
IF (IFILTER.EQ.3) THEN

```

```

CALL INQFLT(' Fwhm of filter(pixe))='C,FW,0.01,NX/2.0)
FWA=FW
FWB=FW
END IF
IF (ISOURCE.EQ.0) CALL INQANS
1      ( 'INPUT FILE (.PRJ) ? 'C, INFILE, 30 )
CALL INQANS ( 'OUTPUT FILE(.REC and .INF) ? 'C, OTFILE, 30 )
CALL INQNUM(' Log.Unit of OUTPUT(6=console,9=file ...INF)'C,
1      LUMOUT,0.1000)
LL=LEN_STR(OTFILE)
DO 12 I=1,LL
IF (OTFILE(I:1).EQ.'.')GOTO 13
12  ARGINF(I:1)=OTFILE(I:1)
L=LL
GOTO 14
13  L=L-1
14  ARGINF=ARGINF(1:L)//'.INF'C
if (itest.eq.5) then
NLX=NX
NAX=NY
ELSE
NLX=NL
NAX=NA
ENDIF
IF (ISOURCE.EQ.0) CALL INDATA( INFILE, PROJ, NLX, NAX, HD, NH ) !center =NL/2
IF (ISOURCE.EQ.1) THEN
DO 18 J=1,NA ! DELTA F.
DO 17 I=1,NL
17  PROJ(I,J)=0.0
18  PROJ(NL/2+1,J)=1.0/NA !to keep compatibility with proj.files
END IF
IF (ISOURCE.EQ.2) THEN
CALL INQNUM('Nivel do plateau='C,nplateau,1,1000)
DO 6 J=1,NA
DO 6 I=1,NL
6  PROJ(I,J)=nplateau
ENDIF
IF (ITEST.EQ.5) THEN
INOISE_TEC=0
IFILTER=0
INTERP=0
IPROC=0
EQUALITY=1
GOTO 92
END IF
IF (ITEST.EQ.6) GOTO 90
IF (ITEST.EQ.7) THEN
DO 7 J=1,NA
DO 7 I=1,NL
WS=PROJ(I,J)
7  PROJ(I,J)=RAND(WS,IASF)
GOTO 90
ENDIF
IF ((ITEST.EQ.8).or.(ITEST.EQ.9)) INOISE_TEC=5
IF (INOISE_TEC.EQ.0)GOTO 19
C (((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((
call msqim('Beginning Noise elimination..'C)
IF((Inoise_tec.GE.7).AND.(INOISE_TEC.LE.12))
8  call ANSCOMBE(PROJ,NL,NA)

```



```

      goto (26,27,15,16,31,32,1,2,3,4,4,1012), Inoise_tec
26  call BAND_LIN(PROJ,NL,NA)
      GOTO 19
27  CALL BLUR_BAYEST(PROJ,NL,NA,CR_TX,16,16)
      GOTO 19
15  CALL RABBANI(PROJ,NL,NA)
      GOTO 19
16  CALL KUAN(PROJ,NL,NA)
      goto 19
31  CALL MAEDA(PROJ,NL,NA,ITEST)
      IF ((ITEST.EQ.8).OR.(ITEST.EQ.9)) GOTO 90
      goto 19
32  CALL MAEDA_FS(PROJ,NL,NA)
      GOTO 19
C    apos transformacao de ANscombe
1    call FILTER1_2_1(PROJ,PROJ,NA*NL)
      GOTO 20
2    CALL GSS_KUAN(PROJ,NL,NA)
      GOTO 20
3    continue
C    CALL RABBANI(PROJ,NL,NA)
      GOTO 20
C    Legendick apos transf.Anscombe
1012 call INQFLT('Param.Regular.='C,alfa,0.0,100.0)
      call INQFLT('Fator Relaxacao='C,beta,0.0,100.0)
      call INQFLT('Lim.Varianca='C,sigma2,0.0,100.0)
      CALL LAG2D(PROJ,NL,NA,alfa,beta,sigma2)
      GOTO 20
4    CONTINUE
      CALL INQFLT('Param.Regulariz='C,alfa,0.0,100.0)
C    1-D FFT
      DO 9 J=1,NA
      DO 11 I=1,NL
      CI_XIN(I)=0.0
11   CR_XIN(I)=PROJ(I,J)
      CALL FFTS1(CR_XIN,CI_XIN,CR_XOUT,CI_XOUT,NL,WST,WCT,
      &          WLBR,2)
C    Filtros
C    CUSTO
      IF(INOISE_TEC.EQ.10) CALL I_LAPL(CR_XOUT,CI_XOUT,NL,ALFA)
      IF(INOISE_TEC.EQ.11) CALL ISLAPL(CR_XOUT,CI_XOUT,NL,ALFA)
C    1-D IFFT
      CALL FFTS1(CR_XOUT,CI_XOUT,CR_XIN,CI_XIN,NL,WST,WCT,WLBR,-2)
      DO 8 I=1,NL
8    PROJ(I,J)=CR_XIN(I)
9    CONTINUE
20   CALL I_ANSCOMBE(PROJ,NL,NA)
19   IF (ITEST.EQ.1) then
      NNX=NL
      NNY=NA
      GOTO 90
      endif
c ))))))) ...end of noise and deblurring )))))))

C (((((((((((((((((((((((...Zero padding and 1D-FFT ..((((((((((((((((
CONTINUE
CALL MSGTIM('Beginning FFT-1D of proj..'C)
C    output of FFT forward is not normalized
C    i.e. output=Sum(i=1,NL*NPAD)
C    goto (23,33,37), Iproc+1

```

```

33 CONTINUE
37 CONTINUE
C #####
23 CONTINUE
CTHSL CALL FFTCI(NL*NPAD,WFFTC) ! initialize FF1
WRITE(*,21)
21 FORMAT(1X)
DO 30 IA=1,NA
write(*,22)IA
22 FORMAT(1H+,I4)
DO 24 I=1,NL*NPAD ! Padding
CR_XIN(I)=0.0
CI_XIN(I)=0.0
24 CONTINUE
DO 25 I=1,NL
c II=I
II=NL*NPAD/2-NL/2+I
C !It is necessary to centralize in
c !space domain,AND DON'T shift right one
c !position ,because PROJECTION is centered at
c !(NL/2+1) and F2TCF is
c !at NL*NPAD/2+1
CR_XIN(II)=PROJ(I,IA) !ANG=PI/NA*IA
CI_XIN(II)=0.0
25 CONTINUE
C
C Optical_Fourier Transform .Ja normalizado por sqrt(NL*NPAD)
C
CALL FFTSI(CR_XIN,CI_XIN,CR_XOUT,CI_XOUT,NL*NPAD,WST,WCT,
& WLCR,2)
DO 28 I=1,NL*NPAD
CR_TEMP(I,IA)=CR_XOUT(I)
CI_TEMP(I,IA)=CI_XOUT(I)
28 CONTINUE
30 CONTINUE
GOTO 39
C #####
C #####...END OF 1d-FFT .. #####
c***** TEST 2 .....testing fft 1d
39 IF (ITEST.NE.2)GOTO 48
XMAX=1.0E-10
DO 40 J=1,NA
DO 40 I=1,NL
I1=NL*NPAD/2-NL/2+I
XX=CR_TEMP(I1,J)
YY=CI_TEMP(I1,J)
XX=SQRT(XX*XX+YY*YY)
PROJ(I,J)=XX
IF (PROJ(I,J).GT.XMAX) THEN
XMAX=PROJ(I,J)
IX=I
IY=J
END IF
40 CONTINUE
WRITE(*,45)XMAX,IX,IY
45 FORMAT(1X, '(ABS(MAX),IXMAX,IYMAX)=',G(3.1,215)
GOTO 90
c***** TEST 2.... end of test of fft 1 d

```



```

C CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
78  call MSGTIM('Beginning IFFT-2D.....')
C  output of IFFT-2D is scaled ( by NA/NX/NY)
C  NA is due to projections are divided by NA
  GOTO (111,200,300), IPROC+1

```

```

200  CONTINUE
300  CONTINUE

```

```

C CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

```

```

C  VAX/VMS ,IMSL option
111  continue

```

```

c
c versao HUGE do FFTS2
c Ja' normaliza pelo sqrt(NX*NY)

```

```

      CALL FFTS2(CR_FREQ,CI_FREQ,CR_TX,CI_TX,NX,NY,
&              W2ST,W2CT,W2LBR,NX+NY,-2,JERR)

```

```

      IF(JERR.NE.0) WRITE(*,*) 'Erro no FFTS2'

```

```

C  scale :=NA devido a atenuacao na geracao das proj.(PHAGENFS)
C  *sqrt(NPAD.NL.NX.NY) :compensar atenuacao em FFTS1 e FFTS2
C  / (NX,NY)           : fator de normalizacao do FFT (definicao)
scale=NA*sqrt(NPAD*1.0*NL)/SQRT(NX*1.0*NY)

```

```

C  pondo no formato (f.f) canto sup.esq.

```

```

      N0=NX/2+1
      DO 80 J=1,NY
        DO 80 I=1,NX
          R=SQRT(1.0*(I-N0)*(I-N0)+1.0*(J-N0)*(J-N0))+1.0E-6
          IF(R.LT.(NX/2-1)) THEN
            PROJ(I,J)=CR_TX(I,NY-J+1)*SCALE
          ELSE
            PROJ(I,J)=0.0
          ENDIF

```

!NORMALIZING

```

80    CONTINUE
      GOTO 400

```

```

C )))))))))))))))))))))))))))))))))))))))))))))))))))))))))

```

```

C )))))))))))))))))))...END OF 2D-FFT .. )))))))))))))))))))))))

```

```

400  CALL MSGTIM('End of reconstruction')

```

```

90  CONTINUE
  OBS='Imagem resultado:vide arquivo.INF '
  CALL OUTARQ( OFILE, PROJ ,NNX,NNY, OBS )
  CALL MSGTIM('End of storing')
  IF (ITEST.EQ.1) GOTO 9000

```

```

92  CALL TOT_COUNT(PROJ,NX,NY,XMIN,XMAX,XMED,I_MAX,J_MAX,
1    TOT_ABS,TOT_POS,TOT_2)

```

```

  CALL BEEP
  IF (LUNOUT.NE.6) THEN
    L=LEN_STR(ARGINF)
    OPEN(UNIT=LUNOUT,ACCESS='SEQUENTIAL',
&       FILE=ARGINF(1:L),STATUS='UNKNOWN')

```

END IF

```

WRITE(LUNOUT,95)NPAD,TP_SOURCE(ISOURCE+1),INFILE,ITEST,TP_NOISE
1 (INOISE_TEC), TP_FILTER(IFILTER+1),TP_INTERP(INTERP+1)(1:20),
2 OTFILE,TP_PROG(Iproc)
95 FORMAT(//,1X,'Options selected:',/,
1 1X,'Padding :X',I4,/,
2 1X,'Source :',A20,2X,A20,/,
3 1X,'Test :',I4,/,
3 1X,'Noise@81ur Met:',A20,/,
4 1X,'Filter :',A20,/,
5 1X,'Interpolation :',A20,/,
6 1X,'Output file :',A20,/,
7 1X,'Processor used:',A20,/)
IF (IFILTER.EQ.3) WRITE(LUNOUT,96)FW
96 FORMAT(1X,'FWHM for filter=',F10.2)
IF((INOISE_TEC.GE.10).AND.(INOISE_TEC.LE.12))WRITE(LUNOUT,93)
& alfa
IF(INOISE_TEC.EQ.12)WRITE(LUNOUT,94)beta
93 FORMAT(1X,'Param.Regulariz=',F10.3)
94 FORMAT(1X,'Fator Relaxacao=',F10.3)
WRITE(LUNOUT,97)XMIN,XMED,XMAX,PROJ(NX/2+1,NY/2+1),L_MAX,J_MAX,
1 TOT_ABS,TOT_POS
97 FORMAT (1X,' Min,Med,Max,at center : ',4F12.3,/,
1 1X,' Pos. [r,c] of max : ',2I10,/,
2 1X,' Total [abs,pos]count : ',2G12.6)
IF (IQUALITY.EQ.1) THEN
call INGANS(' File (.IMG) to compare with 'C,INFILE,30)
CALL INQNUM(' total band(% of radius)='C,IPEC,5,200)
CALL INDATA(INFILE,CR_TX,NX,NY,HD,NH)
write(*,99)NX,NY
99 FORMAT(1X,'NX,NY=',2I4)
CALL MEAS_QUALITY(PROJ,CR_TX,NX,NY,QUALITY,A_NMSE,
1 B_NMSE,OUT_MAX,D_POS,D_NEG,out_rms,centro,
2 IPERC)
WRITE(LUNOUT,500) QUALITY,A_NMSE,IPERC/2.0,B_NMSE,
1 centro,OUT_MAX,D_POS,D_NEG,out_rms
500 FORMAT(1X,'GLOBAL NRMSE(-1,1) =',F13.4,/,
1 1X,'INNER NRMSE(-1,1) =',F13.4,/,
2 1X,'EDGE(+-,F8.2.% ) NRMSE(-1,1) =',F13.4,/,
2 1X,'valor no centro do circ.ideal =',F13.2,/,
3 1X,'Outside max.(%) (central value)='F13.3,/,
4 1X,'Positive max. error(Z) =',F13.3,/,
5 1X,'Negative max. error(Z) =',F13.3,/,
6 1X,'Outside(Z) RMSE/(Value at center)='F13.3)
OBS='Arquivo imagem ideal - imagem reconstruida'C
CALL OUTARQ('TESTE.DIF', PROJ , NX, NY, OBS )
WRITE(*,510)
END IF
510 FORMAT(//,1X,' Difference image on file:TESTE.DIF')
IF (LUNOUT.NE.6) CLOSE (UNIT=LUNOUT)
STOP 'Ok.'
9000 STOP '4'
END

```

```

c file: RC.SUB.FOR set of subroutines for reconstruction using Direct
c Fourier methods
C 18/Nov/88 Furue,S.S
C MODIFIED : in order to fit in an IBM-PC

```



```

20  FORMAT(1H+,I4)
    DO 200 I=1,NY
      WRITE(*,20)I
      Y=INT(I-NC_Y)*DT_Y
      DO 100 J=1,NX
        X=INT(J-NC_X)*DT_X
        R=SQRT(X**2+Y**2)/DT_R
        IF (R.LT.N0) THEN
          R=-R*ISIGN(1,J-NC_X)
          IF (J.EQ.(NC_X)) THEN
            ALFA=0.0
            R=ABS(R)*ISIGN(1,I-NC_Y)
          ELSE
            ALFA=ATAN(Y/X)+PI/2
            IF (ALFA.LT.0) ALFA=ALFA+PI
          END IF
          ALFA=ALFA/DT_TETA+1
          IANG1=INT(ALFA)
          IF (IANG1.GT.NA) STOP 'ERROR=Iang2 > NA'
          IANG2=IANG1+1
          IR1=INT(R)
          IF (IR1.EQ.0) THEN
            IR2=IR1+ISIGN(1,NINT(SIGN(1.0,R+EPS)))
          ELSE
            IR2=IR1+ISIGN(1,IR1)
          END IF
          COEF=(R-IR1)/(IR2-IR1)
          IX1=IR1+NC_L
          IX2=IR2+NC_L
          IF ((IX2.LT.1).OR.(IX2.GT.(2*N0))) THEN
            BR(J,I)=0.0
            BI(J,I)=0.0
          ELSE
            IF((I.EQ.NC_Y).AND.(J.EQ.NC_X))THEN
              BR(J,I)=AR(NC_L,I)
              BI(J,I)=AI(NC_L,I)
              GOTO 60
            ELSE
              FR11=AR(IX1,IANG1)
              FI11=AI(IX1,IANG1)
              FR12=AR(IX2,IANG1)
              FI12=AI(IX2,IANG1)
              FR21=AR(IX1,IANG2)
              FI21=AI(IX1,IANG2)
              FR22=AR(IX2,IANG2)
              FI22=AI(IX2,IANG2)
            END IF
            GOTO (30,40,50),ITYPE+1
            Angular bi_linear interpl.
            FRA=FR11+(ALFA-IANG1)*(FR21-FR11)
            FIA=FI11+(ALFA-IANG1)*(FI21-FI11)
            FR0=FR12+(ALFA-IANG1)*(FR22-FR12)
            FIB=FI12+(ALFA-IANG1)*(FI22-FI12)
            BR(J,I)=FRA+COEF*(FR0-FRA)
            BI(J,I)=FIA+COEF*(FIB-FIA)
            GOTO 60
            continue
          CONTINUE
          ! NL,NPAD(R(NL,NPAD
          ! IANG=(1,NA)
          ! NEXT
          ! extremities of cir.
          ! IANG2-IANG1=1

```

C  
30  
40  
50









```

C      .LLMSE technique
C      f=E(q)+VAR(f)/(var(f)+E(f)) * (q(i)-E(qi))
C      E(qi)=Sum[k=IL,IS] (q(i+k))*w(k)           !ML if w(k)=cte
C      E(fi)=E(qi)
C      var(fi)=var(qi)-E(fi)
C      var(qi)=sum[k=IL,IS]wv(k)(q(i+k)-E(q(i+k)))**2   !sum(w(k))=1=sum(wv)
C      var(fi)=max(var(fi),0)
C      Obs.:
C      . in local var(qi), we use point average           !!
C C      . non-negativity constraint
C C      . forces same counts/projection
C      modification: 03/nov/88      04/NOV/88      7/NOV/88
C      *****

PARAMETER (NAP=64,NLP=512,NWINDOW=5)

REAL*4      W(-2:2),WV(-2:2),SUM(NAP),EF(NLP), EG(NLP), G0(NLP)
DATA      PI/3.141593/ , EPS/1.0E-10/
IL=-NWINDOW/2           ! WINDOW
IS=NWINDOW/2
WV(-2)=1.0
WV(-1)=2.0
WV(0)=4.0
WV(1)=2.0
WV(2)=1.0
S=0.0
DO 2 I=IL,IS
W(I)=1.0/NWINDOW
2      S=S+WV(I)
DO 3 I=IL,IS
3      WV(I)=WV(I)/S
S=0.0

DO 200 J=1,NA
SUM(J)=0.0           !TOTAL per projection
DO 5 I=1,NL           !not recursive
5      G0(I)=G(I,J)
DO 10 I=1,-IL           ! at boundary w/o estimation
10     EG(I)=G0(I)     ! G(i,j) not altered
EF(I)=G0(I)
SUM(J)=SUM(J)+EG(I)
10     CONTINUE

DO 20 I=NL-IS+1,NL
EG(I)=G0(I)
EF(I)=G0(I)
SUM(J)=SUM(J)+EG(I)
20     CONTINUE

DO 50 I=1-IL,NL-IS           !estimating E(q)
EG(I)=0.0
DO 30 K=IL,IS
30     EG(I)=EG(I)+G0(I+K)*w(k)
EF(I)=EG(I)
50     CONTINUE

DO 100 I=1-IL,NL-IS           !estimation of var(q) and f
var2q=0.0
DO 60 K=IL,IS
60     VAR2G=VAR2G+(G0(I+K)-EG(I+K))*(G0(I+K)-EG(I+K))*WV(k)           !Kuan

```



```

var_ruido=1.0
IL=-NWINDOW/2
IS=NWINDOW/2
WV(-2)=1.0
WV(-1)=2.0
WV(0)=4.0
WV(1)=2.0
WV(2)=1.0
S=0.0
DO 2 I=IL,IS
W(I)=1.0/NWINDOW
2   S=S+WV(I)
DO 3 I=IL,IS
3   WV(I)=WV(I)/S

S=0.0

DO 200 J=1,NA
SUM(J)=0.0
DO 5 I=1,NL
5   G0(I)=G(I,J)
DO 10 I=1,-IL
10  EG(I)=G0(I)
EF(I)=G0(I)
SUM(J)=SUM(J)+EG(I)
CONTINUE

DO 20 I=NL-IS+1,NL
20  EG(I)=G0(I)
EF(I)=G0(I)
SUM(J)=SUM(J)+EG(I)
CONTINUE

DO 50 I=1-IL,NL-IS
50  EG(I)=0.0
DO 30 K=IL,IS
30  EG(I)=EG(I)+G0(I+K)*w(k)
EF(I)=EG(I)
CONTINUE

DO 100 I=1-IL,NL-IS
100 VAR2G=VAR2G+(G0(I+K)-EG(I+K))*(G0(I+K)-EG(I+K))*Wv(k)
100 VAR2F=VAR2G-var_ruido
IF (VAR2F.LT.0)VAR2F=0.0
IF ((VAR2F+var_ruido).LT.EPS) THEN
ELSE
alfa=VAR2F/(VAR2F+var_ruido)
F=alfa*G0(I)+(1-alfa)*EF(I)
END IF
C   IF (F.LT.0) F=0.0
SUM(J)=SUM(J)+F
G(I,J)=F
100 CONTINUE
200 S=S+SUM(J)

C   ascertain that same total per projection
C   DO 500 J=1,NA
! WINDOW
!TOTAL per projection
!not recursive
! at boudary w/o estimation
! G(i,j) not altered
!estimating E(g)
!estimation of var(q) and f
!Kuan

```











200

CONTINUE  
IF (I.FEST.EQ.9) RETURN  
RETURN  
END

/\* File : TOMOSF.C

Sergio Furuie SP0SE/TNCOR

Programa para reconstrucao tomografica  
.geracao de phantoms  
.varios metodos de reconstrucao  
.armazenamento/recuperacao do disco

Modificacoes:

11/abr/89	12/abr/89	13/abr/89		
14/abr/89	16/abr/89	21/abr/89		
24/ABR/89	25/abr/89	01/MAI/89		
18/mai/89	20/mai/89	22/mai/89		
23/mai/89	24/mai/89	25/mai/89	26/maio/89	
27/mai/89	29/mai/89	30/mai/89	02/jun/89	
14/jun/89	18/jul/89	19/JUL/89	31/jul/89	02/ago/89
04/ago/89	15/ago/89	16/ago/89	17/ago/89	19/ago/89
24/ago/89	25/ago/89s	26/ago/89	30/out/89	3/nov/89
7/nov/89	08/nov/89	09/nov/89	10/nov/89	29/nov/89
27/dez/89	4/2/90	16/fev/90		

\*/

#include <stdio.h>  
#include <math.h>  
#include <stdlib.h>  
#include <errno.h>  
#include <malloc.h>  
#include <conio.h>

#include <dos.h>  
#include <bios.h>

#include <atr.h> /\* includes do SITIM-150 \*/  
#include <info.h>  
#include <dia.h>  
#include <menu.h>

#include <ctype.h>  
#include <graph.h>

#include "constant.c" /\* includes gerais,defines,types e var globais \*/

/\* check de consistencia \*/

#include "imga\_bib.h"  
#include "disc\_bib.h"  
#include "mat\_bib.h"  
#include "util\_bib.h"

#include "tomo\_bib.h"  
#include "tomo\_ite.h"

float huge \*tabela\_psf;

/\*( .....function ..... \*/

```

int RECONSTR(float far *proj,int janela_proj[4],int EXISTE_PROJ,
            float far *rec_im,int janela[4],unsigned char *BUF_UVI,
            float far *tabela_psf, float far *pbuf3)

/* Algoritmos de reconstrucao
   .retorna a imagem reconstruida em *rec_im, com as caracteristicas em 'janela'
   .as projecoes estao em '*proj' e 'janela_proj'
   .EXISTE_PROJ =1 -> ja' foi carregado projecoes em 'proj.'
               0 -> nao
   * usa BUF_UVI como matriz de PSF (H)
   )*/
(
register int i,j;
int
    nlin,ncol,nview,nsamp,EXISTE_PSF,EXISTE_TAB,
    errosf,opcao_to, FIM_OP, STORE_REC,
    x,y,HGRANDE,ndist,
    n_op,opcao,a,k.iter0=0;

texto20x40 s;
char
    ch,*args[2],arq[TAM_NOME], arq[TAM_NOME],obs[TAM_OBS],sx[40];
struct
    IMAG_INFO H_i,tabela_i,imag_i;
FILE
    *pfile;
float
    *H, z,xx,d_d,lpixel;

H=BUF_UVI; lpixel=1.0; strcpy(arqH,"");
errosf=0; EXISTE_PSF=EXISTE_TAB=0;
s[0]="sub_menu:Reconstrucao";
s[1]="SOTC * retirado*";
s[2]="Ler imagem pre-rec.";
s[3]="ART ";
s[4]="EM";
s[5]="2 passos";
s[6]="Gerar tabela p/ matriz de PSF";
s[7]="Ler tabela p/ matriz de PSF";
s[8]="CBP=tomo_for/imagfs";
s[9]="DTF=tomo_for/rc_ft";
n_op=9;

ndist=50;
nlin=janela[3];          ncol=janela[1];
nview=janela_proj[3];   nsamp=janela_proj[1];

if(((long)nview*nsamp*nlin*ncol) >= (long)64*64) HGRANDE=1;
else HGRANDE=0;

opcao_to=1; FIM_OP=1; STORE_REC=0;
do (inicio_rec:
    if((opcao_to < 1)|| (opcao_to > n_op)) opcao_to=1;
    opcao_to=CONS_MENU(n_op,s,opcao_to,LINHA_MENU,COL_INIC_FUNC,BLINK,NOINTENS0);
    switch(opcao_to)
        (case 0:dia_limpa(YMSG,1);
            if(dia_pegq_string(YMSG,XMSG,"s,n","Retorno.Certeza (s/n):",&ch ,1)==FIM)break;
            if (ch!='s') break;
            FIM_OP=0; dia_limpa(3,1);break;
        case 8:
        case 9: break;
        case 2:
            /* carregar imagem pre-reconstruida */
            dia_limpa(YMSG,XMSG);
            if(dia_pegq_string(YMSG,XMSG,"nome do","arquivo imag.",arq,TAM_NOME)==FIM) break;
            dia_cursor(YMSG-3,1);
            if(DISC_MATRIZ_LE(arq.pbuf3.&imag_i)!=0)break;

```

```

i=imaq_i.num_cols; j=imaq_i.num_lins;
if((i != ncol) || (j != nlin)) (CONS_MENSAG("Imagem incompativel "); break;)
if(INGNUM("Qual num.de iter.ja'proc.?", &iter0, 0, 1000) != 0) break;
/*re-transpor e reinverter eixos (90-270 graus projecoes*/
for (i=1; i<nlin;i++) /* a partir de 1 pois a troca de col deve manter a posicao relativa */
  for (j=0; j<ncol;j++) (*(rec_im+0*ncol+i)*=(pbuf3+(ncol-j-1)*nlin);) /* nlin=ncol*/
for (i=1; i<nlin;i++) /* a partir de 1 pois a troca de col deve manter a posicao relativa */
  for (j=0; j<ncol;j++) (*(rec_im+i*ncol+j)*=(pbuf3+(ncol-j-1)*nlin+(nlin-i));) /* nlin=ncol

CONS_MENSAG("Ok. Lido tabela do disco");
dia_limpa(YMSG-3,XMSG);
break;
case 3:
case 4:
dia_limpa(YMSG,XMSG);
if (!EXISTE_PROJ)(CONS_MENSAG("Leia antes as proj.");break;)
if(iter0!=0) (if(dia_peg_string(YMSG,XMSG,"sim,nao","Re-inicializar?(s/n)=",&ch,1)==FIN)break;)
if((ch=='s') || (ch=='S')) iter0=0;
/*if(dia_peg_string(YMSG,XMSG,"t=tabela.h=PSF","modo de calc.=",&ch,1)==FIN)break;
*/ ch='t';
if((ch=='t') || (ch=='T'))
  (if (!EXISTE_TAB) (CONS_MENSAG("Leia antes ou gere a TABELA"); break;)
  switch(opcao_to)
    ( case 1:/*if(SOTC(proj, nview, nsamp, rec_im, nlin, ncol,tabela_psf,ndist.nview,
      pbuf3,-lpixel)!=0)break;
      */
      break;
    case 3:if(ART(proj, nview, nsamp, rec_im, nlin, ncol,tabela_psf,ndist.nview,
      H,1,0,-lpixel,&iter0)!=0)break;
      break;
    case 4:if(EH(proj, nview, nsamp, rec_im, nlin, ncol,tabela_psf,ndist.nview,
      H,pbuf3,-lpixel,&iter0)!=0)break;
      break;
    )
  )
/* else (if (!EXISTE_PSF) (CONS_MENSAG("Leia antes ou gere a PSF"); break;)
switch(opcao_to)
  ( case 1:if(SOTC(proj, nview, nsamp, rec_im, nlin, ncol,tabela_psf,ndist.nview,
    pbuf3,-lpixel)!=0)break;
    break;
  case 3:if(ART(proj, nview, nsamp, rec_im, nlin, ncol,tabela_psf,ndist.nview,
    H,0,arqH,1,0,-lpixel,&iter0)!=0)break;
    break;
  case 4:if(EM(proj, nview, nsamp, rec_im, nlin, ncol,tabela_psf,ndist.nview,
    H,0,arqH,pbuf3,-lpixel,&iter0)!=0)break;
    break;
  )
)
*/
if(INGANS("Armaz.imaq.rec. em disco?(s/n)=",&ch,1)!=0) break;
STORE_REC=1;
if((ch=='n') || (ch=='N')) STORE_REC=0;
break;
case 5:
break;
case 6: x=nview;
if(INGNUM("Num.de views=".3x,4,100) !=0)break;
if(INGFLT("Tam.pixel(em bin)=",&lpixel,0,25,4,0) !=0)break;
i=ECT_PSF_TAB(tabela_psf, ndist,x,lpixel,lpixel,1,0,50);

```

```

nview=x;
if(i==0) EXISTE_TAB=1;
if(INGANS("Armacenar em disco ?(s/n):",&ch,1)!=0) break;
if((ch!='n')&&(ch!='N'))
  (dia_limpa(YMSG,XMSG);
  if(dia_peg_string(YMSG,XMSG,"nome da","tabela:",arq.TAM_NONE)==FIM) break;
  gcvt(1/pixel,6,obs); strcat(obs," ");
  if(dia_peg_string(YMSG+1,XMSG,"comentario","obs.:",sx,TAM_OBS)==FIM) break;
  strcat(obs,sx);
  if(DISC_MATRIZ_ESC(arq,tabela_psf,ndist,nview,obs) != 0) perror ("Erro:RECONSTR");
  break;
  )
  )
CONS_MENSAG("Ok.");
break;
case 7: dia_limpa(YMSG,XMSG);
if(dia_peg_string(YMSG,XMSG,"nome do","arquivo tab.:",arq.TAM_NONE)==FIM) break;
dia_cursor(YMSG-3,1);
if(DISC_MATRIZ_LE(arq,tabela_psf,&tabela_i)!=0)break;
strcpy(obs,tabela_i.obs);
1/pixel=atof(obs); if (1/pixel < EPSILON) 1/pixel=1.0;
printf("\n 1/pixel=%Z10.4f",1/pixel);
i=tabela_i.num_cols;
if(i != nview)( CONS_MENSAG("Tabela incompativel com NVIEW"); break;)
/* check de consistencia na tabela (deve ser 1)*/
xx=0.0; d_d=(1+1/pixel*sqrt(2.0))/2.0/ndist;
for (a=0; a<nview; a++)
  (z=0.0;
  for (k=-ndist;k<=ndist;k++)
    (i=k; if(k<0) i=-1*k;
    z=z+(*(tabela_psf+i*nview+a));
    )
  xx=xx+z;
  dia_cursor(YMSG+2,1);printf("ang=%Z3d soma(1.0)=%Z10.3f ,Acum(%Z3d)=%Z10.3f".a,z*d_d,nview,xx*d_d)
  )
getchar();
CONS_MENSAG("Ok. Lido tabela do disco");
EXISTE_TAB=1;
dia_limpa(YMSG-3,XMSG);
break;
}
if(STORE_REC) (
dia_limpa(YMSG,XMSG);
if(INGANS("Reprojecao. Armacenar em disco ?(s/n):",&ch,1)!=0) break;
if((ch=='s')||(ch=='S'))
  (
  if(dia_peg_string(YMSG,XMSG,"nome ","arq.Reprojecao:",arq.TAM_NONE)==FIM) goto inicio_rec;
  if(dia_peg_string(YMSG+1,XMSG,"comentario","obs.:",obs,TAM_OBS)==FIM) goto inicio_rec;
  if(DISC_MATRIZ_ESC(arq,pbuf3,nview,nsamp,obs) != 0) perror ("Erro:RECONSTR");
  )
  )
dia_limpa(YMSG,XMSG);
if(dia_peg_string(YMSG,XMSG,"nome do","arq.p/Reconst.:",arq.TAM_NONE)==FIM) goto inicio_rec;
if(dia_peg_string(YMSG+1,XMSG,"comentario","obs.:",obs,TAM_OBS)==FIM) goto inicio_rec;
/* transposta e inverter eixos,para por no formato (0.0)=upper left*/
for (i=1; i<nlin;i++) /* a partir de 1 pois a troca de col deve manter a posicao relativa +
  for (j=0; j<ncol;j++) (*(pbuf3+(ncol-j-1)*nlin)=*(rec_lm+0*ncol+i)); /* nlin=ncol*/
for (i=1; i<nlin;i++) /* a partir de 1 pois a troca de col deve manter a posicao relativa +
  for (j=0; j<ncol;j++) (*(pbuf3+(ncol-j-1)*nlin+i)=*(rec_lm+i*ncol+j)); /* nlin=ncol
if(DISC_MATRIZ_ESC(arq,pbuf3,ncol,nlin,obs) != 0) perror ("Erro:RECONSTR");

```

```

else STORE_REC=0;
    )
} while (FIM_OP):

return (errosf);
}

/* .....function ..... */

int TOMO(float far *proj,float far *rec_im, unsigned char *8UF_UVI,float far *tabela_psf,
float far *pbuf3)

/*
Bloco de rotinas envolvendo TOMOGRAFIA
.gerar matriz de projecoes
.visualizar matriz de proj.          ok.28/2/89
.metodos de reconstrucao:
    -transformada de Fourier
    -Convolution Back Proj.
    -ART
    -EM
    -Iterativo e regularizado
    -Iterativo, regularizado e adaptativo
.visualizar matriz de imagens
.armazenar matriz em disco
.ler matriz do disco

* coloca o SITIM no quadrante (0,0)
* fator de zoom depende da dimensao da matriz 'proj'
  (nlin,ncol)
* visualiza as projecoes no canto sup. esq. e a imagem ao lado dir.

*/

(
register int i,j;
int janela_proj[4],janela[4].EXISTE_PROJ,
errosf,opcao_to, FIM_OP,
x,y,nlin,ncol,N_VIEW,N_SAMP,NLTN,NCOL,
n_dp,opcao;
float      xx;
texto20x40 s;
char      ch.arq_imaqem[TAM_NOME], observacao[TAM_OBS],*arqs[2];
struct    IMAG_INFO      proj_i;

/* .....INICIALIZANDO.....*/

errosf=0;
s[0]="sub_menu:Tomografia";
s[1]="Simular Projecoes";
s[2]="Carregar Projecoes";
s[3]="Reconstrucao";
s[4]="Tonalidade do SITIM";
s[5]="Apagar tela do SITIM";

```

```
s[6]="Visualizar arquivo ";
n_op=6;
```

-141-

```
/* .....INICIALIZACAO DE VARIAVEIS..... */
```

```
/* ..... ROTINA .....*/
```

```
opcao_to=1; FIM_OP=1; EXISTE_PROJ=0;
```

```
do (inicio_to:
```

```
    dia_titulo("Sub-menu :Tomografia - SPDSE/INCOR");
```

```
    opcao_to=CONS_MENU(n_op,s,opcao_to,LINHA_MENU,COL_INIC_BL,8LINK,NOINTENSO);
```

```
    switch(opcao_to)
```

```
        (case 0:if(dia_peg_string(YMSG,XMSG,"s.n","Retorno.Certeza (s/n):",&ch ,1)==FIM)break;
```

```
            if (ch!='s') break;
```

```
            FIM_OP=0; dia_limpa(3,1);break;
```

```
        case 1: N_VIEW=N_SAMP=32;
```

```
            if(INQNUM("Num.de views:",&N_VIEW,4,128) != 0) break;
```

```
            if(INQNUM("Num.de amostras:",&N_SAMP,4,128) != 0) break;
```

```
            nlin=N_VIEW; ncol=N_SAMP;
```

```
            errosf=PROJ_GERA(proj,nlin,ncol,janela_proj);
```

```
            if(errosf==0)CONS_MENSAG(" Ok.<RET>..");
```

```
            EXISTE_PROJ=1;
```

```
            break;
```

```
        case 2: if(dia_peg_string(YMSG,XMSG,"nome do","arquivo:",arq_imagem,TAM_NOME)==FIM) break;
```

```
            dia_cursor(YMSG-3,1);
```

```
            if(DISC_MATRIZ_LE(arq_imagem, proj,&proj_i)!=0)break;
```

```
            nlin=proj_i.num_lins;
```

```
            ncol=proj_i.num_cols;
```

```
            /* check de total por linha */
```

```
            xx=0.0;for(i=0;i(nlin:i++)
```

```
                (
```

```
                    for(j=0;j(ncol:j++) xx=xx+*(proj+i*ncol+j));
```

```
                    dia_cursor(YMSG+4,XMSG+20);printf("Soma ate 'linha %3d=%10.1f",i,xx);
```

```
                )
```

```
            CONS_MENSAG("Ok. Lido arquivo em disco");
```

```
            EXISTE_PROJ=1;
```

```
            janela_proj[0]=janela_proj[2]=0;
```

```
            janela_proj[1]=ncol; janela_proj[3]=nlin;
```

```
            /* if(IMAG_POS_VOO_ZOOM(janela_proj) !=0) break;
```

```
            errosf=IMAG_MATRIZ_VIS(proj,nlin,ncol,janela_proj,1);
```

```
            if(errosf==0)(CONS_MENSAG(" Ok.<RET>..");
```

```
                IMAG_BORDA(CANALSF,janela_proj);
```

```
            )
```

```
            */
```

```
            break;
```

```
        case 3: NLIN=janela_proj[3]; NCOL=janela_proj[1];
```

```
            if((NLIN <0)||((NLIN)>128)||((NCOL<0)||((NCOL)>128)) (NLIN=32; NCOL=32;)
```

```
            if(INQNUM("Num. de linhas p/ imagem rec.:",&NLIN,4,128) !=0)break;
```

```
            if(INQNUM("Num. de colunas p/ imagem rec.:",&NCOL,4,128) !=0)break;
```

```
            janela[1]=NCOL; janela[3]=NLIN;
```

```
            errosf=RECONSTR(proj,janela_proj,EXISTE_PROJ,rec_im,janela.8UF_UVI,tabela_psf,pbuf3);
```

```
            if(errosf==0)CONS_MENSAG(" Ok.<RET>..");
```

```
            break;
```

```
        case 4: uvi_ini(); uvi_voo(0.0);
```

```
            IMAG_TONS();
```

```
            break;
```

```
        case 5: uvi_ini(); uvi_voo(0,0);
```

```
            APAGUE(janela_proj);
```



```
        break;
    case 6: args[0]="MATRDISP"; args[1]=NULL;
           UTIL_EXEC("\\furuie\\desenv_c\\matrdisp.exe",args);
           break;
    )
) while (FIM_OP);

/* ..... FINALIZANDO E SALVANDO INFORMACOES NOVAS OU ALTERADAS ....*/

return (errosf);
)

/* *****-----PROGRAMA PRINCIPAL-----***** */

char erro_mem[30]=" Erro:mem.insuf.no ";

float huge *pbufh1;
float huge *pbufh2;
float huge *pbufh3;
float huge *pbufh4;

main()
(
/* .....VARIABLES LOCAIS AO MAIN .....*/
unsigned int      ux,uy1,uy2,uy3,uy4,mem_disponivel;
unsigned long int lx;
struct SREGS     segregs;

/* .....INICIALIZACAO .....*/

dia_titulo(" Parametros iniciais p/ check");
dia_limpa(3,1);

segread(&segregs); /* ler registros */
printf(" hexa : cs=%x, ds=%x, es=%x, ss=%x\n",segregs.cs,segregs.ds,segregs.es,segregs.ss);

pi=4.0*atan(1.0);

/* alocar area no heap para uso geral ( armazenamento temporario da imagem..)
   Sera' alocado area para 4 imagens (float ) 128 x 128 no heap( DOS),
   */
lx=(unsigned long) 100*64*sizeof(float)/16; /* num. de paragr.*/
uy1=(segregs.ss+0x1000);
printf("\nmemsize=%u kB end.topo=%lu",_bios_memsz().(long)uy1*16);
mem_disponivel=_bios_memsz()/16*1000-(segregs.ss + 0x1000);
printf("\n\n solicitado 4x(100x64x4B)=%lu B disponivel %lu B", (long)4*16*lx, (long)16*mem_disponivel);

(
strcat(erro_mem," HEAP");
lx=(unsigned long) 100*64;
```

```

pbufh1=(float huge *)malloc(1x,sizeof(float));
if (pbufh1==NULL) (CONS_MENSAG(erro_mem);exit(1));
pbufh2=(float huge *)malloc(1x,sizeof(float));
if (pbufh2==NULL) (CONS_MENSAG(erro_mem);exit(2));
pbufh3=(float huge *)malloc(1x,sizeof(float));
if (pbufh3==NULL) (CONS_MENSAG(erro_mem);exit(3));
pbufh4=(float huge *)malloc(1x,sizeof(float));
if (pbufh4==NULL) (CONS_MENSAG(erro_mem);exit(4));
tabela_psf=(float huge *)malloc(1x,sizeof(float));
if(tabela_psf==NULL) (CONS_MENSAG(erro_mem); exit(11));

printf("\n\n alocado 4 x mem. %lu (float)\n dispon.no heap=%zuBytes",1x,_memavl());
printf("\n a partir de %p H ate %p",pbufh1,tabela_psf);
CONS_MENSAG("Ok. Alocado no HEAP");
}
pbuf1=pbufh1;
pbuf2=pbufh2;
pbuf3=pbufh3;
pbuf4=pbufh4;
/* alocando espaco p/ BUF_UVI 128x128 char */
BUF_UVI=(unsigned char *)malloc((unsigned int)(128*128*sizeof(char)));
if (BUF_UVI==NULL) (CONS_MENSAG("IMAG_MATR [Z_VTS:Mem.insuf.no heap"); exit(11));

TOMO(pbuf1,pbuf2,BUF_UVI,tabela_psf,pbuf3);

(hfree(pbufh1);
hfree(pbufh2);
hfree(pbufh3);
hfree(pbufh4);
hfree(tabela_psf);

free(BUF_UVI);
CONS_MENSAG("Ok. De-alocado heap");
}

dia_cursor(23,0);

) /* . */

/*( .....function ..... */

int EM(float far *proi,      int nview, int nsamp,
float far *rec_im,      int nlin, int ncol,
float far *tabela_psf, int ndist, int nanq,
float *H,
float far *pbuf3,      double lpixel, int *(ITERACA00)

/* Metodo EM de reconstrucao
.(0,0) no canto inf. esq.
.assume lpixel=lupixel=lpixel
      lsamp=faixa_raios=1.0
. centro da imagem original projetada no centro do bin='nsamp/2'+1
. cada amostra na imagem e' o centro sdo pixel

```

```

.se 'lpixel' ( 0 -) emissao
      ) 0 -) transmissao
.se TABELA=1 calcular usando a tabela.
      else le matriz de proj. H[nview*nsamp][nlin*ncol] do disco linha a
          linha ou coluna a coluna
.assume que matriz de proj.H esta normalizada por coluna (pixel)=1
.yi^=sumi( Hij,zj)/EMISSAO
.zlj=z[lj]/(sum_i(Hij)). sum_i(Hij.yi/yi^ )
. ITER0: numero de iteracao inicial. Se 0 inicializa uniformemente
* Usa global auxiliar pbuf3 p/ reprojecao
)*/

```

```

( register      int          i,j,c,l;
                int          ji,js, ZERAR.n,k,CONVERG1U.colH,rowH,janela[4],janelaHC[4],
                              nerro=0,zoom,NJ,EMISSAO,N_ITER=5,ITER0,TABELA,ci,cs;
                float        LXPIX,LSAMP,x,y,yp,z,hlc,tpix,tproj,pi,seno,coseno,cosxx,cte,yy,
                              LYPIX,like_ant,vero,RUIDO,l diaq,dmaior,EPSILON2,vari,d_c;
                double        w,likelihood,distancia_p;
                char          ch[10],arq[10];
                FILE          *pfile;

```

```

ITER0=*ITERACA00;      TABELA=1;      EPSILON2=0.1/nview;      /* 1 decimo de 1 foton */
if (!TABELA) return(-1);      /* retirado devido a rotina 'reprojecao' */
RUIDO= EPSILON;      /* abaixo deste valor de proj. os pixels sao postos a zero */
n=0; if(nlin>32) zoom=1;
      else zoom=2;
if(IN0NUM("Video: 0=HRESBW, 1=HRES16COLOR",&n,0,1) !=0) return(-1);
if(IN0NUM("Mais qtas iteracoes",&N_ITER,0,1000) !=0) return(-1);
if(n==0)_setvideomode(_HRESBW);
      else _setvideomode(_HRES16COLOR);
_clearscreen(_GCLEARSCREEN);
/* como normalizar em caso de tabela ? */
pi=4.0*atan(1.0); LYPIX= LXPIX=fabs(lpixel); LSAMP=1.0; ldiag=sqrt(LXPIX*LXPIX+LYPIX*LYPIX);
if(lpixel <0) EMISSAO=nview; else EMISSAO=1;
NJ=(ldiag*1.0+LSAMP)/LSAMP+1;
/* if(!TABELA)( nrowH=nview*nsamp; ncolH=nlin*ncol;
      if( (pfile=fopen(arq,"r+b"))==NULL) ( perror("EM: Erro p/ abrir arq.");
          return(-1);
      )
      janelaHC0=janelaHC2]=0; janelaHC1]=ncolH; janelaHC3]=nrowH;
)
*/
/* Obtendo a contagem media */
MAT_XMMABS(proj,(unsigned)nview*nsamp,&x,&y,&z,&yp,&vari);
y=4*nview*nsamp;
dia_cursor(YMSG-7,1): MSGTIM("Inicio do EM=");
dia_cursor(YMSG-5,40): printf("Total (proj)      Total(reconstr)");
dia_cursor(YMSG-4,1):      printf("Contag.total inicial(pr)=%10.1f".y);
dia_cursor(YMSG-3,1):      printf("Likelihood :");

if (ITER0==0)
(vero=4;      /* nlin=ncol */
y=vero;      /* chute inicial e colocando 0 fora do circulo */
for(i=0; i<nlin; i++)
for(j=0; j<ncol; j++) ( x=(i-nlin/2)*1.0*(i-nlin/2)+(j-ncol/2)*1.0*(j-ncol/2);
      if(x > (nlin/4.0*ncol)) (*{rec_lim+i*ncol+i}=0.0);
          else (*{rec_lim+ i*ncol+i}=4);
      )
)
/* reescalando chute inicial .....*/

```

```

y=0.0;
for( i=0; i<nlin; i++)
  for(j=0; j<ncol; j++) y=y+*(rec_im+ i*ncol+j));
vero=vero/y;
for(i=0; i<nlin; i++)
  for(j=0; j<ncol; j++) *(rec_im+ i*ncol+j)=(*(rec_im+ i*ncol+j))*vero;
}
/*      fim da otimizacao do EM .....*/

CONVERGIU=0;      n=ITER0;  like_ant=0.0;
janela[0]=janela[2]=10;  janela[1]=ncol;  janela[3]=nlin;
dia_cursor(YMSG,1);  printf("EM: reprojando (0.0,0.0)",nview-1,nsamp-1);
dia_cursor(YMSG+1,1);  printf("EM: estimando pix.(0.0,0.0)",nlin-1,ncol-1);
if (N_ITER > 0)
  do(                                     /* loop */
    PC_MATRDISP(_HRESBW,rec_im, janela,zoom);
    dia_cursor(YMSG-1,1);  printf("iteracao:%2d",n);
    n=n+1;
    /* reprojacao */
/*    likelihood=0.0;
    reprojacao (rec_im,nlin,ncol,(double)lpxel,
                pbuf3, nview, nsamp, proj, tabela_psf,ndist,nano,
                &w,&distancia_p, &likelihood) ;
*/
    w=0.0;  likelihood=0.0;  cte=ndist*2.0/(LSAMP+ldiag);
    for(i=0; i<nview; i++)
      (if(TABELA) (seno=sin(i*pi/nview); coseno=cos(i*pi/nview);cosxx=coseno*cte);
        for(j=0; j<nsamp; j++)
          (yp=(proj+i*nsamp+j);
            dia_cursor(YMSG,40);
            printf("%3d %3d",i,j);
            z=0.0;
            tproj=(j- nsamp/2)*LSAMP;
            rowH=i*nsamp+j;
            if(!TABELA) (if(DISC_LER_VET(pfile.H.4,rowH,janelaH)!=0) return(-1);
                          )
            if(i!=(nview/2)) d_c=LSAMP/fabs(coseno)/LXPPIX+2;
            for(l=0; l<nlin; l++)
              (y=(1-nlin/2)*LXPPIX;
                ji=1*ncol;
                uy=(y*seno-tproj)*cte;
                if(i==nview/2) (ci=0; cs=ncol;)
                  else (x=(tproj-y*seno)/coseno;
                        c=x/LXPPIX+ncol/2;
                        ci=c-d_c;  cs=c+d_c;
                        if((ci)=ncol)!(cs=0)) continue;
                        if(ci<0) ci=0;
                        if(cs>ncol) cs=ncol;
                        )
                for (c=ci;c<cs; c++)
                  (colH=ji+c;
                    if(TABELA)
                      ( x=(c-ncol/2)*LXPPIX;
                        k=fabs(x*cosxx + uy);
                        if(k)=ndist) continue ;
                        else hlc=(tabela_psf+k*nano +i);
                      )
                  else hlc=H[colH];
                  z=z+*(rec_im+ i*ncol+c))*hlc;
                )
          )
    )
  )

```

```

    )
    z=z/EMISSAO;
    *(pbuf3+i*nsamp+j)=z;
    w=w+z;
    if(z > EPSILON) vero=-z+up*log(z); else vero=0.0;
    likelihood=likelihood+vero;
  )
}

dia_cursor(YMSG-4,40); printf("%10.1f",w);
dia_cursor(YMSG-3,40); printf("%10.1f",likelihood);

/* estimando pixels */
w=0.0; vero=0.0;
for(l=0; l<nlin; l++)
  (y=(l-nlin/2)*LYPIX;
  for(c=0; c<ncol; c++)
    (if ((* (rec_im+l*ncol+c)) > EPSILON) /* pixel # 0 */
     (z=0.0; vero=0.0;
      colH=l*ncol+c;
      x=(c-ncol/2)*LXPIX;
      dia_cursor(YMSG+1,40);
      printf("%3d %3d",l,c);
      /* ler coluna 'colH' de H
      if(!TABELA)(if(DISC_LER_COL(pfile,H,4,colH,janelaH)!=0)return(-1);
      )
      */
      for(i=0; i<nview; i++)
        ( seno=sin(i*pi/nview); coseno=cos(i*pi/nview);
          tpix=x*coseno+y*seno;
          ji=tpix/LSAMP+nsamp/2-NJ; if(ji<0)ji=0;
          js=ji+2*NJ; if(js>nsamp) js=nsamp;
          for (j=ji;j<js; j++)
            (
              if(TABELA)
                (
                  tproj=(j-nsamp/2)*LSAMP;
                  k=fabs(tpix-tproj)*cte; /* ldiag=LSAMP */
                  if(k>ndist) continue; /* hlc=0 */
                  else hlc=*(tabela_psf+k*nanq+i);
                )
              /* else
                (
                  rowH=i*nsamp+j;
                  hlc=H[rowH];
                )
              */
              if (hlc < EPSILON) continue; /* skip */
              if ((* (pbuf3+i*nsamp+j))>EPSILON2) up=*(proj+i*nsamp+j)/((* (pbuf3+i*nsamp+j)));
              else ( up=0.0; /* erro */
                )
              vero=vero+hlc;
              z=z+up*hlc;
            )
          )
        x=(* (rec_im+l*ncol+c)) * z/vero;
        *(rec_im+l*ncol+c)=x;
        w=w+x;
      )
    )
}

```

```

    )
  )
  dia_cursor(YMSG+4,50); printf("X10.2f (Z2d)",w,n);
  if(fabs(likelihood-like_ant) < EPSILON) CONVERGIU=1;

  like_ant=likelihood;
  if(kbhit()) {
    dia_cursor(YMSG+3,XMSG);printf("Finalizar ?(s/n):");gets(ch); i=strlen(ch)-1;
    if ((ch[i]!='s')&&(ch[i]!='S')) CONVERGIU=1;
  }
  } while ( !CONVERGIU && (n<((ITER0+N_ITER)));

  dia_cursor(YMSG-7,40); MSGTIN("Fim do EX=");printf("n_iter=X3d",N_ITER);
  *ITERACA00=n+ITER0;
  BIRD_EYE(rec_im,nlin,ncol,10,100,64/nlin,192/ncol,90,20,1);
  MAT_HMMABS(rec_im,(unsigned)nlin*ncol,&x,&y,&z,&u,&v,&var i);
  dia_cursor(1,20); printf("min=X10.1f,med=X10.1f,max=X10.1f,tot=X10.1f",x,y,z,u*nlin*ncol);

  CONS_MENSAG("Ok. (ret)");
  _setvideomode(_DEFAULTMODE);
  if(!TABELA) fclose(pfile);
  return(0);
}

/*( ..... funcao .....*/
int ECT_PSF_TAB(float far *tabela, int NDIST, int NVIEW, double lxpix,double lypix,
               double faixa_raios,int NSUBDIV)

/* Rotina p/ gerar uma tabela p/ agilizar a obtencao do PSF em ECT
. tabela[NDIST][NVIEW]
. tabela[k][a] k corresp. a distancia=k*(faixa_raios+diagonal pixel)/2/NDIST
. a ang=a*pi/NVIEW
. a tabela contem os elementos # 0
. calcula a area(percentual) do pixel interceptado pela faixa de raios de largura 'faixa_raio'
. dimensoes do pixel ('lxpix, lypix')
. num. de vistas 'NVIEW'
. angulo varia de 0 a pi (tabela a=0 a NVIEW-1) e e' normal 'a 'faixa_raios' e comeca
e orientacao de acordo com sistema trigonometrico
. tabela=tabela(distancia, angulo)
. distancia= dist. do centro do pixel a linha central dos raios
. NDIST : num. de distancias tabeladas
. NSUBDIV : num. de subdivisao do pixel (1-0) p/ checkar interinidade
)*/
{ register int i,j;
  int a,k;
  float pi,t,s,s0,d_d,d_x,d_y,x,y,ldiaq,ang,dist,li,tm,ts;
  float seno,coseno,xx,EPSTLON6=1.0E-6;

  s0=lxpix*lypix/(NSUBDIV*NSUBDIV); ldiaq=sqrt(lxpix*lxpix+lypix*lypix);
  pi=4.0*atan(1.0);
  d_d=(faixa_raios+ldiaq)/2.0/NDIST; /* elemento de variacao p/ distancia */
  d_x=lxpix/NSUBDIV; d_y=lypix/NSUBDIV;
  dia_cursor(YMSG,XMSG);printf("#ang,#dist=0..Z2d,0..Z3d pixel=(Z5.1f,Z5.1f).bin=Z5.1f \n",
                               NVIEW-1,NDIST-1,lxpix,lypix,faixa_raios);
  xx=0.0;
  for (a=0; a<(NVIEW ;a++)

```

```
{ ang=a* pi/NVIEW;
  seno=sin(ang);      coseno=cos(ang);
  for (k=0; k<NDIST; k++)
  {
    dia_cursor(YMSG+1,XMSG);printf(" (%3d,%3d)",a,k);
    tm=k*d_d;
    ti=tm-faixa_raios/2;
    ts=tm+faixa_raios/2;
    s=0.0;
    for(i=0; i<NSUBDIV; i++)
      for (j=0; j<NSUBDIV; j++)
        { x=(j+0.5-NSUBDIV/2)*d_x;
          y=(i+0.5-NSUBDIV/2)*d_y;
          t=x*coseno+y*seno-EPSILON6;
          if((t<t) && (t<ts))s=s+s0;
        }
    *(tabela+k*NVIEW +a)=s/(l*pix*l*pix);
  }
/* check de consistencia na tabela (deve ser 1)*/
x=0.0;
for (k=-NDIST;k<=NDIST;k++) (i=k; if(i<0) i=-i*k;
                             x=x+(*(tabela+i*NVIEW+a));
                             )
xx=xx+x;
dia_cursor(YMSG+2,1);printf("ang=%3d soma(1.0)=%10.3f ,Acum(%3d)=%10.3f",a,x*d_d,NVIEW.xx*d_d);
} ~

return(0);
}
```

REFERÊNCIAS BIBLIOGRÁFICAS

- 1) ALPERT, N.M.; CHESLER, D.A.; CORREIA, J.A.; ACKERMAN, R.H.; CHANG, J.Y.; FINKLESTEIN, S.; DAVIS, S.M.; BROWNELL, G.L.; TAVERAS, J.M. Estimation of the local statistical noise in emission computed tomography. *IEEE Transactions on Medical Imaging*, New York, 1(2):142-6, Oct.1982.
- 2) ALTSCHULER, M.D.; CENSOR, Y.; HERMAN, G.T.; LENT, A.; LEWITT, R.M.; SRIHARI, S.N.; TUY, H.; UDUPA, JK. Mathematical aspects of image reconstruction from projections. In: KANAL, L.N.; ROSENFELD, A.,ed. *Progress in pattern recognition*. Amsterdam, North-Holland, 1981. p. 323-75.
- 3) ANSCOMBE, F.J. The transformation of Poisson, binomial and negative-binomial data. *Biometrika*, Cambridge,UK, 35:246-54, 1948.
- 4) ANDREWS, H.C.; HUNT, B.R. *Digital image restoration*. Englewood Cliffs, Prentice - Hall, 1977. 238p.
- 5) BRACEWELL, R.N. *The Fourier transform and its applications*, 2.ed. New York, McGraw Hill, 1986. 474p.
- 6) CENSOR, Y. Row-action methods for huge and sparse systems and their applications. *SIAM Review*, 23(4):444-66, Oct. 1981.
- 7) CENSOR, Y. Finite series : expansion reconstruction methods. *Proceedings of the IEEE*, New York, 71 (3): 409-19, Mar. 1983.
- 8) CHANG, L.T. A method for attenuation correction in radionuclide computed tomography. *IEEE Transactions on Nuclear Science*, New York, 25(2):638-42, Feb. 1978.
- 9) COHEN, A. *Biomedical signal processing*. Boca Raton, CRC Press, 1986. v.1, 167p.
- 10) DEMPSTER, A.P.; LAIRD, N.M.; RUBIN, D.B. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Ser. B*, 39:1-38, 1977.
- 11) DOUGHERTY, E.R.; GIARDINA, C.R. *Image processing: continuous to discrete*. Englewood Cliffs, Prentice-Hall, 1987. v.1, 452p.
- 12) DUFF, I.S. A survey of sparse matrix research . *Proceedings of the IEEE*, 65: 500-35, 1977.



- 13) ENDO, M.; IINUMA, T.A. Software correction of scatter coincidence in positron CT. *European Journal of Nuclear Medicine*, 9:391-6, 1984.
- 14) ENDO, M.; IINUMA, T.A.; TANAKA, E.; TATENO, Y. Effect of correction function on image characteristics of Positologica: a positron CT device for the head. *European Journal of Nuclear Medicine*, 7:130-6, 1982
- 15) FURUIE, S.S.; TOMITANI, T. Agilização na reconstrução de imagens utilizando método direto de Fourier e array processor. In: SIMPÓSIO BRASILEIRO DE COMPUTAÇÃO GRÁFICA E PROCESSAMENTO DE IMAGENS,2, Águas de Lindóia, 1989. Anais. São José dos Campos, SBC/INPE, 1989. p469-78.
- 16) GOODMAN, J.W.; BELSHER, J.F. Fundamental limitations in linear invariant restoration of atmospherically degraded images. *Proceedings of SPIE*, 75:141-54, 1976.
- 17) HEBERT, T.; LEAHY, R. A generalized EM algorithm for 3-D Bayesian reconstruction from Poisson data using Gibbs priors. *IEEE Transactions on Medical Imaging*, New York, 8(2):194-202, June 1989.
- 18) HERMAN, G.T. *Image reconstruction from projections: the fundamentals of computerized tomography*. New York ,Academic Press, 1980. 316p.
- 19) HOUNSFIELD, G.N. Computerized transverse axial scanning (tomography) I: description of system. *Brit. J. Radiol.* 46:1016-22, 1973.
- 20) HUESMAN, R.H. The effects of a finite number of projection angles and finite lateral sampling of projections on the propagation of statistical errors in transverse section reconstruction. *Phys.Med.Biol*, 22(3):511-21, 1977.
- 21) HUESMAN, R.H.; GULLBERG, G.T.; GREENBERG, W.L.; BUDINGER, T.F. *RECLBL library users manual*. Berkeley, Lawrence Berkeley Laboratory of the University of California, 1977. (Pub 214) 283p.
- 22) INOUYE, T. Square root transform for the analysis of quantum fluctuations in spectrum data. *Nuclear Instruments and Methods*, Amsterdam, 91:581-84, 1971.
- 23) JIANG, S.S. *Image restoration and speckle suppression using the noise updating repeated Wiener filter*. Los Angeles, 1986. 144p. Tese (Ph.D) - University of Southern California.

- 24) JIANG, S.S.; SAWCHUK, A.A. Noise updating repeated Wiener filter and other adaptive noise smoothing filters using local statistics. *Applied Optics*, New York, 25(14):2326-37, July 1986.
- 25) JOHNSON, R.A.; WICHERN, D.W. *Applied multivariate statistical analysis*. Englewood Cliffs, Prentice-Hall, 1982. 593p.
- 26) KASTURI, R.; WALKUP J.F. Nonlinear image restoration in signal-dependent noise. In: *Advances in Computer Vision and image processing*, JAI press , 1986. v.2, chap.4, p.167-212.
- 27) KATSAGGELOS, A.K. Iterative image restoration algorithms. *Optical Engineering*, 28(7):735-48, July 1989.
- 28) KLOTZ, H.; SNYDER, D.L. A hardware architecture using finite-field arithmetic for computing maximum-likelihood estimation in Emission Tomography. *IEEE Transaction on Medical Imaging*, New York, 7(4):279-90, Dec. 1988.
- 29) KUAN, D.T. *Nonstationary recursive restoration of images with signal-dependent noise with application to speckle reduction*. Los Angeles, 1982. 231p. Tese (Ph.D) - University of Southern California.
- 30) KUAN, D.T.; SAWCHUK, A.A.; STRAND, T.C.; CHAVEL, P. Adaptive noise smoothing filter for image with signal-dependent noise. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, New York, 7(2):165-177, Mar.1985.
- 31) LAGENDIJK, R.L.; BIEMOND, J.; BOEKEE, D.E. Regularized iterative image restoration with ringing reduction. *IEEE Transactions on Acoustic, Speech and Signal Processing*, New York, 36(12): 1874-88, Dec. 1988.
- 32) LANGE, K.; CARSON, R. EM reconstruction algorithm for emission and transmission tomography. *Journal of Computer Assisted Tomography*, New York, 8 (2):306-16, Apr. 1984.
- 33) LEWITT, R.M. *Reconstruction algorithms: transform methods*. *Proceedings of the IEEE*, New York, 71( 3): 390-408, Mar. 1983.
- 34) LO, C.M. *Estimation of image signals with Poisson noise*. Los Angeles, 1979. 185p. Tese (Ph.D.) - University of Southern California.
- 35) LO, C.M.; SAWCHUK, A.A. Non linear restoration of filtered images with Poisson noise. *Proceedings of SPIE* , Bellingham, 207:84-95, 1979

- 36) LUENBERGER, D.G. Optimization by vector space methods, New York ,John Wiley , 1969. 325p.
- 37) MAEDA, J.; MURATA, K. Digital restoration of scintigraphic images by a two-step procedure. IEEE Transactions on Medical Imaging, 6 (4): 320-24, Dec. 1987.
- 38) MASCARENHAS, N.D.A.; FERNANDES, L.F.V. New methods for picture reconstruction: recursive and causal techniques. IEEE Transactions on Pattern Analysis and Machine Intelligence, New York, 2(4):369-76, July 1980.
- 39) MASCARENHAS, N.D.A.; VELASCO, F.R.D. Processamento digital de imagens. São Paulo, Instituto de Matemática e Estatística da USP, 1984. 333p. ESCOLA DE COMPUTAÇÃO,4, São Paulo, 1984.
- 40) MILLER, T.R.; ROLLINS,E.S. A practical method of image enhancement by interactive digital filtering. The Journal of Nuclear Medicine, 26(9):1075-80, Sept.1985.
- 41) MILLER, T.R.; SAMPATHKUMARAN, K.S. Design and application of finite impulse response digital filters. Eur.J.Nucl.Med. 7:22-7, 1982
- 42) NUNES, P.R.R.L. Estimation algorithms for medical imaging including joint attenuation and emission reconstruction. 134p. Tese (Ph.D.)-Stanford University, 1980.
- 43) OPPENHEIM, B.E.; APPLIEDORN, C.R. Single-photon emission computed tomography. In: GELFAND, M.J.; THOMAS S.R. Effective use of computers in nuclear medicine. New York, McGraw-Hill, 1988. Chap.2, p.31-74, 604p.
- 44) PRIESTLEY, M.B. Spectral analysis and time series. London, Academic Press, 1981. 890p.
- 45) RABBANI, M.E. Bayesian filtering of Poisson noise using local statistics. IEEE Transactions on Acoustic,Speech, and Signal Processing, New York, 36(6):933-7, June 1988.
- 46) RABINER, L.R.; GOLD, B. Theory and application of digital signal processing. Englewood Cliffs, Prentice-Hall, 1975. 762p.
- 47) ROCKMORE, A.J.; MACOVSKI, A. A maximum likelihood approach to emission image reconstruction from projections. IEEE Transactions on Nuclear Sciences, New York, 23(1):1428-32, Aug. 1976.
- 48) SCHWARTZ, M.; SHAW, L. Signal processing: discrete spectral analysis, detection, and

- estimation. Tokyo, McGraw-Hill, 1975. 396p.
- 49) SHEPP, L.A.; LOGAN, B.F. The Fourier reconstruction of a head section. *IEEE Transactions on Nuclear Science*, New York, 21(6):21-43, June 1974.
- 50) SHEPP, L.A.; VARDI, Y. Maximum likelihood reconstruction for emission tomography. *IEEE Transactions on Medical Imaging*, New York, 1(2):113-22, Oct. 1982.
- 51) SHIM, Y.S.; CHO, Z.H. SVD pseudoinversion image reconstruction. *IEEE Transactions on Acoustics, Speech and signal processing*, New York, 29 (4): 904-9, Aug. 1981.
- 52) STARK, H.; WOODS, J.W.; PAUL I.; HINGORANI, R. Direct Fourier reconstruction in computer tomography. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, New York, 29(2): 237-45, Apr.1981.
- 53) STARK, H. *Image recovery : theory and application*. Orlando, Academic Press, 1987, 543p.
- 54) TANAKA, E. A fast reconstruction algorithm for stationary positron emission tomography based on a modified E.M. Algorithm. *IEEE Transactions on Medical Imaging*, New York, 6 (2): 98-105, June 1987.
- 55) TRUSSEL, H.T.; HUNT, B.R. Sectioned methods for image restoration. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, New York, 26( ):157-64, 1978.
- 56) TRUSSEL, H.J. A priori knowledge in algebraic reconstruction methods. In: *Advances in Computer Vision and Image Processing*, JAI Press, 1984. v.1, Chap. 6, p. 265 - 316.
- 57) TODD-POKROPEK, A. Image processing in nuclear medicine. *IEEE Transactions on Nuclear Science*, 27(3):1080-94, June 1980.
- 58) TOMITANI, T. An edge detection algorithm for attenuation correction in emission CT. *IEEE Transactions on Nuclear Science*, New York, 34(1):309-12, Feb. 1987.
- 59) VAN TREES, H.L. *Estimation and modulation theory*, New York, John Wiley, 1968, part I.
- 60) VARDI, Y.; SHEPP, L.A.; KAUFMAN, L. A statistical model for positron emission tomography. *Journal of the American Statistical Association*, Boston, 80(389): 8 - 37, Mar. 1985.
- 61) WEBB, S. SPECT reconstruction by simulated annealing. *Phys. Med. Biol.*, 34(3):259-81, 1989.