

Reconstrução tomográfica em SPECT: aspectos gerais

Silvia De Francesco, Paulo J. S. G. Ferreira

Resumo – Este artigo é o primeiro de dois trabalhos dedicados à reconstrução tomográfica, e em especial à modalidade de SPECT. Neles abordamos certas classes de algoritmos de reconstrução que são especialmente úteis no contexto da tomografia. Neste primeiro artigo discutem-se aspectos gerais e computacionais do problema de reconstrução tomográfica. Deixaremos para o segundo artigo o estudo do problema da correcção da atenuação em SPECT.

Abstract – This paper is the first of two works devoted to the study of certain reconstruction algorithms which find application in computerized tomography. In this first paper we will study general aspects and computational issues in reference to tomographic reconstruction. The follow up paper will be exclusively concerned with the problem of attenuation in SPECT and its correction.

Palavras chave – Tomografia, algoritmos de reconstrução, SPECT, equações lineares, métodos iterativos.

I. INTRODUÇÃO

Existe grande interesse na determinação da distribuição bi-dimensional (ou tri-dimensional) de um determinado parâmetro ou grandeza física no interior de um dado objecto. Esse conhecimento pode ajudar a entender as propriedades físicas dos materiais constituintes do objecto (ou tecidos, no caso do corpo humano). No caso da tomografia de emissão (SPECT e PET) estuda-se com frequência a concentração de um radiofármaco.

A reconstrução tomográfica consiste na determinação da distribuição espacial do parâmetro em causa a partir de um número finito de medições, frequentemente denominadas projecções. Uma vez determinada, a distribuição espacial pode ser examinada num sistema de visualização de imagem.

A estrutura deste artigo é a seguinte. Na secção II, o problema da reconstrução é inicialmente formulado e resolvido no espaço contínuo. Introduce-se o conceito de projecção, e menciona-se o teorema da secção central e as fórmulas analíticas de inversão. Na secção III discute-se aspectos relacionados com a discretização e a implementação numérica dos algoritmos. Consideram-se aspectos específicos relativos ao método directo de Fourier, à retroprojecção filtrada e aos métodos algébricos. Estes últimos conduzem a equações lineares de dimensão normalmente elevada, pelo que se referem alguns dos métodos iterativos ou semi-iterativos que podem ser usados com vantagem na solução destas equações. É o caso dos métodos de Jacobi e Gauss-Seidel, com e sem relaxação, dos métodos de Kaczmarz e das projecções alternadas, e do método dos gradientes conjugados, entre outros. Salientamos que, dada a grande variedade de algoritmos de reconstrução existen-

tes não pretendemos fazer uma análise exaustiva, mas apenas tentar fornecer uma panorâmica sobre o vasto leque de abordagens possíveis.

Os problemas específicos relacionados com a correcção da atenuação em SPECT serão abordados em [1].

II. FORMULAÇÃO DO PROBLEMA NO ESPAÇO CONTÍNUO

O problema matemático da determinação de uma função bidimensional a partir de um conjunto de integrais de linha foi formulado e resolvido pela primeira vez em 1917 pelo matemático Radon, embora, em 1963, também o físico nuclear Cormack tenha desenvolvido a sua própria solução [2], pela qual recebeu o prémio Nobel da Fisiologia e Medicina em 1979.

Muitos problemas de reconstrução equivalem à determinação de uma função f a partir de um conjunto de funcionais de f , e constituem exemplos de *problemas inversos*. Uma das questões essenciais relacionadas com este tipo de problemas é a sua classificação em problemas *bem* ou *mal* colocados. Diz-se que o problema está bem posto se a solução existe e é única, e se depende de forma contínua dos dados.

Esta questão tem grande importância. Quando o problema está mal posto, e mesmo quando a solução existe e é única, surge a questão da sua estabilidade face a pequenas variações nos dados fornecidos pelos funcionais. A existência de erros nas medições efectuadas, uma deficiente modelação do processo de aquisição, ou ainda, a própria discretização do problema associada à implementação numérica, podem ser suficientes para afastar a solução da função original.

Voltaremos a esta questão mais tarde, quando discutirmos métodos numéricos. Na secção seguinte, após a definição de integrais de linha e da relação dos mesmos com as projecções de um objecto, relação estabelecida pelo teorema da secção central, são derivadas fórmulas analíticas para a solução do problema inverso.

A. Integrais de linha e projecções

A distribuição de um determinado parâmetro na secção de um objecto (figura 1) pode ser modelada por uma função bidimensional $f(x, y)$. Por integral de linha designamos um integral da função $f(x, y)$ calculado ao longo de uma linha no plano XOY , e por isso uma função dos parâmetros θ e t .

A equação da recta AB na figura 1 é $x \cos \theta + y \sin \theta = t$, e o integral de linha $p_\theta(t)$ ao longo de AB é dado por

$$p_\theta(t) = \int_{\text{linha}(\theta, t)} f(x, y) ds, \quad (1)$$

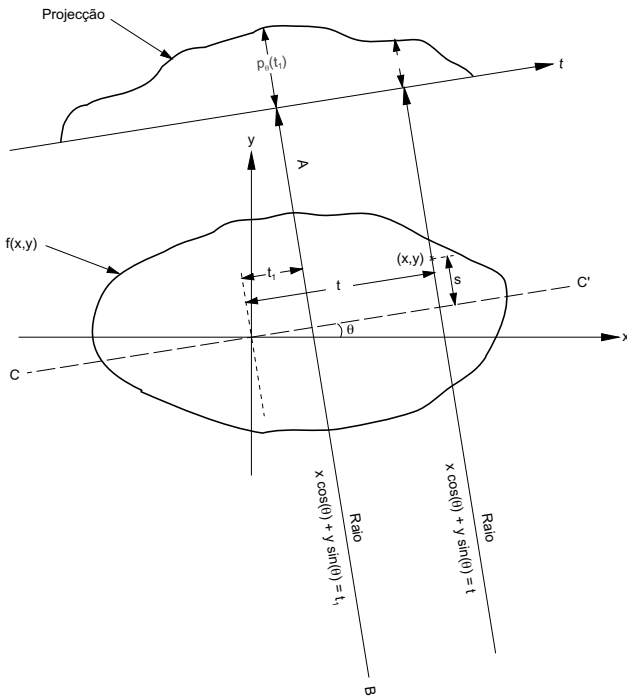


Figura 1 - Um objecto $f(x, y)$ e as suas projecções $p_\theta(t)$.

ou, utilizando a símbolo δ de Dirac,

$$p_\theta(t) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \delta(x \cos \theta + y \sin \theta - t) dx dy,$$

fórmula conhecida por transformada de Radon de $f(x, y)$.

Uma projecção é um conjunto de integrais de linha obtidos segundo uma determinada geometria. Na geometria paralela os integrais são obtidos ao longo de linhas paralelas correspondendo a um determinado ângulo θ (figura 2.a). Na geometria divergente (*fan beam*) os integrais são medidos ao longo de linhas que formam um feixe divergente (figura 2.b). Os algoritmos que descreveremos aplicam-se ao caso das projecções paralelas, podendo sempre ser modificados e adaptados ao caso de projecções divergentes.

B. Teorema da secção central

O teorema da secção central (*Fourier slice theorem*) estabelece a ligação entre a transformada de Fourier bidimensional $F(u, v)$ da função $f(x, y)$ e as transformadas de Fourier unidimensionais, $P_\theta(\omega)$, das projecções $p_\theta(t)$.

Apresenta-se em seguida a demonstração [3] deste teorema para o caso de projecções paralelas.

A relação entre as coordenadas de um ponto com respeito a um referencial TOL obtido rodando de um ângulo θ o sistema XOY é dada por

$$\begin{bmatrix} t \\ l \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}. \quad (2)$$

No sistema de coordenadas TOL , o integral de linha (1) reduz-se a

$$p_\theta(t) = \int_{-\infty}^{\infty} f(t, l) dl \quad (3)$$

e a sua transformada de Fourier é

$$P_\theta(\omega) = \int_{-\infty}^{\infty} p_\theta(t) e^{-j2\pi\omega t} dt.$$

Substituindo,

$$P_\theta(\omega) = \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} f(t, l) dl \right] e^{-j2\pi\omega t} dt.$$

Voltando ao sistema de coordenadas XOY e utilizando a relação (2), obtém-se

$$P_\theta(\omega) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-j2\pi\omega(x \cos \theta + y \sin \theta)} dx dy.$$

O termo da direita representa a transformada de Fourier bidimensional da função $f(x, y)$ à frequência espacial ($u = \omega \cos \theta$, $v = \omega \sin \theta$), de modo que

$$P_\theta(\omega) = F(\omega, \theta) = F(\omega \cos \theta, \omega \sin \theta). \quad (4)$$

É esta a relação entre as transformadas de Fourier das projecções e a transformada de Fourier bidimensional da função $f(x, y)$.

O teorema da secção central afirma que a transformada de Fourier de uma projecção paralela da imagem $f(x, y)$ segundo um determinado ângulo θ equivale à transformada de Fourier bidimensional $F(u, v)$, ao longo de uma recta formando um ângulo θ com o eixo u , e passando pela origem no espaço das frequências.

A transformada de Fourier de $p_\theta(t)$ fornece os valores de $F(u, v)$ ao longo da linha BB na figura 3.

C. Fórmulas analíticas de inversão

As fórmulas analíticas de inversão pressupõem um conhecimento completo das projecções. Não são por isso directamente aplicáveis na prática, já que os processos de medida fornecem apenas estimativas para as projecções e em conjuntos finitos de amostras. O interesse destas fórmulas deriva do facto de elas servirem como base a uma classe muito importante de algoritmos, obtidos por discretização de expressões analíticas, e de permitirem deduzir características gerais que as estimativas da função f apresentam, independentemente do tipo de algoritmo de reconstrução utilizado.

C.1 Método directo de Fourier

O teorema da secção central mostra que calculando as projecções da função $f(x, y)$ correspondentes aos ângulos $\theta_1, \theta_2, \dots, \theta_k$, e calculando a transformada de Fourier de cada projecção, podemos determinar os valores de $F(u, v)$ ao longo de linhas radiais (fig. 4).

Na hipótese de se poder adquirir um número infinito de projecções, $F(u, v)$ seria conhecido em todo o plano UOV e a função $f(x, y)$ poderia ser reconstruída utilizando a transformada inversa de Fourier:

$$f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u, v) e^{j2\pi(ux+vy)} du dv. \quad (5)$$

Na prática dispomos de um número finito de projecções, de modo que a função $F(u, v)$ é conhecida somente ao longo de um número finito de linhas radiais. Portanto, para poder aplicar a fórmula analítica de inversão torna-se necessária a interpolação dos pontos radiais para os pontos duma grelha quadrada.

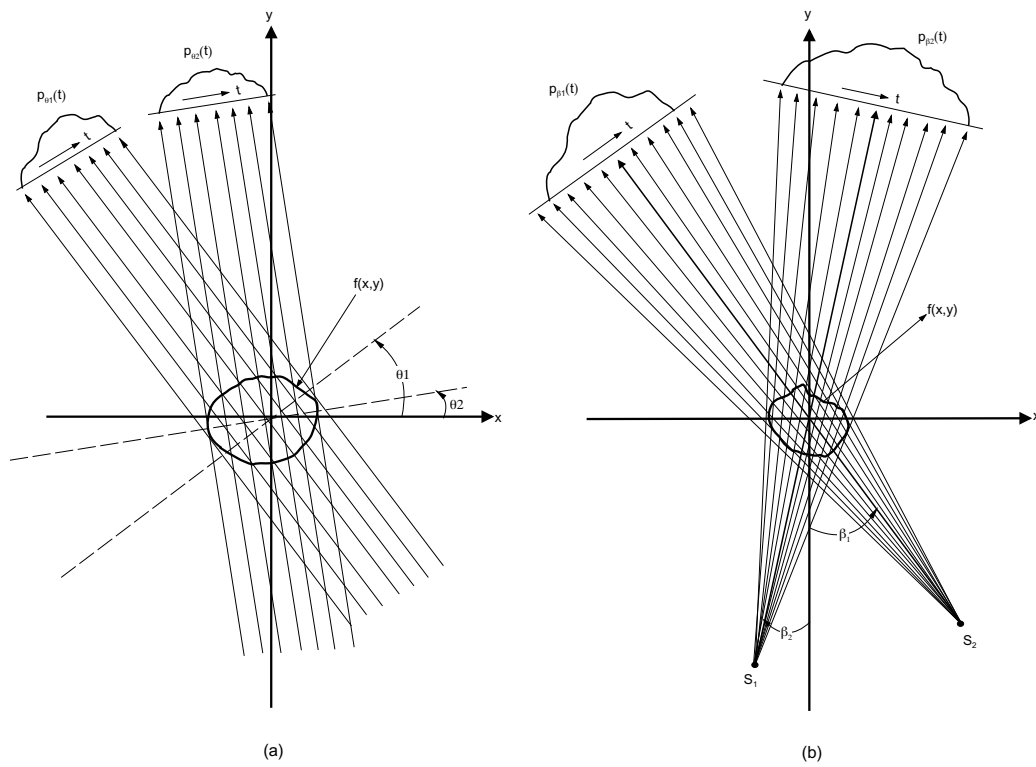


Figura 2 - Projecções paralelas (a) e divergentes (b).

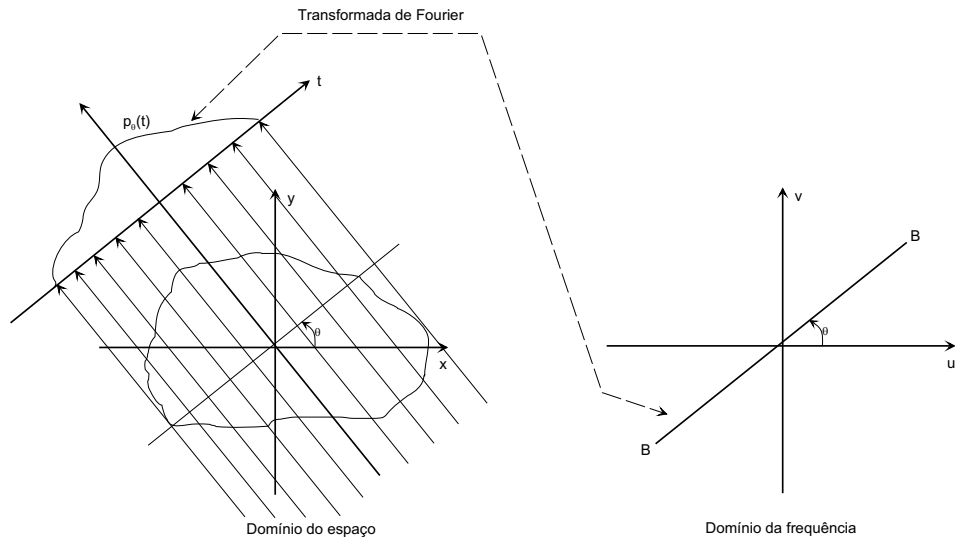


Figura 3 - O teorema da secção central.

C.2 Retroprojecção filtrada

O algoritmo de reconstrução mais conhecido, habitualmente designado por retroprojecção filtrada (*Filtered Back-projection*), é justificável com base na fórmula de inversão analítica.

Consideramos a expressão (5) para o cálculo analítico da função $f(x, y)$. Se passamos a representar o domínio da frequência em coordenadas polares (ω, θ) , e se efectuarmos as substituições:

$$u = \omega \cos \theta, \quad v = \omega \sin \theta$$

e

$$du dv = \omega d\omega d\theta,$$

a equação (5) transforma-se em:

$$f(x, y) = \int_0^{2\pi} \int_0^{\infty} F(\omega, \theta) e^{j2\pi\omega(x \cos \theta + y \sin \theta)} \omega d\omega d\theta. \quad (6)$$

Separando a integração segundo a variável θ nos intervalos $(0, \pi)$ e $(\pi, 2\pi)$, e depois utilizando a propriedade $F(\omega, \theta +$

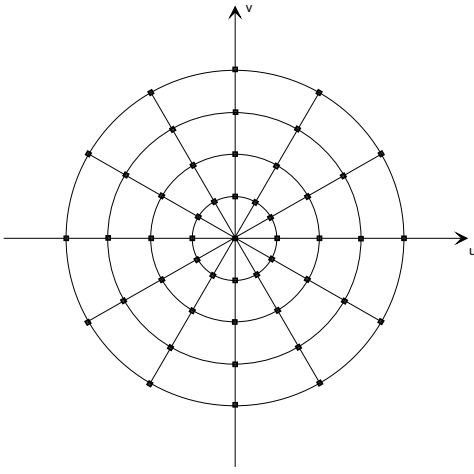


Figura 4 - O domínio das frequências.

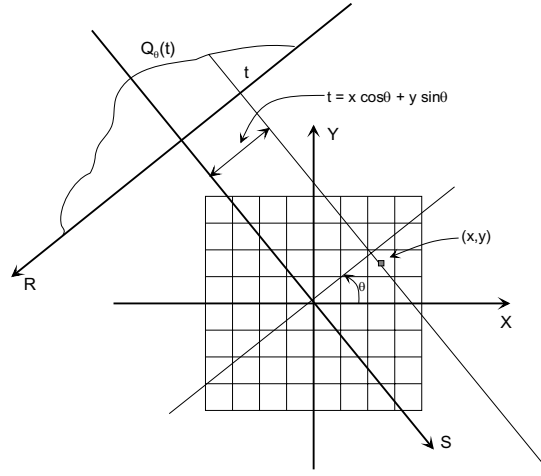


Figura 5 - Representação gráfica da retroprojectão.

$\pi) = F(-\omega, \theta)$, a equação (6) pode ser assim reformulada:

$$f(x, y) = \int_0^\pi \left[\int_{-\infty}^{\infty} F(\omega, \theta) |\omega| e^{j2\pi\omega t} d\omega \right] d\theta, \quad (7)$$

onde $t = x \cos \theta + y \sin \theta$. Pelo teorema da secção central podemos substituir $F(\omega, \theta)$ por $P_\theta(\omega)$, obtendo

$$f(x, y) = \int_0^\pi \left[\int_{-\infty}^{\infty} P_\theta(\omega) |\omega| e^{j2\pi\omega t} d\omega \right] d\theta.$$

Este integral pode escrever-se como

$$f(x, y) = \int_0^\pi Q_\theta(x \cos \theta + y \sin \theta) d\theta, \quad (8)$$

onde

$$Q_\theta(t) = \int_{-\infty}^{\infty} P_\theta(\omega) |\omega| e^{j2\pi\omega t} d\omega, \quad (9)$$

que permite o cálculo de $f(x, y)$ a partir das transformadas das projecções.

A equação (9) representa uma operação de filtragem (com resposta em frequência $|\omega|$), razão pela qual o termo $Q_\theta(t)$ é chamado “projecção filtrada”.

Para obter a função $f(x, y)$ é necessário integrar as projecções filtradas segundo θ , operação designada “retroprojectão”. Para percebermos o significado desta operação, consideramos a figura 5. A cada ponto (x, y) do plano da imagem, para cada projecção θ , corresponde um valor $t = x \cos \theta + y \sin \theta$ e a projecção filtrada Q_θ contribui para a reconstrução no ponto (x, y) com o seu valor em t . Para cada ângulo de projecção θ , o valor de t é o mesmo para todos os pontos ao longo da linha LM, de modo que a projecção filtrada Q_θ dará a mesma contribuição à reconstrução de todos estes pontos, pelo que podemos dizer que ela é retroprojectada no plano da imagem.

Em suma, o algoritmo de retroprojectão filtrada pode ser subdividido em duas fases distintas:

- Cálculo das projecções filtradas.
- Cálculo do integral de 0 e π (retroprojectão).

III. RECONSTRUÇÃO TOMOGRÁFICA NO ESPAÇO DISCRETO

Na secção anterior estudou-se o problema da reconstrução tomográfica no espaço contínuo, sem ter em conta as limitações impostas pelos sistemas físicos de aquisição das projecções, nem a necessária discretização associada à implementação numérica dos algoritmos.

Quando passamos a considerar o problema no espaço discreto, temos que ter em conta o facto de que os dados fornecidos pelo sistema de aquisição serem, na realidade, discretizados e sujeitos a ruído. Os dados de que se dispõe constituem uma estimativa das projecções, amostradas em $M \times N$ pontos, e designadas por $p_{\theta_n}(m\Delta t)$.

A partir destes dados, o algoritmo de reconstrução discreto deverá fornecer uma estimativa da função $f(x, y)$, amostrada em $I \times J$ pontos e designada por $f_r(i\Delta x, j\Delta y)$.

A. Algoritmos baseados em transformações

Em [4] encontra-se uma descrição de alguns destes algoritmos. Mencionamos apenas os dois talvez mais conhecidos e utilizados na prática. O primeiro é o método directo de Fourier, aplicável só no caso de projecções paralelas, mas que fornece uma solução simples e directa do problema. O segundo é o algoritmo das retroprojectões filtradas, que desempenha um papel central na reconstrução tomográfica, particularmente em CT.

A.1 Método directo de Fourier

Como vimos, no espaço contínuo (secção II-C.1), as duas equações

$$P_\theta(\omega) = F(\omega, \theta) = F(\omega \cos \theta, \omega \sin \theta)$$

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

constituem uma solução do problema inverso. O que se torna necessário na prática é calcular a transformada de

Fourier das projecções, obtendo uma estimativa da transformada de Fourier de f , e por aplicação da transformada inversa bidimensional de Fourier reconstruir a própria função f . Mas na implementação numérica destas equações surge uma dificuldade, a incompatibilidade do referencial polar e cartesiano, que requer uma fase intermédia de interpolação.

Para percebermos melhor o problema consideremos em maior detalhe o processo de discretização e, para tornar mais simples as fórmulas, admitimos que o número de amostras por projecção, que designaremos por M , seja ímpar. Admitiremos que no domínio das frequências o ponto (θ_0, ω_0) ocupa a origem do referencial. O cálculo numérico da transformada de Fourier das diversas projecções conduz à utilização da DFT, ou seja,

$$P_{\theta_n}(\omega_l) = \frac{1}{M} \sum_{m=-\frac{M-1}{2}}^{\frac{M-1}{2}} p_{\theta_n}(m\Delta t) e^{-j2\pi m \frac{l}{M}}, \quad (10)$$

para $l = -\frac{M-1}{2}, \dots, 0, \dots, \frac{M-1}{2}$. Uma vez calculada para todos os N ângulos θ_n , esta fórmula fornece uma estimativa, F_r , do espectro em frequência da função f num conjunto de pontos radialmente dispostos no espaço de frequência.

O cálculo da transformada inversa de Fourier bidimensional da função F_r , por sua vez, implica a utilização da IDFT,

$$\begin{aligned} f_r(i\Delta x, j\Delta y) &= f_r(i, j) = \\ &= \sum_{h=-\frac{H-1}{2}}^{\frac{H-1}{2}} \sum_{k=-\frac{H-1}{2}}^{\frac{H-1}{2}} F_r(h, k) e^{j2\pi \frac{ih+jk}{H}}. \end{aligned}$$

Precisa-se assim de estimativas para os valores da função F_r em $H \times H$ pontos localizados sobre uma grelha quadrada, podendo recorrer-se a diversos tipos de interpolação (por exemplo, interpolação bilinear — ver [5] para mais detalhes).

Longe do centro diminui a densidade de pontos radiais e aumenta o erro de interpolação, o que significa erro maior nas componentes de alta relativamente às de baixa frequência. Torna-se portanto conveniente aumentar “artificialmente” a densidade de pontos sobre o referencial polar, antes de se iniciar a interpolação. Esta operação é normalmente efectuada por introdução de zeros antes da realização das DFTs (*zero padding*).

Em suma, o método directo de Fourier é constituído pelas três etapas seguintes:

- Transformada de Fourier das projecções (com eventual *zero padding*).
- Interpolação.
- Transformada bidimensional inversa de Fourier.

A.2 Retroprojecção filtrada

Como vimos na secção II-C.2, a primeira fase deste algoritmo consiste no cálculo das projecções filtradas (9), podendo este ser efectuada no domínio da frequência em três fases:

- transformada de Fourier das projecções,
- multiplicação pela função de transferência do filtro ($|\omega|$),

- transformada inversa de Fourier.

Passando à implementação numérica do algoritmo, o cálculo da transformada de Fourier das projecções é efectuado com a DFT (equação (10)). Antes disso, para minimizar o efeito da interferência entre símbolos devida à convolução periódica no domínio do tempo, torna-se necessário aumentar o número de pontos sobre os quais calcular a DFT (*zero padding*).

Devido à natureza da função que desejamos reconstruir, as projecções $p_{\theta_n}(t)$ têm, em princípio, banda ilimitada, o que implica a sobreposição (*aliasing*) dos espectros na função amostrada $p_{\theta_n}(m\Delta t)$. Este efeito pode ser minimizado limitando em banda as projecções. Assim, na implementação numérica do segundo passo, a função de transferência do filtro a aplicar será:

$$H(\omega) = |\omega|W(\omega).$$

Existem várias possibilidades para a função janela, $W(\omega)$, algumas das quais são mencionadas em [6].

Por último, a transformada inversa realiza-se utilizando um algoritmo tipo IDFT, ou seja, implementando a fórmula:

$$Q_{\theta_n}(t_k) = \sum_{l=-\frac{M-1}{2}}^{\frac{M-1}{2}} P_{\theta_n}(\omega_l) H(\omega_l) e^{j2\pi \frac{\omega_l t_k}{M}}$$

que pode ser obtida a partir da equação (9), substituindo $|\omega|$ com $H(\omega)$ e discretizando.

A segunda fase do algoritmo consiste no cálculo do integral de retroprojecção (8) que, no espaço discreto, pode ser assim reformulado:

$$f_r(i\Delta x, j\Delta y) = \Delta\theta \sum_{n=1}^N Q_{\theta_n}(i\Delta x \cos \theta_n + j\Delta y \sin \theta_n). \quad (11)$$

Pode verificar-se que para um dado ponto $(i\Delta x, j\Delta y)$ e um dado ângulo θ_n , o argumento $(i\Delta x \cos \theta_n + j\Delta y \sin \theta_n)$ não corresponda a um ponto t_k em que o valor da função $Q_{\theta_n}(t_k)$ é conhecido. A interpolação mais utilizada para o efeito é a interpolação linear. Apresentando a interpolação um considerável peso computacional, às vezes faz-se uma pré-interpolação da função Q_{θ} , utilizando depois o critério do vizinho mais próximo para obter o valor de Q_{θ_n} em cada ponto $(i\Delta x \cos \theta_n + j\Delta y \sin \theta_n)$. Um método comum para efectuar a pré-interpolação consiste em fazer um *zero padding* das projecções no domínio da frequência antes de calcular a IDFT.

A implementação numérica do algoritmo das retroprojecções filtradas é constituída pela sequência de operações :

- *Zero padding* das projecções discretizadas.
- DFT.
- Multiplicação por $H(\omega)$.
- *Zero padding* para pré-interpolação.
- IDFT.
- Cálculo do integral de retroprojecção.

B. Algoritmos algébricos

Nos algoritmos de reconstrução algébricos o problema é discretizado logo a partida, sendo reduzido a um sistema de equações lineares.

Para a definição do modelo matemático, a função f é substituída por uma versão discretizada espacialmente sobre uma grelha cartesiana que se supõe cobrir toda a região de interesse no plano. A imagem por reconstruir é subdividida em N *pixels*, cada um dos quais é referenciado por um único parâmetro j . O valor da função discretizada no elemento j , igual ao valor médio da função f no seu interior, será representado por x_j .

Designando por y_i ($i = 1, 2, \dots, M$) os integrais de linha de f ao longo dos M raios projecção, podemos representar o processo de projecção pelas equações lineares

$$\sum_{j=1}^N a_{ij} x_j = y_i \quad i = 1, 2, \dots, M \quad (12)$$

onde os x_j representam os elementos desconhecidos da imagem por reconstruir e os a_{ij} (designados por *pesos*) representam a contribuição do *pixel* j do objecto para o elemento da projecção i , descrevendo o modelo físico de aquisição de dados em causa.

Em tomografia de transmissão os valores a_{ij} correspondem ao valor do comprimento (ou área, se se levar em conta a dimensão transversal do feixe) de intersecção do *pixel* j com o caminho i . No entanto, em tomografia de emissão simples os valores a_{ij} correspondem ao valor do comprimento (ou área) da intersecção do *pixel* j com o caminho i , multiplicado por um factor exponencial que descreve a atenuação sofrida pelos fotões emitidos a partir do centro do *pixel* j na direcção i .

O cálculo da matriz dos pesos, constituindo parte integrante de qualquer método algébrico, é uma tarefa que merece ser analisada com cuidado. Conhecida a geometria do sistema é possível calcular os pesos a_{ij} . Contudo, o número de pesos é elevado e há quem prefira calculá-los em *run time*, ainda que efectuando aproximações.

O sistema (12) pode ser representado em forma matricial,

$$Ax = y \quad (13)$$

sendo o vector N -dimensional x designado por *vector de imagem*, o vector M -dimensional y por *vector de medida* e a matriz ($M \times N$) A por *matriz de projecção*.

A matriz A apresenta características particulares:

- Grandes dimensões: é habitual M e N serem na ordem das dezenas ou centenas de milhares, o que faz com que o número de elementos de A seja superior a 10^8 .
- Esparsa: só uma pequena percentagem de elementos de A ($\leq 1\%$) é não nula, pois são poucos os *pixels* intersectados por um dado raio de projecção.
- Pouco estruturada: a distribuição interna da matriz A é muito irregular.

As grandes dimensões e a pouca estruturação tornam difícil a resolução numérica do sistema por métodos directos. Quando a dimensão do sistema o permite (SPECT), pode

tentar recorrer-se à decomposição em valores singulares e à construção de uma generalização da matriz inversa [7].

Nalguns casos, devido à falta de informação, o sistema é subdeterminado ($M < N$), apresentando uma infinidade de soluções. Noutros casos é sobredeterminado ($M > N$) e, devido ao ruído e aos erros no processo de medida, torna-se inconsistente. Não existindo em geral solução única ou exacta, o problema pode formular-se como um problema de optimização: dada a matriz de projecção A e o vector de medida y , procura-se obter uma estimativa x da função f que satisfaça um ou mais critérios previamente definidos.

Existem diversos algoritmos iterativos para a resolução deste género de problemas, permitindo gerar, a partir de uma aproximação inicial $x^{(0)}$, uma sucessão vectorial de aproximações, $x^{(1)}, x^{(2)}, \dots, x^{(k)}$, que converge para uma solução x do sistema quando $k \rightarrow +\infty$ [8–10].

Alguns destes algoritmos requerem que A seja uma matriz quadrada. Neste caso, se A for rectangular, será necessário transformar previamente o sistema. A transformação consiste na pré-multiplicação de ambos os membros da equação (13) pela transposta da matriz A , obtendo

$$A^T A x = C x = A^T y = b.$$

A matriz C resultante desta operação é quadrada e de dimensão N de modo que se o sistema $Ax = b$ for sobredeterminado, o novo sistema $Cx = z$ terá um número inferior de equações relativamente ao sistema inicial. No entanto, se $Ax = b$ for subdeterminado o novo sistema terá um número superior de equações. A eventual redução das dimensões do sistema nem sempre é um facto positivo, dado que muitas vezes a matriz resultante é mal condicionada.

B.1 Métodos de Jacobi e Gauss-Seidel

Alguns métodos iterativos, de entre os quais o de Jacobi e o de Gauss-Seidel, começam com a transformação do sistema $Ax = y$ num sistema equivalente da forma:

$$x = Tx + c. \quad (14)$$

Após a escolha do vector inicial $x^{(0)}$, a sequência de soluções aproximadas é gerada calculando:

$$x^{(k+1)} = Tx^{(k)} + c, \quad k = 0, 1, \dots \quad (15)$$

Por exemplo: a matriz A (quadrada) pode ser decomposta da seguinte forma:

$$A = D - L - U$$

onde D é a matriz diagonal cuja diagonal principal é a mesma da matriz A , $-L$ é a submatriz triangular inferior de A , e $-U$ é a sua submatriz triangular superior. A equação $Ax = (D - L - U)x = y$ é então transformada em:

$$Dx = (L + U)x + y,$$

e ainda, se nenhum dos elementos da diagonal principal de A for nulo,

$$x = D^{-1}(L + U)x + D^{-1}y.$$

Os métodos de Jacobi e de Gauss-Seidel desenvolvem-se a partir desta igualdade. Uma iteração do método de Jacobi fica definida por

$$x^{(k+1)} = D^{-1}(L + U)x^{(k)} + D^{-1}y,$$

onde:

$$\begin{aligned} [L + U]_{ij} &= \begin{cases} -a_{ij} & \text{se } i \neq j \\ 0 & \text{se } i = j \end{cases} \\ [D^{-1}]_{ii} &= \frac{1}{a_{ii}} \\ [D^{-1}(L + U)]_{ij} &= \begin{cases} 0 & \text{se } i = j \\ \frac{-a_{ij}}{a_{ii}} & \text{se } i \neq j \end{cases}. \end{aligned}$$

Cada $x_i^{(k)}$ pode ser calculado recorrendo à fórmula

$$x_i^{(k+1)} = \frac{y_i - \sum_{j=1, j \neq i}^N a_{ij}x_j^{(k)}}{a_{ii}}, \quad i = 1, 2, \dots, N.$$

O método de Gauss-Seidel, ao contrário do que acontece no método de Jacobi, utiliza na iteração $k + 1$ os valores já calculados do vector $x^{(k+1)}$, sendo cada iteração definida por

$$x^{(k+1)} = D^{-1}Lx^{(k+1)} + D^{-1}Ux^{(k)} + D^{-1}y,$$

ou, segundo a forma da equação (15):

$$x^{(k+1)} = (D - L)^{-1}Ux^{(k)} + (D - L)^{-1}y. \quad (16)$$

Se $1 < i < N$, cada $x_i^{(k+1)}$ pode ser calculado por meio da fórmula:

$$x_i^{(k+1)} = \frac{y_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^N a_{ij}x_j^{(k)}}{a_{ii}},$$

tendo-se ainda

$$x_i^{(k+1)} = \frac{y_i - \sum_{j=2}^N a_{ij}x_j^{(k)}}{a_{ii}},$$

caso $i = 1$, ou

$$x_i^{(k+1)} = \frac{y_i - \sum_{j=1}^{N-1} a_{ij}x_j^{(k+1)}}{a_{ii}},$$

caso $i = N$.

B.2 Métodos com relaxação, JOR e SOR

Pode melhorar-se a velocidade de convergência dos métodos baseados na iteração da equação (14) introduzindo nos mesmos o termo de “relaxação”. Obtem-se assim os métodos JOR (*Jacobi Over-Relaxation*) e SOR (*Successive Over-Relaxation*).

A ideia é transformar o sistema (14) no equivalente

$$x = (1 - \mu + \mu)x = (1 - \mu)x + \mu x = (1 - \mu)x + \mu T x + \mu c$$

de onde vem

$$x^{(k+1)} = [\mu T + (1 - \mu)I]x^{(k)} + \mu c.$$

O escalar μ é a constante de relaxação, e os valores próprios λ_N da nova matriz de iteração $N = \mu T + (1 - \mu)I$ são relacionados com os da matriz T , λ_T , segundo a fórmula

$$\lambda_N - 1 = \mu(\lambda_T - 1).$$

É possível determinar o valor óptimo da constante de relaxação [10], [11].

O método JOR obtém-se por introdução do termo de relaxação no método de Jacobi, e exprime-se por

$$\begin{aligned} x^{(k+1)} &= (1 - \mu)x^{(k)} + \mu(D^{-1}(L + U)x^{(k)} + D^{-1}y) \\ &= [\mu D^{-1}(L + U) + (1 - \mu)I]x^{(k)} + \mu D^{-1}y, \end{aligned}$$

onde, escolhendo oportunamente o valor de μ , a matriz de iteração $[\mu D^{-1}(L + U) + (1 - \mu)I]$ terá raio espectral inferior ao da matriz do método de Jacobi.

O método SOR é o método de Gauss-Seidel (eq. (16)) com relaxação, definindo-se através de

$$x^{(k+1)} = [\mu(D - L)^{-1}U + (1 - \mu)I]x^{(k)} + \mu(D - L)^{-1}y.$$

B.3 Métodos algébricos, ART

Os algoritmos de tipo ART (*Algebraic Reconstruction Techniques*), introduzidos por Gordon e outros em 1970 [3], têm como base comum o método das projecções de Kaczmarz para resolução de sistemas de equações lineares e podem ser aplicados a sistemas em que o número de equações é diferente do número de incógnitas.

Após escolha da aproximação inicial $x^{(0)}$, estes métodos produzem uma sequência de aproximações à solução do problema. Para preceber a ideia que está na base deste algoritmo descrevemos o método das projecções de Kaczmarz começando pelo simples caso de um sistema de duas equações em duas incógnitas:

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 &= y_1 \\ a_{21}x_1 + a_{22}x_2 &= y_2. \end{aligned}$$

Cada equação representa uma recta no plano \mathbb{R}^2 e a solução do sistema (se existir e for única) é dada pelo ponto de intersecção entre as duas rectas (fig. 6).

Intuitivamente, olhando para a interpretação geométrica, vemos que se partirmos de um ponto qualquer do plano e projectarmos ortogonal e alternadamente numa recta e noutra, o ponto projectado tende para a solução x (fig. 6), sendo tanto mais rápida a convergência quanto maior for o menor dos ângulos entre as duas rectas. É este o princípio sobre o qual se baseia o método das projecções de Kaczmarz.

No caso da reconstrução tomográfica o sistema $Ax = y$ é um sistema de grandes dimensões constituído por M

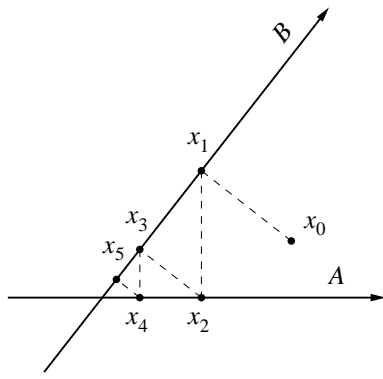


Figura 6 - Representação geométrica de um sistema de duas equações lineares independentes a duas incógnitas e convergência do método de Kaczmarz.

equações em N incógnitas. Sendo a_i a transposta da i -ésima linha da matriz A , o sistema pode ser representado pelas equações

$$\langle a_i, x \rangle = y_i \quad i = 1, \dots, M \quad (17)$$

onde y_i é o i -ésimo elemento do vector y . Cada uma destas equações representa um hiperplano H_i em \mathbb{R}^N . Assumindo que a intersecção entre os hiperplanos seja constituída por um conjunto de pontos não vazio, este conjunto representa as soluções de $Ax = y$.

Uma vez escolhido o ponto $x^{(0)} \in \mathbb{R}^N$ como aproximação inicial, a primeira aproximação ($x^{(1)}$) será dada pela sua projecção sobre o hiperplano H_1 definido pela primeira das equações do sistema, a segunda pela projecção de $x^{(1)}$ sobre o hiperplano H_2 , e assim sucessivamente.

Para formular o algoritmo é necessário encontrar uma expressão explícita para a projecção de um ponto N -dimensional p sobre um hiperplano H . Em termos formais procuramos o ponto $p' \in H$ tal que

$$\inf_{x \in H} \|p - x\| = \|p - p'\|.$$

A função auxiliar

$$L(x) = \|p - x\|^2 + \lambda(\langle a_i, x \rangle - y_i)$$

tem o mínimo precisamente no ponto p' . Sendo a derivada de $L(x)$ a função $\nabla L = 2(x - p) + \lambda a_i$, pondo $\nabla L = 0$ obtém-se

$$x = p - \frac{\lambda}{2} a_i. \quad (18)$$

Fazendo o produto interno e tendo em conta a equação (17) obtém-se [12]:

$$\frac{\lambda}{2} = \frac{\langle p, a_i \rangle - y_i}{\langle a_i, a_i \rangle},$$

de onde, substituindo em (18):

$$x = p - \frac{\langle p, a_i \rangle - y_i}{\langle a_i, a_i \rangle} a_i.$$

Portanto, a expressão explícita para a projecção $P_i p$ do ponto p no hiperplano H_i é dada pela fórmula:

$$P_i p = p + \frac{y_i - \langle p, a_i \rangle}{\langle a_i, a_i \rangle} a_i, \quad (19)$$

O método de Kaczmarz para a solução de $Ax = y$ pode ser resumido pela expressão

$$x^{(k)} = P_k \dots P_2 P_1 x^{(0)}.$$

A sequência dos hiperplanos sobre os quais se vai projectar pode não coincidir com a ordem das equações. Pelo contrário, a relação geométrica entre os vários raios projecção sugere que as linhas consecutivas correspondem hiperplanos quase paralelos. Logo, se a sequência escolhida tivesse a mesma ordem das equações do sistema a convergência do método poderia ser lenta. Normalmente, sendo impossível determinar a sequência óptima, a escolha da sequência é efectuada empiricamente com base nas características geométricas do sistema físico de aquisição dos dados.

Podem apresentar-se três situações diferentes:

- O sistema tem solução única. Neste caso a sequência de aproximações converge para a solução.
- O sistema tem um número infinito de soluções. Neste caso a sequência de aproximações converge para uma das soluções dependendo da aproximação inicial e da ordem seguida na escolha dos hiperplanos sobre os quais efectuar a projecção.
- O sistema não tem solução. Neste caso a sequência de aproximações oscilará entre um certo número de pontos soluções de subsistemas do sistema $Ax = y$, fornecendo soluções sub-óptimas dependendo da aproximação inicial e da ordem seguida na escolha dos hiperplanos.

Os vários métodos do tipo ART distinguem-se pela maneira de calcular e de aplicar o termo de correcção. Por exemplo, no método MART (*Multiplicative Algebraic Reconstruction Technique*) o factor de correcção é multiplicativo em vez de aditivo. A ordem pela qual as equações do sistema são tomadas tem influência no resultado.

No método SIRT (*Simultaneous Iterative Reconstructive Technique*) as correcções são calculadas utilizando a fórmula (19) mas a actualização é efectuada só no fim de um ciclo completo de projecções sobre os M hiperplanos. A correcção a aplicar é a média das correcções calculadas ao longo do ciclo. Na variante por blocos do método ART (*block ART*), as equações são divididas em grupos e a actualização do vector x é efectuada após o processamento de cada grupo, usando-se também a média das correcções calculadas no grupo.

B.4 Métodos de optimização

Em geral, as técnicas baseadas em optimização podem apenas ser aplicadas quando a matriz do sistema é simétrica e definida positiva. Para poderem ser utilizados na resolução do sistema $Ax = y$ (onde A é uma matriz esparsa, não simétrica e por vezes não quadrada) é necessário operar uma transformação no sistema.

Como a matriz $C = A^T A$ é simétrica e não-negativa definida, mesmo que A o não seja, pode tentar-se resolver o sistema

$$A^T A x = C x = A^T y = b$$

cuja solução é a mesma do sistema $Ax = y$. A forma quadrática na qual se baseiam estes métodos é uma função $\mathbb{R}^N \rightarrow \mathbb{R}$ dada por

$$f(x) = \frac{1}{2}x^T Cx - b^T x + c,$$

onde c é um escalar constante. Se a matriz C for simétrica e definida positiva, $f(x)$ é minimizada pela solução de $Cx = b$, pelo que a minimização de $f(x)$ e a resolução de $Cx = b$ são problemas equivalentes.

Dois conceitos que serão utilizados de seguida são os de erro e de residual. O erro é definido por $e^{(i)} = x^{(i)} - x$, onde $x^{(i)}$ é a aproximação da solução x na i -ésima iteração de um qualquer método iterativo. O residual, definido por $r^{(i)} = b - Cx^{(i)}$, é uma medida de erro relativamente ao vector dos termos independentes, sendo importante salientar a relação $r^{(i)} = -Ce^{(i)}$ com o erro e , ainda mais, a relação $r^{(i)} = -f'(x)$ com o gradiente da forma quadrática, ou seja, com o vector:

$$f'(x) = \begin{bmatrix} \frac{\partial}{\partial x_1} f(x) \\ \frac{\partial}{\partial x_2} f(x) \\ \vdots \\ \frac{\partial}{\partial x_N} f(x) \end{bmatrix} \quad (20)$$

apontando, em qualquer ponto do espaço \mathbb{R}^N , na direcção de maior crescimento de $f(x)$.

A partir de um ponto inicial $x^{(0)}$, estes métodos (nomeadamente: *Steepest Descent*, Direcções Conjugadas e Gradientes Conjugados) procuram iterativamente o ponto de mínimo de $f(x)$. A direcção e o tamanho de cada passo são determinados de várias formas dependendo do método.

O método chamado *Steepest Descent* utiliza o vector em cuja direcção $f(x)$ decresce mais depressa (a direcção oposta à do gradiente $f'(x)$, ou seja a do resíduo). A equação que relaciona uma aproximação $x^{(i+1)}$ com a aproximação anterior $x^{(i)}$ será, portanto:

$$x^{(i+1)} = x^{(i)} + \alpha^{(i)} r^{(i)}. \quad (21)$$

O comprimento $\alpha^{(i)}$ do passo a dar determina-se procurando o ponto de mínimo de $f(x)$ na direcção $r^{(i)}$, calculando e igualando a zero a derivada de $f(x^{(i+1)})$ em função de α :

$$\frac{d}{d\alpha} f(x^{(i+1)}) = f'(x^{(i+1)})^T \frac{d}{d\alpha} x^{(i+1)} = 0. \quad (22)$$

Utilizando a definição de resíduo e a fórmula (21), obtém-se a equação:

$$r^{(i+1)T} r^{(i)} = 0$$

donde se vê que no *Steepest Descent* cada gradiente é ortogonal ao gradiente anterior, logo a aproximação $x^{(i)}$ progride aos zigue-zagues.

Substituindo $r^{(i+1)}$ pela sua definição, e operando suces-

sivas transformações, obtém-se a expressão para $\alpha^{(i)}$:

$$\begin{aligned} (b - Cx^{(i+1)})^T r^{(i)} &= 0 \\ \Downarrow \\ [b - C(x^{(i)} + \alpha^{(i)} r^{(i)})]^T r^{(i)} &= 0 \\ \Downarrow \\ [(b - Cx^{(i)}) - \alpha^{(i)} Cr^{(i)}]^T r^{(i)} &= 0 \quad (23) \\ \Downarrow \\ [r^{(i)} - \alpha^{(i)} Cr^{(i)}]^T r^{(i)} &= 0 \\ \Downarrow \\ \alpha^{(i)} &= \frac{r^{(i)T} r^{(i)}}{r^{(i)T} Cr^{(i)}}. \end{aligned}$$

Assim, o algoritmo *Steepest Descent* pode ser resumido pelas três equações:

$$\begin{aligned} r^{(i)} &= b - Cx^{(i)} \\ \alpha^{(i)} &= \frac{r^{(i)T} r^{(i)}}{r^{(i)T} Cr^{(i)}} \\ x^{(i+1)} &= x^{(i)} + \alpha^{(i)} r^{(i)}. \end{aligned}$$

O facto de que no *Steepest Descent* a aproximação progrida aos zigue-zagues implica a oscilação da progressão segundo algumas das coordenadas, no entanto, seria preferível que não se verificassem oscilações e que, uma vez dado um passo segundo uma direcção, não se voltasse atrás segundo essa mesma direcção.

A ideia que está na base do método das Direcções Conjugadas é mesmo aquela de, partindo de um conjunto de direcções ortogonais d_0, d_1, \dots, d_{N-1} , dar um passo em cada uma destas direcções, deixando de ter a oscilação característica do *Steepest Descent*. Ao fim de N passos dados segundo a equação:

$$x^{(i+1)} = x^{(i)} + \alpha^{(i)} d_i = x^{(0)} + \sum_{j=0}^i \alpha^{(j)} d_j$$

encontrar-se-á a solução.

Para que não se volte atrás segundo a direcção d_i , o novo erro deve ser ortogonal a esta direcção, ou seja:

$$d_i^T e^{(i+1)} = d_i^T (e^{(i)} + \alpha^{(i)} d_i) = 0$$

donde

$$\alpha^{(i)} = -\frac{d_i^T e^{(i)}}{d_i^T d_i}. \quad (24)$$

É necessário evidenciar o facto de que é impossível calcular $\alpha^{(i)}$ segundo a fórmula (24) dado que $e^{(i)}$ não é conhecido (se assim fosse o problema estaria resolvido), portanto este método deverá sofrer alterações. Considerando as direcções C -ortogonais (ou seja, "conjugadas", tais que $d_i^T C d_j = 0 \Leftrightarrow i \neq j$) a nova condição para que não se volte atrás segundo a direcção d_i , é que o novo erro seja C -ortogonal a esta direcção, ou seja:

$$d_i^T C e^{(i+1)} = d_i^T C (e^{(i)} + \alpha^{(i)} d_i) = 0$$

donde

$$\alpha^{(i)} = -\frac{d_i^T C e^{(i)}}{d_i^T C d_i} = \frac{d_i^T r^{(i)}}{d_i^T C d_i}, \quad (25)$$

sendo este o mesmo valor que se obtém ao minimizar $f(x)$ ao longo da direcção d_i .

A complexidade do algoritmo, em termos de número de multiplicações vector-matriz, pode ser reduzida utilizando, para o cálculo dos $r^{(i)}$ a fórmula recursiva:

$$\begin{aligned} r^{(i+1)} &= -C e^{(i+1)} = -C(e^{(i)} + \alpha^{(i)} d_i) \\ &= r^{(i)} - \alpha^{(i)} C d_i. \end{aligned} \quad (26)$$

Para demonstrar que o método atinge a solução em N passos, considere-se o vector erro $e^{(0)}$. As N direcções C -ortogonais, sendo linearmente independentes [9], [11], constituem uma base em \mathbb{R}^N , portanto, $e^{(0)}$ (como qualquer outro vector em \mathbb{R}^N) pode ser expresso como combinação linear das N direcções:

$$e^{(0)} = x^{(0)} - x = \sum_{j=0}^{N-1} \delta_j d_j.$$

Para determinar os coeficientes δ_i , basta pré-multiplicar ambos os membros desta expressão por $d_i^T C$, obtendo:

$$\begin{aligned} d_i^T C e^{(0)} &= d_i^T C \sum_{j=0}^{N-1} \delta_j d_j = d_i^T C (\delta_i d_i) \\ &\Downarrow \\ \delta_i &= \frac{d_i^T C e^{(0)}}{d_i^T C d_i}. \end{aligned}$$

Por outro lado, dado que:

$$e^{(i)} = e^{(i-1)} + \alpha^{(i-1)} d_{i-1} = e^{(0)} + \sum_{j=0}^{i-1} \alpha^{(j)} d_j$$

e que, pela C -ortogonalidade, $d_i^T C d_j = 0 \Leftrightarrow i \neq j$, a fórmula para o cálculo dos δ_i pode ser assim transformada:

$$\delta_i = \frac{d_i^T C e^{(i)}}{d_i^T C d_i}$$

onde se vê que $\alpha^{(i)} = -\delta_i$. Assim, o processo de construção de x , componente a componente, pode ser visto como o processo de redução do erro, componente a componente, dado que:

$$e^{(i)} = e^{(0)} + \sum_{j=0}^{i-1} \alpha^{(j)} d_j = \sum_{j=0}^{N-1} \delta_j d_j - \sum_{j=0}^{i-1} \delta_j d_j = \sum_{j=i}^{N-1} \delta_j d_j$$

no fim de N iterações cada componente do erro é eliminada, sendo $e^{(N)} = 0$ e $x^{(N)} = x$.

Para poder aplicar o algoritmo descrito será necessário, antes que mais, gerar o conjunto das N direcções C -ortogonais, a partir de um conjunto de N vectores linearmente independentes u_0, \dots, u_{N-1} , sendo habitualmente utilizado para o efeito o processo de ortogonalização (\equiv conjugação) de Gram-Schmidt [13]. O processo permite construir cada d_i a partir do u_i correspondente, por

eliminação das componentes não C -ortogonais aos vectores d já calculados. Por outras palavras, uma vez posto $d_0 = u_0$, para cada $i > 0$:

$$d_i = u_i + \sum_{k=0}^{i-1} \beta_{ik} d_k$$

onde a expressão para o cálculo dos β_{ik} pode ser obtida a partir da consideração da C -ortogonalidade de d_i com as direcções já calculadas ($i > j$):

$$d_i^T C d_j = u_i^T C d_j + \sum_{k=0}^{i-1} \beta_{ik} d_k^T C d_j \quad (27)$$

\Downarrow

$$0 = u_i^T C d_j + \beta_{ij} d_j^T C d_j \quad (28)$$

\Downarrow

$$\beta_{ij} = -\frac{u_i^T C d_j}{d_j^T C d_j}. \quad (29)$$

Sendo o cálculo dos β_{ik} muito pesado em termos de memória e de complexidade ($\mathcal{O}(N^3)$ operações) o método das Direcções Conjugadas com a ortogonalização de Gram-Schmidt não foi muito utilizado até o advento do método dos Gradientes Conjugados.

O método dos Gradientes Conjugados é um caso particular do método das Direcções Conjugadas em que as direcções de aproximação são geradas por conjugação dos resíduos (ou seja, $u_i = r^{(i)}$). Esta escolha revela propriedades que permitem simplificar enormemente o cálculo dos β_{ik} , e tornar o método muito mais eficiente.

Sendo as direcções de aproximação construídas a partir dos resíduos, o subespaço gerado por $\{d_0, d_1, \dots, d_{i-1}\}$ é o mesmo do gerado por $\{r^{(0)}, r^{(1)}, \dots, r^{(i-1)}\}$, de maneira que, pela sua ortogonalidade com as anteriores direcções de procura, o resíduo é também ortogonal aos resíduos anteriores, sendo:

$$r^{(i)T} r^{(j)} = 0, \quad i \neq j.$$

A expressão simplificada para o cálculo dos β_{ik} , pode ser obtida tomando em consideração o produto interno entre $r^{(i)}$ e $r^{(i+1)}$ (este último expresso segundo a fórmula recursiva (26)):

$$r^{(i)T} r^{(j+1)} = r^{(i)T} r^{(j)} - \alpha^{(j)} r^{(i)T} C d_j$$

de onde se pode deduzir:

$$\begin{aligned} \alpha^{(j)} r^{(i)T} C d_j &= r^{(i)T} r^{(j)} - r^{(i)T} r^{(j+1)} \\ &\Downarrow \\ r^{(i)T} C d_j &= \begin{cases} \frac{1}{\alpha^{(i)}} r^{(i)T} r^{(i)} & i = j \\ -\frac{1}{\alpha^{(i-1)}} r^{(i)T} r^{(i)} & i = j + 1 \\ 0 & \text{outros casos.} \end{cases} \end{aligned}$$

Sendo os β_{ij} definidos só para $i > j$, obtém-se:

$$\beta_{ij} = \begin{cases} \frac{1}{\alpha^{(i-1)}} \frac{r^{(i)T} r^{(i)}}{d_{i-1}^T C d_{i-1}} & i = j + 1 \\ 0 & i > j + 1. \end{cases} \quad (30)$$

Comparando (29) com (30) pode-se notar como, na segunda, desapareceram todos os coeficientes β_{ij} excepto um. Simplificando (considerando (25)):

$$\beta_{i,i-1} = \beta_i = \frac{r^{(i)T} r^{(i)}}{d_{i-1}^T r^{(i-1)}}.$$

Como na expressão

$$d_i^T r^{(i)} = u_i^T r^{(i)} + \sum_{k=0}^{i-1} \beta_{ik} d_k^T r^{(i)} \quad (31)$$

os coeficientes β_{ik} são todos nulos, obtém-se

$$d_i^T r^{(i)} = u_i^T r^{(i)} = r^{(i)T} r^{(i)},$$

o que permite chegar à seguinte expressão simples para o cálculo dos β_i :

$$\beta_i = \frac{r^{(i)T} r^{(i)}}{r^{(i-1)T} r^{(i-1)}}.$$

Resumindo, o método dos Gradientes Conjugados pode ser esquematizado nas equações:

$$\begin{aligned} d_0 &= r^{(0)} = b - Cx^{(0)}, \\ \alpha^{(i)} &= \frac{r^{(i)T} r^{(i)}}{d_i^T C d_i}, \\ x^{(i+1)} &= x^{(i)} + \alpha^{(i)} d_i, \\ r^{(i+1)} &= r^{(i)} - \alpha^{(i)} C d_i, \\ \beta_{i+1} &= \frac{r^{(i+1)T} r^{(i+1)}}{r^{(i)T} r^{(i)}}, \\ d_{i+1} &= r^{(i+1)} + \beta_{i+1} d_i. \end{aligned}$$

Teoricamente, o algoritmo dos Gradientes Conjugados atinge a solução na N -ésima iteração mas, devido aos erros de aproximação, perde-se gradualmente a C-ortogonalidade entre as direcções de procura, de modo que, na sua implementação numérica, o método não converge em N iterações. Contudo, a análise da convergência [13] permite confirmar a eficiência do método e sugere a sua utilização como método semi-iterativo, sendo utilizado como critério de paragem o descer da norma do resíduo abaixo de um determinado valor, o que pode até acontecer em menos de N iterações.

Hoje em dia este método, tomado como semi-iterativo, é bastante utilizado em problemas de grandes dimensões. Os problemas de tomografia têm essa característica, e a respectiva matriz C pode ser esparsa, propriedade que pode ser aproveitada para diminuir a complexidade do algoritmo.

A aplicação do método ao sistema de equações $A^T A x = A^T y$ pode conduzir a resultados numéricos insatisfatórios pelo facto de $A^T A$ ser mal condicionada. Van Dijke refere [14] que Paige e Saunders em 1982 propuseram uma variante do GC, chamada LSQR, aplicável a problemas de mínimos quadrados não simétricos e fornecendo, em caso de problemas mal condicionados, resultados melhores de que o algoritmo dos Gradientes Conjugados.

- [1] Silvia De Francesco and P. J. S. G. Ferreira. Reconstrução tomográfica em SPECT II: correcção da atenuação. *Revista do DETUA* (em preparação).
- [2] A. M. Cormack. Representation of a function by its line integrals, with some radiological applications. *Journal of Applied Physics*, 34:2722–2727, 1963.
- [3] A. C. Kak and M. Slaney. *Principles of Computerized Tomographic Imaging*. IEEE Press, 1988.
- [4] G. T. Herman. *Image reconstruction from projections: the fundamentals of computerized tomography*. Academic Press Inc., New York, 1980.
- [5] A. R. O. S. Borges. *Estruturas computacionais para reconstrução de imagem em tomografia*. Tese de doutoramento, Universidade de Aveiro, Departamento de Electrónica e Telecomunicações, 1994.
- [6] Silvia De Francesco. Técnicas tomográficas e algoritmos de reconstrução em tomografia. O caso de SPECT. Tese de Mestrado, Universidade de Aveiro, Departamento de Electrónica e Telecomunicações, 1997.
- [7] M. F. Smith, C. E. Floyd Jr., R. J. Jaszczak, and R. E. Coleman. Reconstruction of SPECT images using generalized matrix inverses. *IEEE Transactions on Medical Imaging*, 11:165–175, 1992.
- [8] A. E. Todd-Pokropek and M. A. Viergever. *Medical images: Formation, Handling and Evaluation*. Springer-Verlag, 1992.
- [9] G. H. Golub and C. F. Van Loan. *Matrix Computation*. The Johns Hopkins University Press, 1988.
- [10] D. M. Young. *Iterative solution of large linear systems*. Academic Press, 1971.
- [11] R. J. F. da Costa. Reconstrução de sinal e imagem: métodos de dimensão mínima no tempo e na frequência. Tese de mestrado, Universidade de Aveiro, Departamento de Electrónica e Telecomunicações, 1996.
- [12] P. J. S. G. Ferreira. Signal reconstruction: theory and algorithms. Departamento de Electrónica e Telecomunicações, Universidade de Aveiro (texto de Abril 1994).
- [13] J. R. Shewchuk. An introduction to the conjugate gradient method without the agonizing pain. Technical report, School of Computer Science, Carnegie Mellon University, Pittsburgh, Agosto 1994.
- [14] M. C. A. van Dijke, M. A. Viergever, and H. A. van der Vorst. Iterative methods for sparse, linear systems arising from tomographic image reconstruction: the relations between ART, SIRT and CG. Technical Report 51, Faculty of Technical Mathematics and Informatics, Delft University of Technology, The Netherlands, 1990.