

## APLICAÇÃO DE EFICIENTES TÉCNICAS COMPUTACIONAIS A PROBLEMAS DE TOMOGRAFIA SÍSMICA COM ONDAS SUPERFICIAIS

Newton Pereira dos Santos<sup>1</sup> e Jorge Luis de Souza<sup>2</sup>

Recebido em 29 julho, 2005 / Aceito em 11 janeiro, 2006  
Received on July 29, 2005 / Accepted on January 11, 2006

**ABSTRACT.** In order to save computational resources, a non-iterative algorithm based on both Horowitz-Sahni scheme (to manage sparse matrix) and the Ohio State University (OSU) algorithm (to compute matrix inversion) is proposed. The algorithm deals with least-squares solution characteristics, by structuring data for speeding up access. It is applied to the unsymmetric rectangular system matrix, which compaction improves both time and memory requirements up to 98% for only the matrix. Afterwards, symmetrical matrices are compressed and the OSU algorithm is modified in order to save additional demand in about 50%. In all cases, removal of redundant elements yields a significative reduction in access, and consequently, in the overall time and memory requirements. The algorithm was tested in a 2,000 Rayleigh wave paths dataset for various grid cell sizes, and the results were compared with both full storage and Singular Value Decomposition (SVD) techniques. Those tests showed that the economy varies from 65% to 98%. It was also applied in two real cases in Southeastern and Northeastern Brazil, using a standalone Personal Computer (PC), which demand was reduced by a factor of sixteen, with no loss of accuracy. Although the algorithm is intended to a specific problem in Seismology (surface wave tomography), it also allows to whichever is the sparse system solution, including the rigorous computation of resolution and covariance matrices.

**Keywords:** Group-velocity tomography, Damped least-squares, Sparse system, Full resolution and covariance matrices, Speeding up processing.

**RESUMO.** Com o objetivo de economizar recurso computacional, é proposto um algoritmo não-iterativo baseado nos esquemas de Horowitz-Sahni, para gerenciamento de matriz esparsa, e da Ohio State University, para inversão matricial. O algoritmo explora as características da solução por MMQ, estruturando os dados de forma a gerenciá-los eficientemente. Primeiramente, o esquema é aplicado à matriz dos coeficientes (retangular, não simétrica), cuja compactação pode reduzir a demanda em até 98%, na matriz. Posteriormente, matrizes simétricas são também comprimidas, e o algoritmo OSU é modificado, contribuindo para otimizar recursos em adicionais 50%. Em todos os casos, a remoção de elementos redundantes implica na redução nos tempos de acesso e memória exigida. A rotina foi submetida a testes de desempenho envolvendo 2.000 trajetórias de ondas Rayleigh, para várias dimensões de células da grade, cujos resultados foram comparados com as técnicas, tradicional e SVD, e comprovaram uma economia de recursos que varia entre 65% e 98%. O esquema foi, também, utilizado em dois casos reais, no Sudeste e Nordeste do Brasil, empregando-se, exclusivamente, um computador do tipo PC, cuja demanda foi reduzida em dezesseis vezes, sem perda de exatidão nos cálculos. Embora o algoritmo seja aplicado, neste contexto, a um problema específico na área de Sismologia (tomografia com ondas superficiais), ele pode ser usado em quaisquer ocorrências de sistema esparsa, sobretudo quando é exigido o cálculo rigoroso das matrizes resolução e covariância.

**Palavras-chave:** Tomografia com velocidade de grupo, MMQ amortecido, Sistema esparsa, Matrizes resolução e covariância completas, Agilização no processamento.

## INTRODUÇÃO

Embora o Método dos Mínimos Quadrados (MMQ) seja um notável instrumento para a determinação rigorosa das matrizes resolução e covariância, ele é computacionalmente ineficiente em sistemas esparsos de grande volume, se implementados por métodos tradicionais. Em muitos problemas geofísicos, a quantidade de informação envolvida pode exigir o emprego de processadores de grande capacidade de armazenamento para a solução por MMQ. Devido à natureza desses problemas, os sistemas de equação produzidos geralmente são esparsos (Horowitz & Sahni, 1976). Assim como ocorre em Geofísica, sistemas esparsos são, também, freqüentes em inúmeras outras atividades científicas, como, por exemplo, Astrofísica, Estudos de Modelos Oceânicos, Engenharia Química, Engenharia Estrutural, Cartografia, Simulação, Controle de Tráfego Aéreo, e Estatística, para citar algumas (<http://math.nist.gov/MatrixMarket/data/Harwell-Boeing/>). Geralmente, ocorre que nem todas as grandezas físicas observadas se correlacionam estatisticamente com todos os parâmetros incógnitos do modelo, isto é, cada observação vincula-se apenas a uma parcela dos parâmetros. Assim, uma parte reduzida da informação é efetivamente armazenada por linha (ou coluna) na matriz do sistema e o restante é indicado por “zeros”, que traduzem a ausência de correlação entre aquela grandeza e os demais parâmetros. Tal fato dá origem a um sistema redundante e ineficiente, com desperdício significativo de espaço em memória e tempo computacional.

Por outro lado, em várias circunstâncias, é exigido o conhecimento da resolução e da covariância dos dados com o objetivo de se controlar a qualidade e a coerência dos resultados. No entanto, a sua obtenção pode, também, demandar grande parcela de memória computacional se as matrizes forem implementadas pelos métodos convencionais, uma vez que o seu cálculo pode envolver inversão matricial. Em Sismologia, dispõe-se de ferramentas computacionais usadas para contornar o problema da demanda (Zhang & McMechan, 1995). Citam-se, por exemplo, os métodos de Backus-Gilbert (Backus & Gilbert, 1968; Ishii & Tromp, 1999; Vdovin et al., 1999) e os testes de “checkerboard” (Lévêque et al., 1993; Bank et al., 2000; Ritzwoller & Levshin, 1998), que normalmente são associados à técnica dos Gradientes Conjugados (Scales, 1987; Hestenes, 1975; Munksgaard, 1980; Yuan & Iusem, 1996; Paige & Saunders, 1982a, 1982b). Este último esquema é particularmente eficiente, cuja solução converge rapidamente quando o sistema é simétrico e positivo-definido. Porém, tais ferramentas não permitem o cálculo rigoroso das matrizes resolução e covariância, uma vez que uti-

lizam dados sintéticos derivados de modelos aproximados, ao invés da informação original observada. Nesse caso, a resolução é estimada de forma sintética e não analítica. Adicionalmente, em situações muito desfavoráveis, esses métodos podem induzir a equívocos na interpretação dos resultados (ver, por exemplo, Lévêque et al., 1993; Ritzwoller & Levshin, 1998; Vdovin et al., 1999). Uma alternativa para este impasse é o investimento em equipamento computacional, que pode incluir desde computadores de grande porte (supercomputadores) até as redes de processadores do tipo PC (“Beowulf clusters”), operando em ambiente de compartilhamento de memória RAM (Sterling et al., 1999). No entanto, tais soluções podem representar um alto custo financeiro para o projeto, principalmente em instituições de pesquisa com escassez de recursos, como normalmente ocorre em nosso país.

O presente estudo investiga um algoritmo vetorial que pode ser usado como método alternativo para solucionar, eficientemente, problemas geofísicos que envolvam grandes sistemas esparsos, com a inclusão do cálculo rigoroso das matrizes resolução e covariância. Embora várias sugestões sejam disponíveis para abordagem do problema da esparsidade (ver Ray, 1985; Vasco et al., 1998), foi adotado aqui o método de Horowitz-Sahni devido, sobretudo, à simplicidade de implementação e adaptação à maioria das situações encontradas. O desempenho do algoritmo foi testado numa aplicação específica na área de Sismologia, ou seja, tomografia com ondas sísmicas superficiais (Souza et al., 2003; Pacheco, 2003; Souza & Santos, 2003; Vilar et al., 2003; Vilar, 2004), que se apresenta como exemplo típico de grande sistema esparsos resolvido por MMQ. O algoritmo foi, também, confrontado com duas técnicas tradicionais de implementação que não utilizam quaisquer esquemas para compressão dos dados.

## TOMOGRAFIA COM VELOCIDADE DE GRUPO

Nesta seção, é mostrado o MMQ como solução para um problema tomográfico de velocidade de grupo com ondas sísmicas superficiais, e, na seção seguinte, o algoritmo vetorial é discutido como técnica de implementação do sistema esparsos resultante. Obviamente, cada tipo de problema estudado produz um sistema particular de equações a ser resolvido, e, neste caso, o algoritmo deve ser devidamente adaptado para aquela situação. No entanto, permanece a intenção de minimizar recurso computacional através do método proposto para compressão de dados.

As ondas sísmicas superficiais podem ser utilizadas para investigar a estrutura da velocidade da onda *S* no interior da Terra. No entanto, esse tipo de aplicação pode envolver longas tra-

jetórias fonte-estação que percorrem várias províncias tectônicas. A técnica de subdivisão da trajetória fonte-estação em segmentos (regionalização), tem sido utilizada em problemas de propagação de ondas superficiais para inferência de modelos regionais (e.g., Souza et al., 2003; Pacheco, 2003; Souza & Santos, 2003; Vilar et al., 2003; Vilar, 2004; Feng, 2004; Feng et al., 2004).

Vamos supor que uma determinada quantidade ( $t$ ) de trajetórias, entre a fonte sísmica e a estação sismográfica, percorre um certo número ( $c$ ) de células numa determinada região, delimitada por uma grade geográfica. Adicionalmente, considere que o tempo total de propagação de uma dada trajetória seja igual à soma dos tempos parciais de propagação em cada célula percorrida pelas ondas superficiais. Logo, para um período particular, e, considerando uma certa quantidade de trajetórias fonte-estação ( $j$ ) e de receptores ( $i$ ), é possível estabelecer a seguinte relação (Feng & Teng, 1983)

$$\frac{1}{U^j} - \frac{1}{V^j} = \sum_{i=1}^c \frac{d_i^j}{D^j} \left( \frac{1}{u^i} - \frac{1}{v^i} \right), \quad j = 1, \dots, t \quad (1)$$

onde,

$U^j$  – velocidade de grupo observada da  $j$ -ésima trajetória entre a fonte e o receptor;

$V^j$  – velocidade de grupo teórica da  $j$ -ésima trajetória entre a fonte e o receptor;

$D^j$  – distância fonte-estação da  $j$ -ésima trajetória;

$d_i^j$  – distância da  $j$ -ésima trajetória fonte-estação na  $i$ -ésima célula;

$u^i, v^i$  – velocidades de grupo estimada e teórica na  $i$ -ésima célula, respectivamente.

A Figura 1 apresenta um esquema simples em que dois eventos sísmicos são registrados em duas estações sismográficas, numa região subdividida em nove células ( $c = 9$ ), gerando quatro trajetórias fonte-estação ( $t = 4$ ). Reescrevendo a eq. (1) na forma matricial, tem-se

$$\mathbf{Y} = \mathbf{A} \cdot \mathbf{X} \quad (2)$$

onde,

$\mathbf{A}_{(t,c)}$  – matriz dos coeficientes ( $t$  – trajetórias;  $c$  – células), que armazena as respectivas relações entre as distâncias percorridas em cada célula e a distância total ( $d_i^j / D^j$ );

$\mathbf{Y}_{(t,1)}$  – inversos das velocidades de grupo ou vagarosidades (parâmetros);

$\mathbf{X}_{(c,1)}$  – inversos das velocidades de grupo ou vagarosidades em cada célula (incógnitas).

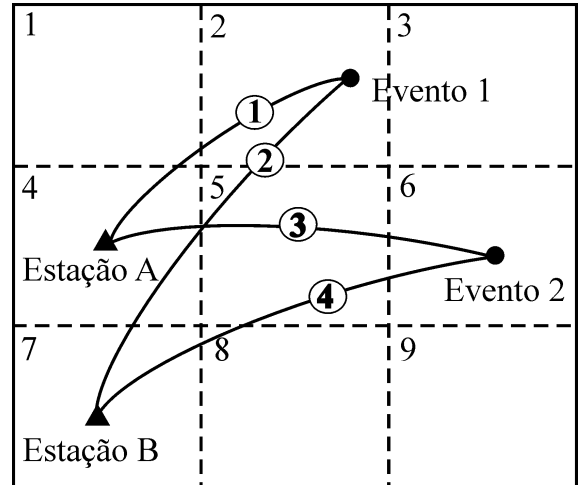


Figura 1 – Representação esquemática de dois eventos sísmicos registrados em duas estações, com as respectivas trajetórias das ondas sísmicas superficiais.

Em termos de MMQ amortecido, a solução é dada por

$$\mathbf{X} = (\mathbf{A}^T \cdot \mathbf{A} + \varepsilon \cdot \mathbf{I})^{-1} \mathbf{A}^T \cdot \mathbf{Y} \quad (3)$$

onde  $\mathbf{I}$  é a matriz identidade. O método de MMQ amortecido (“Levenberg-Marquardt damped least-squares method” – Levenberg, 1944; Marquardt, 1963) é tradicionalmente usado para controlar a instabilidade no produto  $\mathbf{A}^T \cdot \mathbf{A}$  (Franklin, 1970), e consiste na adição de uma constante,  $\varepsilon > 0$ , à diagonal principal da matriz  $\mathbf{A}^T \cdot \mathbf{A}$ .

A incerteza na solução ( $\mathbf{X}$ ) é calculada através da matriz covariância, que indica o grau de confiabilidade estatística ou verossimilhança dos resultados. Ela é fornecida pela equação

$$\mathbf{C} = \sigma^2 (\mathbf{A}^T \cdot \mathbf{A} + \varepsilon \cdot \mathbf{I})^{-1} \mathbf{A}^T \cdot \mathbf{A} (\mathbf{A}^T \cdot \mathbf{A} + \varepsilon \cdot \mathbf{I})^{-1} \quad (4)$$

onde  $\sigma^2$  é a variância das observações. Os elementos da diagonal principal da matriz covariância traduzem os desvios-padrão ou variâncias da solução.

Por outro lado, o grau de detalhe no qual a solução pode ser espacialmente resolvida é conhecido através da matriz resolução, dada por

$$\mathbf{R} = (\mathbf{A}^T \cdot \mathbf{A} + \varepsilon \cdot \mathbf{I})^{-1} \mathbf{A}^T \cdot \mathbf{A} \quad (5)$$

cujos elementos da diagonal principal fornecem a medida da resolução no espaço dos dados (Aki & Richards, 1980). A configuração ideal para a matriz resolução é a matriz identidade. Contudo, na prática, o melhor valor de resolução ocorre nas células cuja amplitude máxima se situa próxima à diagonal principal, e os demais elementos possuem pequeno decaimento (Husen et al., 2000; Toomey & Foulger, 1989). O decaimento espacial da matriz resolução em torno de uma determinada célula da grade

deve ser usado na avaliação da resolução da mesma (Menke, 1989), e traduz a qualidade da solução obtida, revelando o quão próximo da realidade se encontra o modelo matemático adotado. Uma boa estimativa da resolução depende, ainda, do espaçamento usado na grade geográfica e do parâmetro de amortização ( $\epsilon$ ).

## COMPACTAÇÃO DAS MATRIZES DO SISTEMA

A seguir, são examinadas as estruturas de dados utilizadas para a organização dos dois tipos de matrizes (esparsa e simétrica) encontrados no sistema de equações do problema estudado (tomografia com velocidade de grupo). É também apresentado um esquema para inversão matricial em formato compactado, proposto para este trabalho, que se baseia no algoritmo desenvolvido pela Ohio State University.

### Matriz Esparsa

A matriz retangular,  $\mathbf{A}_{(t,c)}$ , na eq. (2), é um exemplo de matriz esparsa cuja ocorrência é muito comum em áreas como Gravimetria, Geodésia e Sismologia. No presente trabalho (tomografia com ondas sísmicas superficiais), ambas as quantidades, trajetórias e células, são, em geral, muito grandes. Neste caso, é conhecido que nem todas as trajetórias sísmicas atravessam todas as células, portanto, um número considerável de “zeros” é armazenado em  $\mathbf{A}$ , que correspondem às células que não são percorridas por nenhuma trajetória. Se ambas as quantidades forem respectivamente aumentadas, a matriz se torna ainda mais esparsa. Esse é um fato comum em Sismologia, onde há necessidade de se aumentar o número de trajetórias para se obter um acréscimo na resolução final das imagens.

O esquema de Horowitz-Sahni é usado para compactar a matriz esparsa, no qual apenas os elementos não-nulos e respectivos índices (linha e coluna) são arranjados em uma lista tríplice de vetores. Desta forma, apenas o espaço mínimo indispensável é utilizado para a alocação da matriz, sem perda de informação. Com a finalidade de facilitar as operações matriciais (soma, multiplicação, transposta e acesso aos dados), quatro vetores auxiliares são incluídos no processo.

Como ilustração do esquema de Horowitz-Sahni, consideremos a matriz do sistema,  $\mathbf{A}_{(4,7)}$ , obtida a partir da Fig. 1. As linhas e colunas da matriz correspondem, respectivamente, às quatro trajetórias fonte-estação e sete células efetivamente per-

corridas:

$$\mathbf{A}_{(4,7)} = \begin{pmatrix} 0,1 & 0,6 & 0,4 & 0 & 0 & 0 & 0 \\ 0 & 0,3 & 0,3 & 0,2 & 0 & 0,2 & 0 \\ 0 & 0 & 0,2 & 0,5 & 0,3 & 0 & 0 \\ 0 & 0 & 0 & 0,4 & 0,2 & 0,3 & 0,1 \end{pmatrix}$$

Os valores decimais indicam as relações entre as distâncias,  $d_i^j / D^j$  (eq. (1)), para cada trajetória e respectivas células percorridas. Células não atravessadas por nenhuma trajetória, como as células 3 e 9 (vide Fig. 1), não figuram na matriz, uma vez que as colunas correspondentes são vazias. Cada coluna de  $\mathbf{A}$  possui, pelo menos, um elemento não-nulo. A implementação da matriz  $\mathbf{A}$ , segundo o esquema de Horowitz-Sahni, consta de três vetores que armazenam as respectivas seqüências de índices e valores não-nulos, conforme o diagrama  $\mathbf{T} - \mathbf{C} - \mathbf{V}$  (trajetória-célula-valor), na tabela abaixo:

Índice	<b>T</b>	<b>C</b>	<b>V</b>
1	1	1	0,1
2	1	2	0,6
3	1	3	0,4
4	2	2	0,3
5	2	3	0,3
6	2	4	0,2
7	2	6	0,2
8	3	3	0,2
9	3	4	0,5
10	3	5	0,3
11	4	4	0,4
12	4	5	0,2
13	4	6	0,3
14	4	7	0,1

onde,

$\mathbf{T}_{(n,1)}$  – armazena os índices das linhas de  $\mathbf{A}$ , seqüencialmente, por elemento não-nulo, e relaciona-se às trajetórias fonte-estação. O número ( $n$ ) de elementos não-nulos (número de termos do vetor) é igual a 14;

$\mathbf{C}_{(n,1)}$  – armazena os índices das colunas de  $\mathbf{A}$ , seqüencialmente, por elemento não-nulo, e está relacionado com as células percorridas;

$\mathbf{V}_{(n,1)}$  – armazena os valores dos elementos não-nulos, seqüencialmente, por linha de  $\mathbf{A}$ .

Em resumo, cada linha no diagrama  $\mathbf{T} - \mathbf{C} - \mathbf{V}$  contém a respectiva linha, coluna e elemento não-nulo de  $\mathbf{A}$ . Os vetores  $\mathbf{T}$  e  $\mathbf{C}$  são implementados (código Fortran 77) do tipo integer\*4,

portanto, para cada vetor são reservados  $4 \cdot n$  bytes para armazenamento em memória. O vetor  $\mathbf{V}$  é do tipo real\*8, e exige  $8 \cdot n$  bytes para a sua alocação.

Os quatro vetores, a seguir, auxiliam todo o processamento matricial, e visam à otimização no acesso aos dados, reduzindo significativamente o tempo de processamento:

$$\mathbf{E} = \begin{pmatrix} 1 \\ 2 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{pmatrix} \quad \mathbf{F} = \begin{pmatrix} 0 \\ 3 \\ 7 \\ 10 \\ 14 \end{pmatrix} \quad \mathbf{G} = \begin{pmatrix} 0 \\ 1 \\ 3 \\ 6 \\ 9 \\ 11 \\ 13 \\ 14 \end{pmatrix} \quad \mathbf{H} = \begin{pmatrix} 1 \\ 2 \\ 4 \\ 3 \\ 5 \\ 8 \\ 6 \\ 9 \\ 11 \\ 10 \\ 12 \\ 7 \\ 13 \\ 14 \end{pmatrix}$$

onde,

$\mathbf{E}_{(c,1)}$  – armazena os índices originais das células (excetuando as células 3 e 9);

$\mathbf{F}_{(t+1,1)}$  – acumula a quantidade de células percorridas por cada trajetória (linhas de  $\mathbf{A}$ ). O número de células percorridas por uma dada trajetória ( $j$ ) é obtido através de  $\mathbf{F}(j + 1) - \mathbf{F}(j)$ ;

$\mathbf{G}_{(c+1,1)}$  – acumula a quantidade de trajetórias que atravessam cada célula (colunas de  $\mathbf{A}$ ). O número de trajetórias que percorre a célula ( $i$ ) é dado por  $\mathbf{G}(i + 1) - \mathbf{G}(i)$ ;

$\mathbf{H}_{(n,1)}$  – armazena os índices dos elementos não-nulos, seqüencialmente, por coluna de  $\mathbf{A}$ . Constitui-se em uma inversão indexada do diagrama  $\mathbf{T} - \mathbf{C} - \mathbf{V}$ , para otimização no cálculo da transposta.

Os vetores  $\mathbf{E}$ ,  $\mathbf{F}$ ,  $\mathbf{G}$  e  $\mathbf{H}$  são implementados do tipo integer\*4 e, portanto, cada qual requer quatro vezes o seu número de termos em bytes, para armazenamento.

Somando-se as respectivas dimensões do diagrama  $\mathbf{T} - \mathbf{C} - \mathbf{V}$  e dos vetores  $\mathbf{E}$ ,  $\mathbf{F}$ ,  $\mathbf{G}$  e  $\mathbf{H}$ , obtém-se uma relação que permite comparar as demandas (em bytes), para armazenamento da matriz  $\mathbf{A}$  nos métodos vetorial e tradicional,

$$\underbrace{20 \cdot n + 8 \cdot c + 4 \cdot t + 8}_{\text{vetorial}} \Leftrightarrow \underbrace{8 \cdot t \cdot c}_{\text{tradicional}} \quad (6)$$

Isolando-se o termo  $n$  dos demais, obtemos

$$n \Leftrightarrow [2 \cdot t \cdot c - (2 \cdot c + t + 2)]/5 \quad (7)$$

Ao se atribuir valores respectivos aos termos  $t$  e  $c$ , deduz-se que a representação vetorial é mais eficiente que a tradicional quando  $n < 0,4 \cdot t \cdot c$ , o que nos auxilia a decidir qual o método mais apropriado a ser usado. É importante mencionar que a eq. (7) é válida apenas para os tipos de variáveis integer\*4 e real\*8, usados neste contexto.

Com efeito, o cálculo matricial torna-se bastante ágil e simplificado com o emprego do esquema vetorial acima. Considerando-se, como exemplo, o produto da transposta de uma dada matriz  $\mathbf{A}_{(t,c)}$ , representada pelo algoritmo de Horowitz-Sahni, por ela própria (seja  $\mathbf{N} = \mathbf{A}^T \cdot \mathbf{A}$ ), o cálculo pode ser implementado através do seguinte procedimento (em notação algorítmica):

1º ciclo: Determina os elementos da diagonal principal de  $\mathbf{N}$

$$\left\{ \begin{array}{l} k \leftarrow 1, c \\ i \leftarrow \mathbf{G}(k) + 1, \mathbf{G}(k + 1) \\ \mathbf{N}(k, k) \leftarrow \mathbf{N}(k, k) + \mathbf{V}(\mathbf{H}(i)) \times \mathbf{V}(\mathbf{H}(i)) \end{array} \right.$$

Este ciclo acumula a soma dos quadrados dos elementos não-nulos de  $\mathbf{A}$ , correspondentes às linhas e colunas de mesmo índice;

2º ciclo: Determina os elementos situados acima da diagonal principal de  $\mathbf{N}$

$$\left\{ \begin{array}{l} k \leftarrow 1, c \\ l \leftarrow k + 1, c \\ j \leftarrow \mathbf{G}(k) + 1, \mathbf{G}(k + 1) \\ i \leftarrow \mathbf{G}(l) + 1, \mathbf{G}(l + 1) \\ \text{se } \mathbf{T}(\mathbf{H}(j)) = \mathbf{T}(\mathbf{H}(i)) \text{ então} \\ \mathbf{N}(l, k) \leftarrow \mathbf{N}(l, k) + \mathbf{V}(\mathbf{H}(j)) \times \mathbf{V}(\mathbf{H}(i)) \end{array} \right.$$

Este ciclo acumula a soma dos produtos de elementos correspondentes entre linhas de  $\mathbf{A}^T$  e colunas de  $\mathbf{A}$ , ou seja, o  $n$ -ésimo elemento da linha  $l$  em  $\mathbf{A}^T$ , e o  $n$ -ésimo elemento da coluna  $k$  em  $\mathbf{A}$ . Tendo em vista que a matriz  $\mathbf{N}$  é simétrica, basta calcular apenas uma de suas bandas, pois ela será representada pela estrutura descrita na seção seguinte, para economia de espaço de armazenamento.

Considerando-se que são executados apenas os produtos que não contém "zeros" entre os seus operandos, a rotina mostra-se bastante eficiente, resultando em uma significativa economia no tempo de processamento. Isto ocorre em razão de que somente as operações efetivamente necessárias ao cálculo são efetuadas. Outrossim, embora o algoritmo utilize vetores adicionais ( $\mathbf{F}$ ,  $\mathbf{G}$  e  $\mathbf{H}$ ) para armazenamento de contadores, tal não acresce o gasto

computacional, uma vez que a sua manipulação envolve, na maioria das situações, apenas o acesso aos índices, que se constitui em uma operação extremamente eficiente no computador. Como ilustração, utilizando-se os dados numéricos da matriz  $\mathbf{A}_{(4,7)}$ , mostrada acima, o produto  $\mathbf{A}^T \cdot \mathbf{A}$  exige 96 ciclos de operações pelo método de Horowitz-Sahni, enquanto que o método tradicional exige 112 ciclos, o que implica em uma economia em torno de 14% na quantidade de instruções executadas.

O algoritmo de Horowitz-Sahni é de ampla aplicação prática, pois pode ser basicamente utilizado em todas as ocorrências de matrizes esparsas. Como todos os elementos não-nulos (inclusive os de pequena ordem de grandeza) são efetivamente armazenados, os resultados dos cálculos matriciais são exatos. Naturalmente, algum tipo de regularização (como, por exemplo, “ridge regression”), ou o fator de amortecimento (vide eq. (3)), é exigido em caso de sistemas mal-condicionados. Como o algoritmo não se baseia em métodos iterativos, os problemas de aproximação devido às considerações numéricas (como erros de arredondamento que dependem da precisão do computador) são, então, evitados. O esquema difere, por exemplo, dos métodos que utilizam iterações com sub-espacos vetoriais (auto-decomposição da matriz do sistema, por SVD) e o método de Lanczos (ver Vasco et al., 1998). O algoritmo de Horowitz-Sahni pode ser ainda empregado em situações específicas, em que a matriz do sistema possui um formato particular para o problema em questão. Este é o caso descrito em Ray (1985), por exemplo, onde é aplicado um esquema de decomposição ortogonal por blocos (“QR decomposition”) ao sistema esparso. Qualquer que seja o caso estudado, por mais específico que seja o seu caráter ou formato, não invalida o uso do algoritmo de Horowitz-Sahni.

Ainda como exemplo da eficiência do algoritmo, considerando-se um total de 3.269 trajetórias fonte-estação, que percorrem 10.161 células quadradas de  $1^\circ \times 1^\circ$  (matriz de dimensões  $3.269 \times 10.161$ ), a implementação vetorial economiza cerca de 98% de espaço de armazenamento, quando comparado ao método tradicional. Neste caso particular, são necessárias 46.856.523.881 instruções para computar o produto  $\mathbf{A}^T \cdot \mathbf{A}$  pelo método de Horowitz-Sahni, em contraste com 168.772.066.029 instruções, pelo método tradicional (ou seja, uma redução de 73%). Tais resultados denotam uma excepcional economia de recurso computacional, tendo em vista que o tempo de processamento também é reduzido, devido à diminuição no número de acessos aos dados.

### Matrizes Simétricas

A matriz  $(\mathbf{A}^T \cdot \mathbf{A} + \varepsilon \cdot \mathbf{I})_{(c,c)}$ , na eq. (3), e as matrizes, covariância e resolução (eqs. 4 e 5) possuem os elementos dispostos simetricamente em relação à diagonal principal. Portanto, é conveniente representá-las em uma estrutura não redundante e econômica, o que se agrava muito quando o número de elementos é elevado (por exemplo, maior do que  $10^6$ ). Horowitz & Sahni (1976) propõem uma representação em que uma das bandas da matriz e mais a diagonal principal, são compactadas em um vetor. Dada a matriz simétrica  $\mathbf{N}_{(m,m)}$ , o termo genérico  $(n_{ij})$  é mapeado para o elemento correspondente  $(p_u)$ , no vetor  $\mathbf{P}$ , através da relação  $u = i(i-1)/2 + j$ , para  $i \geq j$ . Inversamente, os índices  $i$  e  $j$  são calculados através de  $i = j = \lceil s/2 \rceil$ , se  $s = \sqrt{1 + 8 \cdot u}$  for inteiro; em caso contrário,  $i = \lceil (s+1)/2 \rceil$ , e  $j = u - i(i-1)/2$ .

Considerando-se que o vetor  $\mathbf{P}$  armazena um total de  $(m^2 + m)/2$  elementos, em contraste com  $m^2$  elementos, reservados para a matriz  $\mathbf{N}$ , o espaço exigido em memória é reduzido em cerca de 50%. Adicionalmente, obtém-se uma economia proporcional no tempo de processamento, devido à redução no número de acessos aos dados.

### Inversão Matricial

O método desenvolvido pelo “Department of Geodetic Science – Ohio State University (OSU)” (autor desconhecido), utilizado neste trabalho, é um algoritmo exato e eficiente para cálculo de inversão matricial. A inversa de uma dada matriz,  $\mathbf{N}_{(m,m)}$ , é obtida em exatamente  $m$  passos, nos quais a original e a resultante compartilham a mesma locação de memória. O esquema utiliza, ainda, um vetor de  $m$  elementos para auxiliar o processo. Entretanto, é curioso notar que, se a matriz original for simétrica, o algoritmo OSU gera um padrão peculiar que é usado, neste contexto, para economizar espaço de armazenamento: em cada etapa intermediária, as matrizes parciais são também simétricas – à exceção de determinados elementos que diferem, em sinal, dos respectivos conjugados. O exemplo numérico, a seguir, ilustra o esquema OSU para inversão de uma matriz simétrica  $3 \times 3$ :

$$\text{Matriz original: } \begin{pmatrix} 9 & 4 & 2 \\ 4 & 3 & 8 \\ 2 & 8 & 1 \end{pmatrix}$$

$$\begin{aligned}
 &1^\circ \text{ passo: } \begin{pmatrix} 1, 22 & 7, 11 & \vdots & -\mathbf{0, 44} \\ 7, 11 & 0, 56 & \vdots & -\mathbf{022} \\ \dots & \dots & \dots & \dots \\ \mathbf{0, 44} & \mathbf{0, 22} & \vdots & 0, 11 \end{pmatrix} \\
 &2^\circ \text{ passo: } \begin{pmatrix} -40, 82 & \vdots & \mathbf{2, 36} & -\mathbf{5, 82} \\ \dots & \dots & \dots & \dots \\ -\mathbf{2, 36} & \vdots & 0, 27 & -0, 36 \\ \mathbf{5, 82} & \vdots & -0, 36 & 0, 82 \end{pmatrix} \\
 &\text{Matriz inversa: } \begin{pmatrix} 0, 16 & -0, 03 & -0, 06 \\ -0, 03 & -0, 01 & 0, 14 \\ -0, 06 & 0, 14 & -0, 02 \end{pmatrix}
 \end{aligned}$$

Em cada uma das etapas do algoritmo OSU, os elementos das matrizes intermediárias são calculados de acordo com o seguinte esquema de atribuições:

Vetor **B**

$$\begin{aligned}
 b_1 &\leftarrow n_{1,2}/n_{1,1} \\
 b_2 &\leftarrow n_{1,3}/n_{1,1} \\
 b_3 &\leftarrow 1/n_{1,1}
 \end{aligned}$$

1ª linha de **N**

$$\begin{aligned}
 n_{1,1} &\leftarrow n_{2,2} - n_{2,1} \times b_1 \\
 n_{1,2} &\leftarrow n_{2,3} - n_{2,1} \times b_2 \\
 n_{1,3} &\leftarrow -n_{2,1} \times b_3
 \end{aligned}$$

2ª linha de **N**

$$\begin{aligned}
 n_{2,1} &\leftarrow n_{3,2} - n_{3,1} \times b_1 \\
 n_{2,2} &\leftarrow n_{3,3} - n_{3,1} \times b_2 \\
 n_{2,3} &\leftarrow -n_{3,1} \times b_3
 \end{aligned}$$

3ª linha de **N**

$$\begin{aligned}
 n_{3,1} &\leftarrow b_1 \\
 n_{3,2} &\leftarrow b_2 \\
 n_{3,3} &\leftarrow b_3
 \end{aligned}$$

onde **B** é um vetor de trabalho contendo *m* termos (no exemplo acima, *m* = 3). Após uma série de *m* execuções desta rotina, a matriz **N**<sub>(*m*,*m*)</sub> é invertida. O algoritmo OSU produz resultados exatos (em dupla precisão), e é de simples implementação em computador.

Os elementos assinalados em negrito nos passos 1 e 2 têm os seus sinais invertidos. Como regra de construção, nas etapas

*k* = 1, ..., *m* - 1, os elementos das linhas *m* - *k* + 1 até *m*, e colunas 1 até *m* - *k*, possuem os sinais invertidos com os respectivos simétricos. O processo pode, ainda, ser visualizado com o auxílio das linhas tracejadas (1º e 2º passos), que se movem, respectivamente, para a esquerda e para cima, e separa os elementos em negrito nos respectivos espaços de simetria.

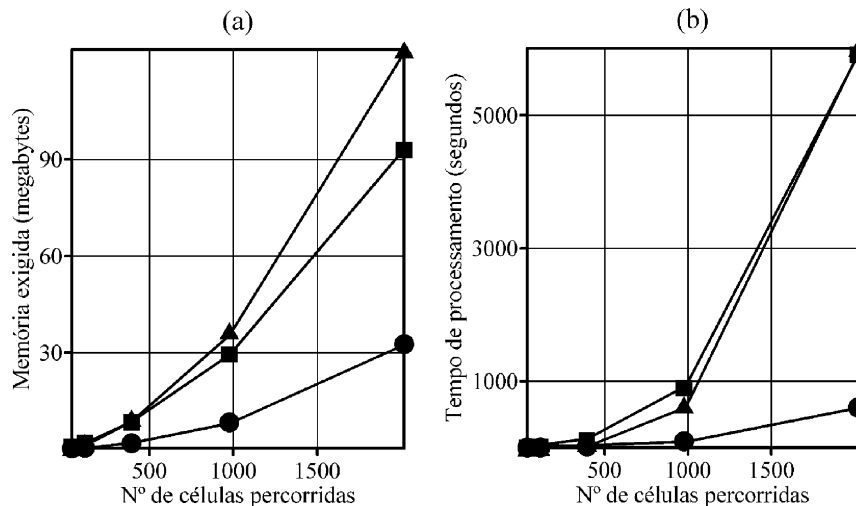
Com base nesta construção, o método OSU é apropriadamente adaptado para um algoritmo de inversão matricial em formato comprimido. O novo esquema requer a alocação de dois vetores para auxiliar o processamento. Considerando-se que o método OSU consome *m*<sup>2</sup> + *m* posições de memória, e o algoritmo proposto utiliza apenas (*m*<sup>2</sup> + *m*)/2 + 2 · *m*, obtém-se uma economia adicional no espaço de armazenamento em torno de 50%. Da mesma forma que nas seções anteriores, a redução no número de acessos aos dados, devido à remoção da estrutura redundante, implica em uma economia proporcional no tempo de processamento.

### DESEMPENHO DO ALGORITMO VETORIAL

O algoritmo proposto foi, primeiramente, confrontado com os métodos clássico e SVD, em um teste de desempenho, cujos resultados estão exibidos nos dois gráficos da Fig. 2. Ambos mostram as respectivas demandas de espaço exigido em memória (a) e tempos de processamento (b), com relação às quantidades de células percorridas por um número fixo de trajetórias fonte-estação, e se referem ao processamento das eqs. (3), (4) e (5).

O conjunto de dados inclui 2.000 trajetórias de ondas sísmicas superficiais (tipo Rayleigh), registradas em 23 estações do consórcio de universidades norte-americanas IRIS (“Incorporated Research Institutions for Seismology”). A região estudada situa-se entre 90°N e 90°S, e 149°W e 31°E, e foi subdividida em células de 2° × 2°, 3° × 3°, 5° × 5°, 10° × 10° e 20° × 20°. Estas subdivisões dão origem às respectivas quantidades de 2018, 976, 394, 115 e 38 células percorridas (abscissas nos gráficos). As rotinas foram implementadas (código Fortran 77) em um computador do tipo PC Pentium III/128MB RAM. É importante ressaltar que a dimensão do conjunto de dados foi limitada, exclusivamente, pela capacidade do processador utilizado. Os conceitos de SVD estão de acordo com a literatura geofísica (Aki & Richards, 1980; Lines & Treitel, 1984; Berry, 1992), cujas rotinas foram extraídas do “Numerical Recipes Software” v3.5 (Press et al., 1992). (Para o experimento acima, foram computadas apenas as variâncias das observações, portanto, a matriz covariância foi reduzida aos elementos da diagonal principal).

Muito embora o teste seja de caráter teórico, é incontestável



**Figura 2** – Demanda exigida em memória computacional (a) e tempo de processamento (b) empregando-se, respectivamente, os métodos de implementação vetorial (●), tradicional (■) e SVD (▲).

a excepcional economia obtida com o uso do algoritmo, em contraste com o evidente desperdício, tanto em tempo de processamento quanto em memória computacional, nos métodos tradicionais. Por exemplo, considerando-se uma subdivisão da grade em células de  $2^\circ \times 2^\circ$ , temos um total de  $2.000 \times 2.018 = 4.036.000$  elementos a serem alocados na matriz  $\mathbf{A}$  (método tradicional), porém, destes, somente 79.320 são não-nulos. Desta forma, uma percentagem de 98% da matriz (ou seja, quase a totalidade) é composta de elementos nulos, sendo este espaço inaproveitado. Os gráficos da Fig. 2 revelam um ganho significativo, principalmente nos tempos de processamento (Fig. 2b), à medida que a quantidade de células da grade aumenta (no caso, para um mesmo número de trajetórias).

É evidente que, no caso da tomografia sísmica, aumentando-se ambos os números de células e de trajetórias, aparentemente “zeros” serão removidos da matriz do sistema. Entretanto, ambas as quantidades de equações e incógnitas serão também acrescidas, portanto, a quantidade de “zeros” também aumentará. Isto deve-se ao fato de que, na prática, as trajetórias não percorrem efetivamente todas as células. É ainda interessante notar que, sendo a economia proporcional à quantidade de células adotadas, o método tradicional pode se apresentar, em algumas situações, como mais eficiente que o esquema vetorial (por exemplo, quando o número de células é bem reduzido. Vide eq. (7)).

O algoritmo foi também utilizado na investigação da estrutura das velocidades da onda  $S$ , em profundidade, nas regiões Sudeste (Souza et al., 2003; Pacheco, 2003; Souza & Santos, 2003) e Nordeste do Brasil (Vilar et al., 2003; Vilar, 2004). O

mesmo conjunto de dados foi utilizado nos dois trabalhos, que inclui 1.059 eventos sísmicos (ondas superficiais do tipo Rayleigh) registrados em 23 estações da rede IRIS, dando origem a 3.354 trajetórias fonte-estação. Ambas as áreas foram subdivididas em grades geográficas de  $2^\circ \times 2^\circ$ , perfazendo um total de 4.500 células. As dimensões dos sistemas de equações resultantes (quantidade de trajetórias  $\times$  células percorridas) variam de acordo com o período e a área investigada, portanto, a economia obtida na demanda computacional também é variável. Considerando-se, por exemplo, o período de 10 s (região Sudeste), temos um total de 2.409 trajetórias atravessando 2.563 células. Neste caso, utilizando-se o método tradicional (sistema de equações e parâmetros, de  $2.409 \times 2.563$ ) consumiu-se 64 h e 49 min de processamento, ao passo que, com o algoritmo vetorial, consumiu-se apenas 4 h e 34 min. Assim, os recursos computacionais foram drasticamente reduzidos em 16 vezes, em comparação com o método clássico. De maneira geral, a economia nos tempos de processamento e memória exigida foi sempre superior a 90%, tendo sido utilizado um único computador do tipo AMD/K7 2GB RAM (Linux OS), em regime de operação ininterrupta. Descrições pormenorizadas e informações adicionais sobre ambos os casos reais acima, estão detalhadas nas referências retro mencionadas.

## CONCLUSÕES

A representação de dados através de estrutura vetorial, proposta como um recurso para reduzir demanda computacional, tem sido aplicada, com êxito, na área de Sismologia do Observatório Naci-



onal. Esta técnica possibilitou o emprego de um único computador do tipo PC em problemas de tomografia com ondas sísmicas superficiais, aplicados no território brasileiro. Além de permitir o cálculo rigoroso das matrizes resolução e covariância, o algoritmo economizou, de maneira significativa, tanto memória RAM quanto tempo de processamento, em comparação com as técnicas tradicionais. Os resultados nos testes de desempenho indicaram uma economia variando entre 65% e 98%, conforme as quantidades de trajetórias fonte-estação e células da grade adotadas, assim como os tempos de processamento nos dois casos reais estudados, que foram reduzidos por um fator de dezesseis. Constatou-se, ainda, que a eficiência do algoritmo aumenta quando o volume de dados é acrescido.

Considerando-se que, para a solução de alguns problemas em Sismologia, é interessante se incrementar, cada vez mais, a quantidade de trajetórias fonte-estação (com o conseqüente e indesejável aumento na esparsidade do sistema), o uso do algoritmo vetorial pode ser um instrumento bastante vantajoso e eficaz para a otimização de recursos, como foi mostrado no trabalho.

Por outro lado, embora o algoritmo tenha sido empregado, neste contexto, a computadores restritos (do tipo PC), ele pode ser amplamente utilizado em ambientes de compartilhamento de memória, como as arquiteturas de processamento paralelo ("Beowulf clusters"). O uso do esquema vetorial, associado àquelas técnicas de otimização por compartilhamento de recursos, tende a maximizar, ainda mais, a economia na demanda computacional.

Adicionalmente, apesar de sua aplicação, neste trabalho, a um problema específico na área de Sismologia (tomografia com ondas sísmicas superficiais), o esquema pode ser utilizado nas demais situações geofísicas que envolvam o gerenciamento de grandes sistemas esparsos, com a inclusão opcional do cálculo das matrizes resolução e covariância. Entre outras aplicações geofísicas, pode-se citar, por exemplo, a solução de determinados problemas de inversão, e os ajustamentos matemáticos por MMQ, de observações gravimétricas e cálculos geodésicos.

Em resumo, o algoritmo vetorial pode ser empregado amplamente, pois, além de apresentar simplicidade na implementação, produz resultados exatos, uma vez que não depende de considerações numéricas que podem induzir a erros de arredondamento, como nos métodos iterativos. Assim sendo, qualquer que seja o sistema esparsos, por maior volume e especificidade de seu formato, não invalida o emprego do algoritmo vetorial como ferramenta computacional de grande eficácia.

## AGRADECIMENTOS

Os autores agradecem à Fundação de Amparo à Pesquisa do Estado do Rio de Janeiro (FAPERJ) pelo apoio financeiro (Processo nº E-26/170.407/2000). Eles agradecem, também, ao Incorporated Research Institutions for Seismology (IRIS) pelos dados utilizados neste estudo, ao MCT / Observatório Nacional pela infraestrutura institucional, ao Prof. Dr. José Bittencourt de Andrade (UFPr – Curitiba) pela cessão da rotina de inversão matricial da Ohio State University, e, finalmente aos dois revisores anônimos pela contribuição com os comentários e sugestões valiosas.

## REFERÊNCIAS

- AKI K & RICHARDS PG. 1980. *Quantitative Seismology Theory and Methods*. W. H. Freeman and Co., San Francisco, 932 pp.
- BACKUS G & GILBERT F. 1968. The resolving power of gross Earth data, *Geophys. J. R. Astr. Soc.* 16(2): 169–205.
- BANK C-G, BOSTOCK MG, ELLIS RM & CASSIDY JF. 2000. A reconnaissance teleseismic study of the upper mantle and transition zone beneath the Archean Slave craton in NW Canada, *Tectonophysics* 319: 151–166.
- BERRY MW. 1992. Large scale sparse singular value computations, *The International Journal of Supercomputer Applications* 6(1): 13–49.
- FENG CC & TENG TL. 1983. Three-dimensional crust and upper mantle structure of the eurasian continent, *J. Geophys. Res.* 88(B3): 2261–2272.
- FENG M. 2004. Tomografia de ondas de superfície na América do Sul: inversão conjunta de velocidade de grupo e forma de onda, Tese de Doutorado, USP/IAG, 122 pp.
- FENG M, ASSUMPÇÃO M & VAN DER LEE S. 2004. Group-velocity tomography and lithospheric S-velocity structure of the South American continent, *Phys. Earth Planet. Inter.* 147: 315–331.
- FRANKLIN JN. 1970. Well-posed stochastic extensions of ill-posed linear problems, *Journal of Mathematical Analysis and Applications* 31: 682–716.
- HESTENES M. 1975. Pseudoinverses and conjugate gradients, *Communications of the ACM* 18(1): 40–43.
- HOROWITZ E & SAHNI S. 1976. *Fundamentals of Data Structures*, Computer Science Press, Inc., 565 pp.
- HUSEN S, KISSLING E & FLUESCH ER. 2000. Local earthquake tomography of shallow subduction in north Chile: a combined onshore and offshore study, *J. Geophys. Res.* 105(B12): 28183–28198.
- ISHII M & TROMP J. 1999. Normal-mode and free-air gravity constraints on lateral variations in velocity and density of Earth's mantle, *Science* 285: 1231–1236.

- LEVENBERG K. 1944. A method for the solution of certain non-linear problems in least squares, *Quarterly Applied Mathematics* II(2): 164–168.
- LÉVÊQUE J-J, RIVERA L & WITTLINGER G. 1993. On the checker-board test to assess the resolution of tomographic inversions, *Geophys. J. Int.* 115: 313–318.
- LINES LR & TREITEL S. 1984. Tutorial A review of least-squares inversion and its application to geophysical problems, *Geophysical Prospecting* 32: 159–186.
- MARQUARDT D. 1963. An algorithm for least-squares estimation of non-linear parameters, *Journal of Society Industry and Applied Mathematics* 11 (2): 431–441.
- MENKE W. 1989. *Geophysical Data Analysis: Discrete Inverse Theory*, Revised Edition, Academic Press, 289 pp.
- MUNKSGAARD N. 1980. Solving sparse symmetric sets of linear equations by preconditioned conjugate gradients, *ACM Transactions on Mathematical Software* 6(2): 206–219.
- PACHECO RP. 2003. Imageamento tridimensional da onda S na litosfera do sudeste brasileiro e adjacências, Tese de Doutorado, MCT/Observatório Nacional, 495 pp.
- PAIGE CC & SAUNDERS MA. 1982a. LSQR: An algorithm for sparse linear equations and sparse least squares, *ACM Transactions on Mathematical Software* 8: 43–71.
- PAIGE CC & SAUNDERS MA. 1982b. ALGORITHM 583 LSQR: Sparse linear equations and least squares problems, *ACM Transactions on Mathematical Software* 8: 195–209.
- PRESS WH, TEUKOLSKY SA, VETTERLING WT & FLANNERY BP. 1992. *Numerical Recipes in FORTRAN: The Art of Scientific Computing*, Second Edition, Cambridge University Press, 963 pp.
- RAY RD. 1985. Correction of systematic error in magnetic surveys: An application of ridge regression and sparse matrix theory, *Geoph.* 63(11): 1721–1731.
- RITZWOLLER MH & LEVSHIN AL. 1998. Eurasian surface wave tomography: Group velocities, *J. Geophys. Res.* 103: 4839–4878.
- SCALES JA. 1987. Tomographic inversion via the conjugate gradient method, *Geophysics* 52: 179–185.
- SOUZA JL de, SANTOS NP dos & PACHECO RP. 2003. Regionalized Rayleigh wave group velocities in southeastern Brazil, EGS-AGU-EUG Joint Assembly, Nice, *Geophys. Res. Abstracts* 5: 1681.
- SOUZA JL de & SANTOS NP dos. 2003. Tomografia com velocidade de grupo de ondas Rayleigh na região sudeste do Brasil, 8th International Congress of the Brazilian Geophysical Society and 5th Latin American Geophysical Conference: 519–524.
- STERLING TL, SALMON J, BECKER DJ & SAVARESE DF. 1999. *How to Build a Beowulf: A Guide to the Implementation and Application of PC Clusters*. The MIT Press, Cambridge, Massachusetts, 2<sup>nd</sup> printing, 239 pp.
- TOOMEY DR & FOULGER GR. 1989. Tomographic inversion of local earthquake data from the Hengill-Grensdalur Central Volcano Complex, Iceland, *J. Geophys. Res.* 94(B12): 17497–17510.
- VASCO DW, PETERSON JR. JE & MAJER EL. 1998. Resolving seismic anisotropy: Sparse matrix methods for geophysical inverse problems, *Geoph.* 63(3): 970–983.
- VDOVIN O, RIAL JA, LEVSHIN AL & RITZWOLLER MH. 1999. Group-velocity tomography of South America and the surrounding oceans, *Geophys. J. Int.* 136: 324–340.
- VILAR CS, SOUZA JL DE & SANTOS NP. 2003. Tomografia com Velocidade de Grupo de Ondas Rayleigh na Região Nordeste do Brasil. 8th International Congress of the Brazilian Geophysical Society and 5th Latin American Geophysical Conference, Rio de Janeiro, 2003. CD-ROM.
- VILAR CS. 2004. Estrutura tridimensional da onda S na litosfera do nordeste brasileiro, Tese de Doutorado, MCT/Observatório Nacional, 258 pp.
- YUAN JY & IUSEM AN. 1996. Preconditioned conjugate gradient method for generalized least squares problems, *Journal of Computational and Applied Mathematics* 71: 287–297.
- ZHANG J & McMECHAN GA. 1995. Estimation of resolution and covariance for large matrix inversions, *Geophys. J. Int.* 121(2): 404–426.

## NOTAS SOBRE OS AUTORES

**Newton Pereira dos Santos.** Engenheiro Cartógrafo pela Universidade do Estado do Rio de Janeiro em 1976. Tecnologista Sênior da Coordenadoria de Geofísica / Observatório Nacional. Mestre em Geofísica pelo Observatório Nacional em 1997. Integrante da equipe responsável pela implantação da Rede Gravimétrica Fundamental Brasileira desde 1980, atuando na área de Gravimetria. Atualmente cursa o doutorado com tese na área de Geodésia Física e integra o grupo de Sismologia no COGE / ON.

**Jorge Luis de Souza.** Graduado em Física pela Faculdade de Humanidades Pedro II, Rio de Janeiro em 1982. Mestrado em Geofísica pelo CNPq / Observatório Nacional, Rio de Janeiro em 1988. Doutorado em Geofísica pelo CNPq / Observatório Nacional, Rio de Janeiro em 1997. Pesquisador da Coordenação de Geofísica do MCT / Observatório Nacional desde 1988.