UNIVERSIDADE FEDERAL DA BAHIA

Curso de Pós-graduação em Geofísica

TESE DE DOUTORADO

INVERSÃO 1,5D E 2D DE DADOS ELETROMAGNÉTICOS E ELÉTRICOS USANDO MÉTODOS DE OTIMIZAÇÃO GLOBAL E PROCESSAMENTO EM PARALELO

JORGE LUIS ABRIL BENJUMEA

SALVADOR – BAHIA
JULHO – 2022

Documento preparado com o sistema LaTeX.

Documento elaborado com os recursos gráficos e de informática do CPGG/UFBA

Inversão 1,5D e 2D de dados eletromagnéticos e elétricos usando métodos de otimização global e processamento em paralelo

por

Jorge Luis Abril Benjumea
Geólogo (Universidad Industrial de Santander – 2014)
Mestre em Geofísica (Universidade Federal da Bahia – 2018)

Orientador: Prof. Dr. Marcos Alberto Rodrigues Vasconcelos

TESE DE DOUTORADO
Submetida em satisfação parcial dos requisitos ao grau de
DOUTOR EM CIÊNCIAS
EM
GEOFÍSICA
ao
Conselho Acadêmico de Ensino
da
Universidade Federal da Bahia

Comissão Examinadora

Walter Medeiros
Dr. Walter Eugênio de Medeiros

Renato RS Dantas
Dr. Renato Ramos da Silva Dantas

Amin Bassrei
Dr. Amin Bassrei

Dr. Milton José Porsani

Dr. Marcos Alberto Rodrigues Vasconcelos

Aprovada em 01 de julho de 2022

A presente pesquisa foi desenvolvida no Centro de Pesquisa em Geofísica e Geologia da UFBA, com recursos próprios, da CAPES

B422 Benjumea, Jorge Luis Abril,

Inversão 1,5D e 2D de dados eletromagnéticos e elétricos usando métodos de otimização global e processamento em paralelo / Jorge Luis Abril Benjumea. — Salvador, 2022.

137 f.

Orientador: Prof. Dr. Marcos Alberto Rodrigues Vasconcelos

Tese (Doutorado) - Universidade Federal da Bahia. Instituto de Geociências, 2022.

Inversão (Geofísica). 2. Otimização global. 3. Eletrorresistividade. 4. Método eletromagnético 5. Processamento paralelo. I. Vasconcelos, Marcos Alberto Rodrigues. II. Universidade Federal da Bahia. III. Título.

CDU 550.8

A minha família dedico inteiramente
este trabalho.
Santandereanos siempre adelante, ni
un paso atrás.

Resumo

O uso de técnicas de otimização global em geofísica para resolver problemas inversos, permitindo incorporar com facilidade diferentes classes de vínculos para criar uma solução matemáticamente estável. A modelagem 1D electromagnética no domínio do tempo (TDEM) e 2D em eletrorresistividade DC é computacionalmente barata, permitindo o uso de métodos de otimização global (GOMs) para resolver problemas inversos 1,5D e 2D, incorporando diferentes regularizadores. No entanto, estas técnicas meta-heurísticas são excessivamente custosas ao enfrentar problemas com muitos parâmetros. Nesse sentido, o uso ferramentas de processamento em paralelo tornam-se indispensável para aliviar o custo computacional do processamento em problemas inversos.

Neste trabalho foi desenvolvida uma nova técnica de otimização de enxame de partículas com elitismo e mutação (EMPSO) usando uma versão paralela híbrida MPI-OpenMP que denominou-se PEMPSO. Este algoritmo, junto com o uso de vínculos de Variação Total (VT) e Suavidade Global (SG), permitiram estudar as variações da distribuição da resistividade 2D em subsuperfície, tanto en dados sintéticos quanto em dados reais. Os resultados da inversão com ambos vínculos obteve uma boa aproximação do modelo verdadeiro, tendo melhor sensibilidade o vínculo de VT na deteção de altos constrastes de resistividade, em relação ao vínculo SG. Os tempos de inversão também demonstraram uma aceleração de 17 vezes nos modelos avaliados, em relação ao tempo de processamento sequencial.

Dos métodos em electrorresistividade 2D, apresenta-se também uma abordagem de inversão 1,5D de dados TDEM, que similarmente ao caso 2D acima mencionado, apresentou bons resultados nos modelos invertidos, usando diferentes técnicas de otimização global tais como Whale Optimization Algorithm (WOA), Particle Swarm Optimization (PSO) e Grey Wolf Optimizer (GWO). Estes algoritmos permitiram mapear e localizar altos e baixos constrastes de resistividade em subsuperfície, tanto em modelos sintéticos quanto em dados reais. Similarmente ao caso 2D, uma abordagem de processamento em paralelo usando MPI puro permitiu acelerar o tempo de computação que envolve o uso destes algoritmos meta-heurísticos. Os principais resultados do caso 1,5D eletromagnético mostram que: (i) ambos vínculos oferecem uma boa aproximação no mapeamento da distribuição de resistividade,

(ii) os algoritmos WOA e PSO oferecem melhor desempenho computacional, convergindo primeiro que GWO, (iii) WOA apresenteu melhor desempenho no valor atingido da função custo que PSO e GWO e (iv) a versão paralelizada MPI pura reflete uma aceleração de 17 vezes no tempo de processamento dos modelos sintéticos e de quase 50 vezes no tempo da computação dos dados reais estudados.

Palavras chaves: inversão 1.5D TDEM, inversão 2D-DC de resistividade, GOMs, vínculos, processamento em paralelo.

Abstract

The use of optimization global techniques in geophysics to solve inverse problems, allowing to incorporate with easiness several kinds of constraints to create an stable mathematically solution. The 1D forward modeling of time domain electromagnetic data (TDEM) and 2D-DC resistivity is computationally inexpensive, allowing to use global optimization methods (GOMs) to solve 1.5D and 2D inverse problems, incorporating several stabilizers. Nonetheless, these meta-heuristic techniques are time-consuming, specially when a high number of parameters is involved. In this sense, the use of paralleling processing tools have become paramount to alleviate the computing time in inverse problems.

In this work was developed a newly particle swarm optimization technique with elitism and mutation (EMPSO) using a hybrid parallel MPI/OpenMP version named (PEMPSO). This algorithm, also supported with the use of Total Variation (TV) and Global Smoothness (GS) constraints allowed studying the resistivity distribution in subsurface, with both synthetic and real data. The inversion outcomes reflect good results for both constraints, having better sensitivity with the Total Variation (TV) constraint in detecting high and low resistivity contrasts. The inversion times, also demonstrated a speedup of 17x in the assessed models in relation with the sequential processing times.

Apart from this 2D electrical resistivity method, we also presented an inversion approach for 1,5D TDEM data, that similarly with the 2D case aforementioned, presented good outcomes in the inverted models using different global optimization techniques such as Whale Optimization Algorithm (WOA), Particle Swarm Optimization (PSO) and Grey Wolf Optimizer (GWO). These algorithms allowed mapping and localizing high and low resistivity contrasts for both synthetic and real data. Similarly with 2D case, an approach of parallel processing using pure MPI allowed to speedup the computational time that involve the use of these meta-heuristic algorithms. The major outcomes of 1,5D electromagnetic case show that: (i) both constraints offer good results to map the resistivity distribution, (ii) the WOA and PSO algorithms offer better computational performance, converging first than the GWO, (iii) WOA presented the better performance in the cost function value attained than PSO and GWO and (iv) a pure MPI parallelized version provided a 17x speedup in the time processing for synthetic models and up to 50x in the time computing of real data studied.

Keywords: 1.5D TDEM inversion, 2D-DC resistivity inversion, GOMs, constraints, parallel processing.

Índice

Índice de Figuras

1

Memorial descritivo

O memorial descritivo contém uma apresentação geral dos métodos de otimização global abordados neste trabalho, os conceitos teóricos, problemas e objetivos de cada método estudado (resistividade 2D e eletromagnético 1,5D no domínio do tempo), uma contextualização da teoria do processamento em paralelo e finalmente um resumo dos principais resultados obtidos dos dois artigos científicos que compõem este trabalho.

1.1 Contexto teórico

1.1.1 Generalidades dos métodos de otimização global

Nesta sub-seção são apresentados os principais aspectos teóricos e equações matemáticas dos métodos de otimização global usados para realizar a inversão de resistividade 2D e eletromagnética no domínio do tempo 1,5D.

Algoritmo Grey Wolf Optimizer (GWO)

Esta técnica prosposta por Mirjalili et al. (2014), emula o comportamento hierárquico e o mecanismo de caça dos lobos cinza na natureza. Esta classificação baseia-se nos lobos alfa, beta, delta e omega. Os lobos alfa estão no comando de guiar o pacote de lobos às regiões promissores de comida. Os lobos beta são subordinados aos alfa, e são os candidatos mais proximos dos lobos alfa quando morrem ou chegam à velhice. Os lobos delta cumpren funções de caça, proteção e guia do pacote, mas são subordinados aos lobos alfa e beta. Finalmente, os lobos omega mudam suas posições segundo as melhores soluções achadas dos lobos alfa, beta e delta. Além de seu comportamento hierárquico, os lobos seguem três mecanismos de caça, que segundo Muro et al. (2011) estão dados por:

- Rastreamento, persecução e focalização da presa.
- Cercamento, circulamento e sufocamento da presa até que para de se mover.
- Ataque à presa.

Para descrever as equações matemáticas do algoritmo, denotamos cada modelo no pacote de lobos em uma iteração particular pelo vetor:

$$\mathbf{m}_i(k) = [m_i^1(k), m_i^2(k), m_i^3(k), \dots, m_i^f(k), \dots, m_i^q(k)]^T, \tag{1.1}$$

sendo q um elemento escalar que envolve o numero de lobos, o subíndice i o número de parâmetros do modelo (N), k é o número da iteração, e k_{max} é o maximo número de iterações.

Para representar os lobos alfa, beta e delta em uma iteração tem-se:

$$\mathbf{m}^l(k) = [m^{l1}(k), m^{l2}(k), m^{l3}(k), \cdots, m^{lf}(k), \dots, m^f(k)]^T, \tag{1.2}$$

sendo $l \in \{\alpha, \beta, \delta\}$.

Os coeficientes de busca podem ser calculados como:

$$\begin{aligned} a_l^f(k) =& a^f(k)(2r_{1l}^f - 1), \\ c_l^f(k) =& 2r_{2l}^f, \quad l \in \{\alpha, \beta, \delta\}, \end{aligned} \tag{1.3}$$

onde r_{1l}^f e r_{2l}^f são dois números aleatórios dentro de [0,1] e $f = 1, \dots ,q$. Os coeficientes $a^f(k)$ são linearmente diminuídos de 2 até 0 durante os processo de busca como segue:

$$a_l^f(k) = 2 - \frac{2k}{(k_{max})}, \quad f = 1, 2, \dots ,q. \tag{1.4}$$

Logo, é possível calcular as distâncias entre a melhor posição atual e as melhores posições α, β, δ como segue:

$$\begin{aligned} d_i^{lf} =& |c_i^{lf} m_{lf}(k) - m_i^f(k)|, \\ i =& 1, 2, \dots, N, \quad l \in \{\alpha, \beta, \delta\}. \end{aligned} \tag{1.5}$$

Calculadas as distâncias, pode-se computar as melhores posições dos lobos α, β, δ, dadas por:

$$\begin{aligned} m^{lf}(k+1) =& m^{lf}(k) - a_l^f(k) d^{lf}(k) \quad f = 1, \dots, q; \\ i =& 1, 2, \dots, N; \quad l \in \{\alpha, \beta, \delta\}. \end{aligned} \tag{1.6}$$

Por último, calcula-se o vetor posição final usando a média aritmética das atualizações dos vetores α, β, δ, dada por:

$$\begin{aligned} \mathbf{m}_i(k+1) =& \frac{\mathbf{m}^{\alpha}(k+1) + \mathbf{m}^{\beta}(k+1) + \mathbf{m}^{\delta}(k+1)}{3}, \\ i =& 1, \dots, N. \end{aligned} \tag{1.7}$$

O coeficiente $c_l^f(k)$ fornece valores aleatórios dentro de [0,2] à presa para que possa decidir com que facilidade envolve a mesma. Se o valor é menor do que 1 favorece o mecanismo de caça (fase de explotação), e se é maior do que 1 favorece o mecanismo de busca pela presa (fase de exploração). Este comportamento evita que a presa fique atrapalhada em mínimos locais.

A presa é atacada quando para de se movimentar, permitindo o ataque dos lobos cinza. Durante a última iteração quando $a^f(k)$ se torna 0, todos os elementos do coeficiente de busca $a_l^f(k)$ se tornam 0. O ataque da presa é matemáticamente modelado diminuindo o valor de $a^f(k)$, que também diminui o intervalo de variação do $a_l^f(k)$. Similarmente às técnicas de otimização por enxame, no GWO os lobos atualizam suas posições baseados em suas posições anteriores. Este mecanismo envolve uma troca de informações em meio a todos os indivíduos, que guiam cada indivíduo até posições de melhor ajuste.

Algoritmo Particle Swarm Optimization (PSO)

Proposto por Kennedy e Eberhart (1995), o algoritmo de otimização por enxame de partículas (PSO) é um dois métodos de otimização global mais conhecidos. A ideia geral do PSO é inspirada pelo comportamento social observado em muitas espécies da natureza, como bandos de pássaros ou cardume de peixes que trabalham coletivamente na busca de alimento ou na sobrevivência ao ataque de um predador. Este algoritmo emula a interação entre os membros para compartilhar informações. Cada individuo do grupo tem sua própria experiência e é capaz de estimar a qualidade desta experiência, além de ter comportamento social; conhecimento sobre como seus vizinhos cercanos se comportam. Estes dois tipos de informação correspondem ao comportamento individual (cognitivo) e o comportamento social. Por tanto, a probalidade que um individuo tome uma decisão vai depender de seu desempenho no passado e o desempenho dos seus vizinhos Kennedy (2006).

No PSO, as potenciais soluções, chamadas partículas, movem-se no espaço do problema $\mathbb{R}^n$, onde n é o tamanho do espaço. Uma partícula i é definida por seu vetor posição, $\mathbf{x}_i = x_{i1},x_{i2},\ldots,x_{in}$, e seu vetor velocidade, $\mathbf{v}_i = v_{i1},v_{i2},\ldots,v_{in}$. O desempenho de cada partícula é medido segundo seu desempenho, mediante a avaliação da sua função custo ou ajuste, que reflete a qualidade da posição da partícula. A melhor localização alcançada pela i-ésima partícula (conhecida como melhor pessoal) é representada por $\mathbf{p}_i = p_{i1},p_{i2},\ldots,p_{in}$, enquanto a melhor localização alcançada pelas partículas vizinhas (conhecida como melhor local) é representada por $\mathbf{g}_i = g_{i1},g_{i2},\ldots,g_{in}$. A cada iteração k, cada partícula muda sua posição de acordo com a nova velocidade como em

$$\mathbf{v_i}^{k+1} = w^k \mathbf{v_i}^k + c_1 \mathbf{u}_1 \otimes (\mathbf{p}_i - \mathbf{m}_i{}^k) + c_2 \mathbf{u}_2 \otimes (\mathbf{g}_i - \mathbf{m}_i{}^k) \tag{1.8}$$

$$\mathbf{m_i}^{k+1} = \mathbf{m_i}^k + \mathbf{v_i}^{k+1} \tag{1.9}$$

onde $i = 1,2,\ldots,n_{part}$, $\mathbf{p}_i$ e $\mathbf{g}_i$ denotam a melhor posição de partícula e a melhor posição da vizinhança e os parâmetros w, $c1$, $c2$, $\mathbf{u}_1$ e $\mathbf{u}_2$ são, respectivamente, o fator de inércia, duas constantes positivas e dois vetores de números aleatórios dentro de [0, 1].

Algoritmo Whale Optimization Algorithm (WOA)

Inicialmente descrito por Mirjalili e Lewis (2016), o algortimo Whale Optimization Algorithm emula o comportamento de caça das baleias jubarte na natureza para procurar alimento. Este comportamento está baseiado em três estratégias como são: (i) Cerca caótica da presa, (ii) Ataque de rede de burbulhas e (iii) Busca pela presa. Além destas estratégias, são consideradas duas fases que são: exploração (busca pela presa) e explotação (cercamento e caça por agressão de rede de burbulhas). Similar ao esquema usado no algoritmo GWO, denota-se cada modelo de baleias em uma iteração particular, conforme o vetor

$$\mathbf{m}_i(k) = [m_i^1(k), m_i^2(k), m_i^3(k), \ldots, m_i^f(k), \ldots, m_i^q(k)]^T, \tag{1.10}$$

sendo q um elemento escalar que envolve o numero de baleias, o subíndice i é o número de parâmetros do modelo (N) e k uma iteração dada.

Para o mecanismo de cercamento da presa a melhor solução é a que se encontra mais perto ao objetivo (presa). Para atualizar as posições dos agentes, a melhor posição do agente é atualizada segundo as equações:

$$\begin{aligned} d_i^f =& |c_i^f m_f^*(k) - m_i^f(k)|, \\ i =& 1, 2, \ldots, N, \quad f = 1, \ldots, q. \end{aligned} \tag{1.11}$$

$$\begin{aligned} m^f(k+1) =& m_f^*(k) - a^f(k) d^f(k), \\ i =& 1, 2, \ldots, N, \quad f = 1, \ldots, q. \end{aligned} \tag{1.12}$$

Em cada iteração k, $m_f^*(k)$ só é atualizado se houver uma melhor solução. Os coeficientes a e c estão dados por:

$$\begin{aligned} a^f(k) =& a^f(k)(2r_1^f - 1), \\ c_l^f(k) =& 2r_2^f, \end{aligned} \tag{1.13}$$

onde r_1^f e r_2^f são dois números aleatórios dentro de [0,1] e $f = 1, \ldots, q$.

A estratégia de ataque por rede de burbulhas é efetivada seguindo três fases que são a redução do mecanismo de encerramento, a atualização em espiral e a escolha de atacar. As fases são descritas como:

- Redução do mecanismo de encerramento: Neste caso o valor de $a^f_(k)$ é diminuído de 2 a 0, permitindo uma melhor proximidade do agente atual à melhor posição do agente.
- Atualização em espiral: Emula a forma do movimento em hélice das baleias onde é feita a busca da posição da baleia seguindo:

$$\begin{aligned} m^f(k+1) =& d_i^{'f} \cdot e^{bl} \cdot cos 2\pi l + m_f^*(k), \\ i =& 1, 2, \ldots, N, \quad f = 1, \ldots, q. \end{aligned} \tag{1.14}$$

$$\begin{aligned} d_i^{'f} =& |m_f^*(k) - m_i^f(k)|, \\ i =& 1, 2, \ldots, N, \quad f = 1, \ldots, q. \end{aligned} \tag{1.15}$$

onde $d_i^{'f}$ representa a distância entre o agente de busca atual e a melhor solução obtida até o momento; b é um escalar que indica a forma do espiral logaritmico, e l é um numero aleatório entre 0 e 1.

- Escolha de atacar: Este comportamento envolve 50% um comportamento de cercamento das presas e o outro 50% um comportamento de movimentos em espiral. Estes comportamentos são matemáticamente modelados por:

$$m^f(k+1) = \begin{cases} m_f^*(k) - a^f(k)d_i^{'f} & \text{se } p \leq 0.5 \\ d_i^{'f} \cdot e^{bl} \cdot cos2\pi l + m_f^*(k) & \text{se } p \geq 0.5 \end{cases} \tag{1.16}$$

sendo p um número aleatório distribuído entre 0 e 1, que indica a probabilidade de escolher entre cercamento da presa ou movimento em espiral.

Por último, o mecanismo de busca pela presa, obedece à exploração e explotação do algoritmo, onde o valor de $a^f_(k)$ é usado para atacar ou evitar a presa. Se a_k^f é maior do que 1 é feita a busca pela presa, do contrário se faz o ataque por rede de burbulha. Este comportamento de busca é matemáticamente simulado por:

$$\begin{aligned} d_i^f =& |c_i^f m_f^{rand}(k) - m_i^f(k)|, \\ i =& 1, 2, \ldots, N, \quad f = 1, \ldots, q. \end{aligned} \tag{1.17}$$

$$\begin{aligned} m^f(k+1) =& m_f^{rand}(k) - a^f(k)d^f(k). \\ i =& 1, 2, \ldots, N, \quad f = 1, \ldots, q. \end{aligned} \tag{1.18}$$

sendo $m_f^{rand}(k)$ é a posição randômica gerada para favorecer a exploração do algoritmo e d_i^f é a distância entre o valor randômico gerado e a melhor posição da presa.

1.1.2 Método de resistividade 2D

A técnica de eletrorresistividade 2D é um dos métodos mais comuns na geofísica aplicada, e que oferece um excelente custo-benefício na exploração geofísica. Diferentes técnicas têm sido usadas para calcular a resposta de resistividade aparente da subsuperfície ρ_a, tais como a técnica da equação integral implementada por Dieter et al. (1969) e Hohmann (1975), aplicada para geometrias simples. Outros como (Sasaki, 1994; Ellis e Oldenburg, 1994; Zhang et al., 1995; Marescot et al., 2008; Papadopoulos et al., 2007) implementam a técnica de elementos finitos para obter a propriedade física de interesse. Finalmente, outros trabalhos como (Mufti (1976), Dey e Morrison (1979), Lowry et al. (1989), (Park e Van, 1991), Spitzer (1995)) implementaram a técnica de diferenças finitas, que foi escolhida como parte esencial do programa de inversão que se desenvolveu.

Contexto teórico do algoritmo de modelagem direta 2D

A modelagem direta de resistividade 2D baseia-se em relações fundamentais tais como a lei de Ohm e o princípio de conservação da carga. A lei de Ohm relaciona a densidade de corrente **J** ao campo elétrico **E** e a condutividade isotropica dada por:

$$\mathbf{J} = \sigma \mathbf{E}. \tag{1.19}$$

Considerando campos elétricos estacionários e conservativos tem-se:

$$\mathbf{E} = -\nabla \phi, \tag{1.20}$$

onde ϕ é o potencial elétrico. Substituindo a equação 1.20 na equação 1.19 tem-se:

$$\mathbf{J} = -\sigma \nabla \phi. \tag{1.21}$$

Aplicando o princípio da conservação da carga sob o volume, se tem a equação de continuidade:

$$\nabla \cdot \mathbf{J} = \frac{\partial q}{\partial t} \delta(x_s)\delta(y_s)\delta(z_s), \tag{1.22}$$

onde q é a densidade carga no punto x, y e z.

A equação acima pode ser escrita para um espaço 3D como:

$$\nabla \cdot [\sigma(x,y,z) \nabla \phi(x,y,z)] = \frac{\partial q}{\partial t} \delta(x_s)\delta(y_s)\delta(z_s), \tag{1.23}$$

onde (x_s,y_s,z_s) são as coordenadas da fonte pontual no espaço x, y, z. Considerando que a condutividade não muda na direção y, a equação acima pode ser rescrita como:

$$\nabla \cdot [\sigma(x,z)\nabla\phi(x,y,z)] = \frac{\partial q}{\partial t}\delta(x_s)\delta(y_s)\delta(z_s). \tag{1.24}$$

Na equação acima o potencial ϕ e o termo fonte são funciones de x, y, z mas a condutividade σ é só função de x, e z. Dey e Morrison (1979) propuseram resolver a equação no domínio da transformada espacial de Fourier (x,k_y,z), transformando y em k_y. Esta transformação obedece à seguinte equação:

$$\tilde{f}(x,K_y,z) = \int_0^\infty f(x,y,z)cos(k_y y)dy, \tag{1.25}$$

e sua transformada inversa dada por:

$$f(x,k_y,z) = \int_0^\infty \tilde{f}(x,k_y,z)cos(k_y y)dk_y, \tag{1.26}$$

onde $f(x,y,z)$ e $\tilde{f}(x,K_y,z)$ são funções pares de y.

Aplicando a equação 1.25 na equação 1.24 tem-se:

$$-\nabla \cdot [\sigma(x,z)\nabla\tilde{\phi}(x,K_y,z)] + K_y^2\sigma(x,z)\tilde{\phi}(x,K_y,z) = \tilde{Q}\delta(x_s)\delta(z_s), \tag{1.27}$$

onde $\tilde{\phi}(x,K_y,z)$ é o potencial transformado e $\tilde{Q}$ é a densidade corrente em estado estacionário no espaço x, k_y e z.

A densidade de corrente é representada por:

$$\tilde{Q}\delta(x_s)\delta(z_s) = \frac{1}{2}\frac{\partial q}{\partial t}\delta(x_s)\delta(z_s) = \frac{I}{2\Delta A}, \tag{1.28}$$

onde ΔA é area no plano x, y, z da injeção de corrente x_s,z_s. O potencial $\tilde{\phi}$ pode ser obtido resolvendo a equação 1.27 sob condições de contorno mistas (Dey e Morrison, 1979). Logo, usando a equação 1.26 podes-se transformar o potencial transformado $\tilde{\phi}$ em ϕ.

A forma geral da condição de contorno mista usada por Dey e Morrison (1979) é dada por:

$$\alpha(x{,}z)\tilde{\phi} + \beta(x{,}z)\frac{\partial\tilde{\phi}}{\partial\eta} = 0, \tag{1.29}$$

onde $\alpha = 0$ e $\beta = 0$, considerando a superfície do terreno e para as bordas do modelo $\beta = 1$ e α é dado por:

$$\alpha = \frac{k_y K_1(k_y.r)}{K_0(k_y.r)}, \tag{1.30}$$

sendo k_y o número de onda espacial, r é a distância da fonte, K_0 é a função de Bessel modificado de ordem zero e K_1 é a função de Bessel modificada de ordem 1.

Para discretizar a malha de diferenças finitas é necessário estabelecer os limites do modelo. Segundo a Figura 1.1, o modelo é dividido em uma área de interesse e em uma área estendida.

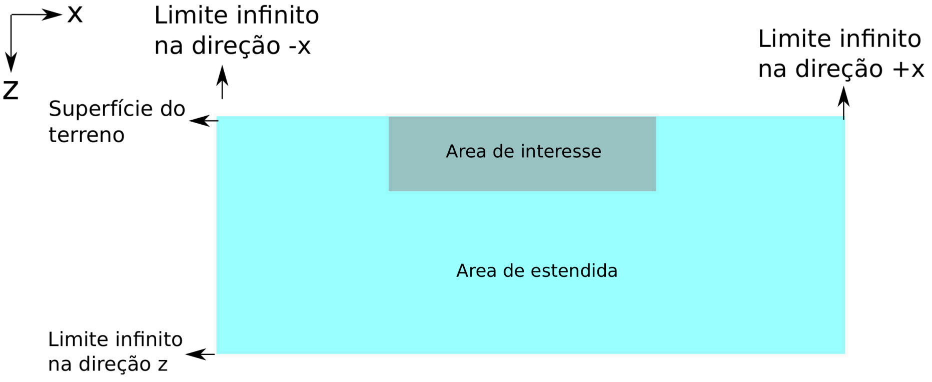

Figura 1.1: Modelo do semi-espaço e limites de contorno. Adaptado de Vachiratienchai, (2007).

A resposta é só calculada na área de interesse. No entanto, o domínio do modelo deve ser estendido no plano infinito com o intuito de reduzir o efeito das bordas. Para calcular o potencial ϕ, o meio espaço deve ser discretizado em blocos rectangulares. A descrição detalhada do $\sigma_{i,j}$, a área da malha $\Delta A_{i,j}$, a linha de contorno $L_{i,j}$ sob o nó (i,j) no interior da malha é ilustrado na Figura 1.2 (Vachiratienchai, 2007).

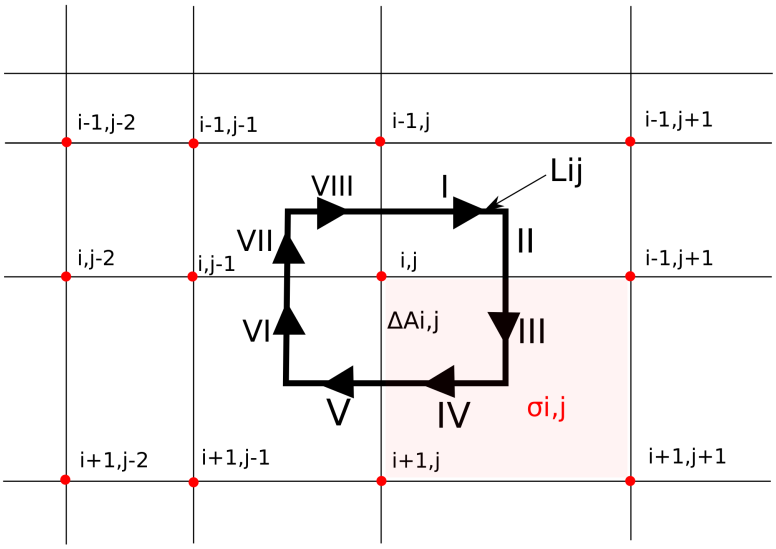

Figura 1.2: Discretização rectangular da malha e elemento de área discretizado $\Delta A_{i,j}$ ao redor do nó (i,j). Adaptado de Vachiratienchai, (2007).

A condutividade $\sigma_{i,j}$ indica a condutividade na região limitada pelos nós (i,j), (i+1,j), (i+1)(j+1) e (i,j+1). O nó (i,j) é definido nos cantos e limitado pela região da malha $\Delta A_{i,j}$ sob o nó, como mostrado na Figura 1.2. Para um ponto no interior do nó, $\Delta A_{i,j}$ é dado por:

$$\Delta A_{i,j} = \frac{(\Delta x_j + \Delta x_{j-1})(\Delta z_i + \Delta z_{i-1})}{4}, \tag{1.31}$$

e na superfície do terreno a expressão de $\Delta A_{i,j}$ é dada por:

$$\Delta A_{i,j} = \frac{(\Delta x_j + \Delta x_{j-1})(\Delta z_i)}{4}. \tag{1.32}$$

Para cada nó (i,j) no qual o potencial $\tilde{\phi}_{i,j}$ é desconhecido, a equação 1.27 é integrada no domínio da malha $\Delta A_{i,j}$ para obter:

$$\iint_{\Delta A_{i,j}} \nabla \cdot \sigma_{i,j} \nabla \tilde{\phi}_{i,j} dx_j dz_i + \iint_{\Delta A_{i,j}} k_y^2 \sigma_{i,j} \tilde{\phi}_{i,j} dx_j dz_j = \iint_{\Delta A_{i,j}} \tilde{Q} \delta(x_s) \delta(z_s) dx_j dz_i. \tag{1.33}$$

substituindo a equação 1.28 na equação 1.33 tem-se:

$$\iint_{\Delta A_{i,j}} \nabla \cdot \sigma_{i,j} \nabla \tilde{\phi}_{i,j} + \iint_{\Delta A_{i,j}} k_y^2 \sigma_{i,j} \tilde{\phi}_{i,j} dx_j dz_j = \frac{I}{2} \delta(x_s)\delta(z_s) dx_j dz_i. \tag{1.34}$$

Aplicando o teorema de Green ao primeiro termo da equação 1.33 tem-se:

$$\iint_{\Delta A_{i,j}} \nabla \cdot (\sigma_{i,j} \nabla \tilde{\phi}_{i,j}) dA = \iint_{L_{i,j}} \sigma_{i,j} \frac{\partial \tilde{\phi}_{i,j}}{\partial \eta} dl, \tag{1.35}$$

onde η é a direção normal externa e $L_{i,j}$ é a integral de linha de contorno na região da malha $\Delta A_{i,j}$. Da equação 1.35 a condição de contorno da equação 1.29 pode ser aplicada ao primeiro termo da equacao 1.34 discretizada. A integral de contorno $L_{i,j}$ pode ser subdividida em 8 partes como indicado na Figura 1.2. Integrando-se ao longo do caminho $L_{i,j}$ e aproximando-se $\frac{\partial \phi}{\partial \eta}$, tem-se:

$$\begin{aligned} \iint_{L_{i,j}} \sigma_{i,j} \frac{\partial \tilde{\phi_{i,j}}}{\partial \eta} dl = & \frac{\Delta x_j \sigma_{i-1,j}}{2} (\frac{\tilde{\phi}_{i-1,j} - \tilde{\phi}_{i,j}}{\Delta z_{i-1,j}}) + \frac{\Delta z_{i-1} \sigma_{i-1,j}}{2} (\frac{\tilde{\phi}_{i,j+1} - \tilde{\phi}_{i,j}}{\Delta x_j}) \\ & + \frac{\Delta z_i \sigma_{i,j}}{2} (\frac{\tilde{\phi}_{i,j+1} - \tilde{\phi}_{i,j}}{\Delta x_j}) + \frac{\Delta x_i \sigma_{i,j}}{2} (\frac{\tilde{\phi}_{i-1,j-1} - \tilde{\phi}_{i,j}}{\Delta z_i}) \\ & + \frac{\Delta x_{j-1} \sigma_{i,j-1}}{2} (\frac{\tilde{\phi}_{i-1,j-1} - \tilde{\phi}_{i,j}}{\Delta z_i}) + \frac{\Delta z_i \sigma_{i,j-1}}{2} (\frac{\tilde{\phi}_{i,j-1} - \tilde{\phi}_{i,j}}{\Delta x_{j-1}}) \\ & + \frac{\Delta z_{j-1} \sigma_{i-1,j-1}}{2} (\frac{\tilde{\phi}_{i,j-1} - \tilde{\phi}_{i,j}}{\Delta x_{j-1}}) + \frac{\Delta x_{j-1} \sigma_{i-1,j-1}}{2} (\frac{\tilde{\phi}_{i-1,j} - \tilde{\phi}_{i,j}}{\Delta z_{j-1}}) \end{aligned} \tag{1.36}$$

Similarmente, o segundo termo do lado direito da equação 1.34 pode ser expandido como:

$$\begin{aligned} \iint_{\Delta Ai,j} k_y^2 \sigma_{i,j} \tilde{\phi}_{i,j} dx_j dz_i = k_y^2 \tilde{\phi_{i,j}} [& \frac{\sigma_{i-1,j-1} \Delta x_{j-1} \Delta z_{i-1}}{4} + \frac{\sigma_{i,j-1} \Delta x_{j-1} \Delta z_i}{4} \\ + \frac{\sigma_{i,j} \Delta x_j \Delta z_i}{4} + \frac{\sigma_{i-1,j} \Delta x_j \Delta z_{i-1}}{4}] \equiv \Lambda(\sigma_{i,j}, A_{i,j} \tilde{\phi}_{i,j}). \end{aligned} \tag{1.37}$$

Susbtituindo a equação 1.36 na equação 1.34, obtem-se:

$$C_L^{ij} \tilde{\phi}_{i,j-1} + C_R^{ij} \tilde{\phi}_{i,j+1} + C_T^{ij} \tilde{\phi}_{i-1,j} + C_B^{ij} \tilde{\phi}_{i+1,j} + C_P^{ij} \tilde{\phi}_{i,j} = \frac{I}{2} \delta(x_s)\delta(z_s), \tag{1.38}$$

onde

$$
\begin{aligned}
C_L^{ij} &= -[\frac{\Delta z_{i-1}\sigma_{i-1,j-1} + \Delta z_i \sigma_{i,j-1}}{2\Delta x_{j-1}}],\\
C_R^{ij} &= -[\frac{\Delta z_{i-1}\sigma_{i-1,j} + \Delta z_i \sigma_{i,j}}{2\Delta x_j}],\\
C_T^{ij} &= -[\frac{\Delta x_{j-1}\sigma_{i-1,j-1} + \Delta x_j \sigma_{i-1,j}}{2\Delta z_{i-1}}],\\
C_B^{ij} &= -[\frac{\Delta x_{j-1}\sigma_{i,j-1} + \Delta x_j \sigma_{i,j}}{2\Delta z_i}],\\
C_P^{ij} &= -[C_L^{ij} + C_R^{ij} + C_T^{ij} + C_B^{ij} - \Lambda(\sigma_{i,j}, A_{i,j})],\\
\Lambda(\sigma_{i,j}, A_{i,j}) &= [\frac{\sigma_{i-1,j-1}\Delta x_{j-1}\Delta z_{i-1}}{4} + \frac{\sigma_{i,j-1}\Delta x_{j-1}\Delta z_i}{4},\\
\frac{\sigma_{i,j}\Delta x_{j-1}\Delta z_i}{4} &+ \frac{\sigma_{i-1,j}\Delta x_j \Delta z_{i-1}}{4}].
\end{aligned}
\tag{1.39}
$$

Os parâmetros C_L^{ij}, C_T^{ij}, C_B^{ij} e C_B^{ij} são os coeficientes de acoplamento entre os nós (i,j), (i,j+1), (i-1,j) e (i+1,j), respectivamente, entanto C_P^{ij} é o coeficiente de auto acoplamento no nó (i,j). Estes coeficientes de acoplamento são funções conhecidas da geometria e a propriedade física. Desta forma, aplicando a equação 1.38 a todos os nós, pode-se obter um conjunto de equações simultâneas que pode ser escrito como:

$$\mathbf{C}\tilde{\phi} = \mathbf{s}, \tag{1.40}$$

onde $\mathbf{C}$ é a matriz capacitância, $\tilde{\phi}$ é o vetor potencial e $\mathbf{s}$ é o vetor das fontes.

Inversão de dados de resistvidade 2D

O problema inverso de resistividade 2D tem sido amplamente estudado na geofísica de exploração usando métodos de otimização local e global. Neste trabalho, foi usado um esquema de inversão global usando o algoritmo Particle Swarm Optimization (PSO). No entanto, o uso de métodos de otimização global (GOMs), assim como qualquer outro método estocástico, apressenta algumas disvantagens que envolvem a baixa convergência, devido ao entrampamento em mínimos locais falsos. Para contornar este problema, se incorporou uma estratégia de exploração elitista-mutado, designada de EMPSO por Nagesh Kumar e Janga Reddy (2007). Nessa estratégia, todas as partículas são classificadas em ordem crescente com base em sua função objetivo e os números de índice para as respectivas partículas são obtidos (isto é, o índice do enxame classificado é registrado num vetor $ASF[i], i = 1, \ldots, n_{part}$). Logo, a mutação elitista é realizada em um número predeterminado NM das piores partículas (de acordo com a classificação) e os respectivos vetores de posição das partículas são substituídos pelos novos vetores de posição mutados obtidos após a execução da mutação em algumas variáveis no melhor vetor de posição global (segundo uma probabilidade de mutação P_{em}). Enquanto isso, os vetores de velocidade dessas partículas permanecem invariáveis.

Além da estratégia elitista, incluímos o processamento paralelo para aumentar a eficiência do algoritmo PSO, pois como é bem sabido, GOMs consomem muito tempo devido ao grande número de iterações necessárias para sua convergência. Uma situação que pode ainda se agravar em problemas envolvendo um elevado número de variáveis, pois isto leva diretamente ao aumento do custo computacional na avaliação da função objetivo. Graças ao seu inerente paralelismo, e ao crescimento do poder computacional nos últimos tempos, numerosos estudos foram realizados sobre a paralelização do algoritmo PSO (Lalwani et al., 2019). Nosso plano de paralelização usa paralelismo de dados com divisão de tarefas em sistemas distribuídos, incluindo processadores *multi-core*. Neste paradigma de programação foram incorporadas rotinas na linguagem Fortran que usam a biblioteca *Message Passing Interface (MPI)* para expandir a computação atraves de múltiplos processadores (Nós computacionais). Além disso, uma versão híbrida *MPI + OpenMP* (*Open Multi-Processing*) do código foi implementada, adicionando directivas do *OpenMP* em trechos específicos do código (laços específicos). O código paralelo usando *MPI* se espelha na implementação do algoritmo genético de Metcalfe e Charbonneau (2003) para aplicações em astrofísica, o qual usa uma partição dinâmica das tarefas, isto é, atribuição das tarefas aos processos MPI sob demanda.

1.1.3 Método eletromagnético no domínio do Tempo (TDEM)

O método eletromagnético no domínio do tempo (TDEM) teve seu origem na união soviética nos anos 60, mas seu uso maior uso começou na decada dos 80, sendo considerado ums dos métodos eletromagnéticos mais novos comparado com o uso dos métodos elétricos e eletromagnéticos no domínio da frequência. Estes últimos métodos, possuem uma grande limitação comparado aos métodos no domínio do tempo, pois o campo magnético secundários (referente aos materiales da subsuperfície) são medidos na presença do campo primário (gerado pela bobina transmissora). Toda vez que o campo magnético primário é maior que o secundário, a remoção dele constitui uma tarefa complicada que induz muitas vezes perda da informação do campo secundário. Para contornar esta perda de informação, é possível gerar um campo pulsado ao invés de um contínuo para medir o campo secundário, uma vez que o primário é desligado (Bortolozo, 2016).

O princípio de funcionamento do método TDEM faz uso desta metodologia, usando uma campo transiente ao invés de uma contínua. Para isso, é usada uma bobina trasmissora e uma receptora. Pela bobina transmissora transita uma corrente contínua que gera um fluxo magnético constante. Este fluxo magnético viaja por qualquer condutor na subsuperfície, mas não gera correntes no condutor, pois o fluxo é constante. De repente a corrente contínua é desligada e este fluxo cai do seu valor inicial para zero, durante um pequeno período de tempo. Desta forma, segundos a lei de Faraday, são geradas correntes secundárias, como mostrado na Figura 1.3 adaptada de Bortolozo (2016). O campo secundário decai com o tempo, uma vez que as correntes são dissipadas na subsuperfície. Esse campo que depende do tempo induz uma força eletromotriz (f.e.m) transiente na bobina receptora. Conforme a dissipação do campo magnético na subsuperfície varia segundo a resistência elétrica do condutor no subsolo, é possivel determinar as carateristicas dos materiales geológicos abaixo do arranjo específico.

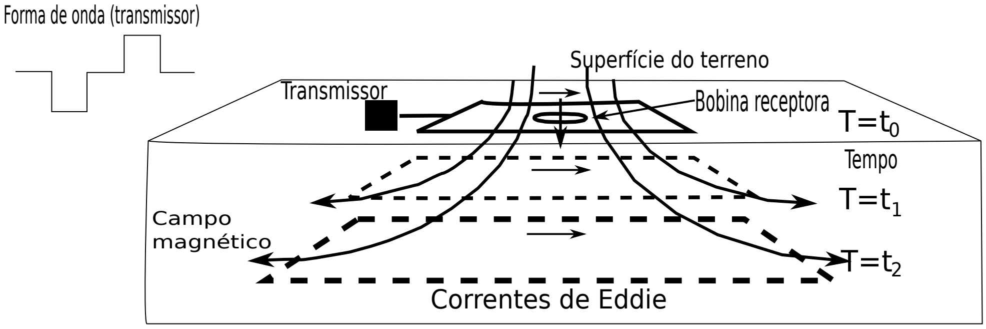

Figura 1.3: Forma da correnta injetada na bobina transmissora, Adaptado de Bortolozo, (2016).

A corrente primária associada com o fluxo não são desligados instantaneamente, pois o fluxo sai do seu valor constante $\mathbf{B}_0$ até zero linearmente com o tempo, em um pequeno intervalo de tempo de ordem de microsegundos τ. Neste sentido, qualquer variação do fluxo e sua força eletromotriz são proporcionais a $\mathbf{B}_0$ e proporcionais a qualquer corrente primária. Além disso, o campo secundário que age na bobina receptora e sua f.e.m induzida são proporcionais à corrente primária da bobina transmissora. A figura 1.4 ilustra este procedimento de aquisição dos dados TDEM, onde processo é feito em inumeráveis ciclos, para diminuir a razão sinal/ruído e melhorar a qualidade dos dados.

Formulação básica do método (TDEM)

A seguir descreve-se a formulação básica para o método TDEM em um meio estratificado 1D com uma bobina transmissora na forma de dipolo magnético vertical. Esta formulação

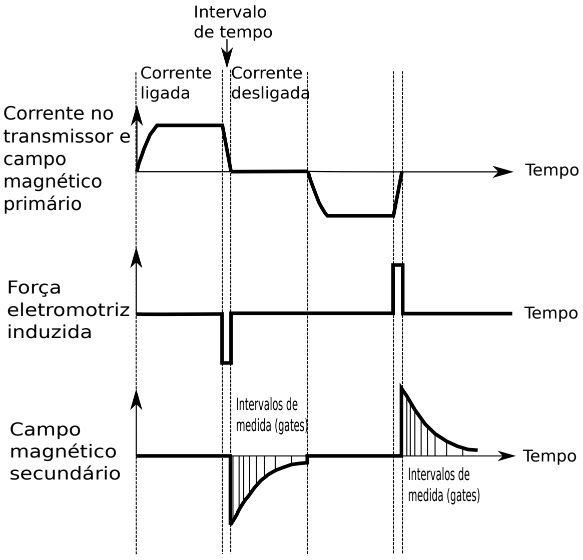
Intervalo de tempo
Corrente ligada
Corrente desligada
Corrente no transmissor e campo magnético primário
Tempo
Força eletromotriz induzida
Tempo
Campo magnético secundário
Intervalos de medida (gates)
Tempo
Intervalos de medida (gates)

Figura 1.4: Geração do campo magnético e as correntes de Eddie. Adaptado de Bortolozo, (2016).

obedece à teoria eletromagnética pelas equações de Maxwell dadas por:

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \tag{1.41}$$

$$\nabla \times \mathbf{H} - \frac{\partial \mathbf{D}}{\partial t} = \mathbf{j} \tag{1.42}$$

$$\nabla \cdot \mathbf{B} = 0 \tag{1.43}$$

$$\nabla \cdot \mathbf{D} = q \tag{1.44}$$

com as suas relações constitutivas:

$$\mathbf{J} = \sigma \mathbf{E} \tag{1.45}$$

$$\mathbf{D} = \epsilon \mathbf{E} \tag{1.46}$$

$$\mathbf{B} = \mu \mathbf{H} \tag{1.47}$$

onde

E é a intensidade do campo elétrico, **B** é a indução magnética, **H** é a intensidade do campo magnético, **D** é o vetor densidade do fluxo elétrico, **j** é a densidade de corrente elétrica, q é a densidade de carga, ϵ é a permissividade elétrica, σ é a condutividade elétrica, e μ é a permeabilidade magnética no vácuo.

A partir do uso da Transformada de Fourier nas equações 1.41 a 1.44 e fazendo uso das relações constitutivas das equações 1.45 a 1.47, obtém-se as equações de Maxwell no domínio da frequência representadas por:

$$\nabla \times \mathbf{E} + i\omega\mu\mathbf{H} = 0 \tag{1.48}$$

$$\nabla \times \mathbf{H} - (\sigma + i\epsilon\omega)\mathbf{E} = 0 \tag{1.49}$$

Nas equações acima são introduzidos os termos de impeditividade e admisividade, sendo os termos $\hat{z} = i\omega\mu_0$ e $\hat{y} = \sigma + i\epsilon\omega$ respectivamente, segundo Harrington (1961). As equações 1.48 e 1.49 só se aplicam para regiões livres de carga, por isso são substituídas pelas seguintes equações não homogêneas:

$$\nabla \times \mathbf{E} + \hat{z}\mathbf{H} = -\mathbf{J}_m^S, \tag{1.50}$$

$$\nabla \times \mathbf{H} - \hat{y}\mathbf{E} = \mathbf{J}_e^S, \tag{1.51}$$

com $\mathbf{J}_m^S$ sendo a corrente de fonte magnética e $\mathbf{J}_e^S$ a corrente de fonte elétrica.

Nas equações 1.50 e 1.51 os vetores **E** e **H** podem ser expressados segundo os potenciais de Schelkunoff, **A** e **F** para facilitar a derivação. Dado que o método TDEM usa só a fonte magnética que transmitte um campo elétrico transversal, só o potencial **F** é necessário. Este potencial é representado como:

$$\mathbf{E}_m = -\nabla \times \mathbf{F}, \tag{1.52}$$

sendo $\mathbf{E}_m$ o campo elétrico gerado pela fonte magnética. Usando a equação 1.50 pode-se obter a equação não homogênea de Helmholtz definida como:

$$\nabla^2\mathbf{F} + k^2\mathbf{F} = -\mathbf{J}_m^S, \tag{1.53}$$

com o número de onda k definido como:

$$k^2 = \omega^2\mu_0\epsilon - i\omega\mu_0\sigma, \tag{1.54}$$

$$k^2 = -\hat{z}\hat{y}. \tag{1.55}$$

Usando uma aproximação quase-estática onde as correntes de deslocamento se tornam desprezíveis, os campos elétrico e magnético gerados pela fonte magnética usando o potencial **F** podem ser calculados como:

$$\mathbf{E}_m = -\nabla \times \mathbf{F}, \tag{1.56}$$

sendo o número de onda k definido por:

$$\mathbf{H}_m = -\hat{y}\mathbf{F} + \frac{1}{\hat{z}}\nabla(\nabla \cdot \mathbf{F}). \tag{1.57}$$

Substituindo a equação 1.57 em 1.55 e 1.56 obtém-se os campos, dadas por:

$$H_x = \frac{1}{\hat{z}}\frac{\partial^2 F_z}{\partial x \partial z}, \tag{1.58}$$

$$E_x = -\frac{\partial F_z}{\partial y}, \tag{1.59}$$

$$H_y = \frac{1}{\hat{z}}\frac{\partial^2 Fz}{\partial x \partial z}, \tag{1.60}$$

$$E_y = \frac{\partial F_z}{\partial x}, \tag{1.61}$$

$$H_z = \frac{1}{\hat{z}}\left(\frac{\partial^2}{\partial z^2} + k^2\right)F_z, \tag{1.62}$$

$$E_z = 0. \tag{1.63}$$

No método TDEM, o *loop* transmissor produz um campo elétrico transversal, fazendo-se necessário calcular o campo magnético vertical. Este campo pode ser calcular como uma integração de dipolos magnéticos verticais sobre a área da bobina transmissora circular. Neste caso, o potencial de Schelkunoff, entendido como um dipolo vertical é dado por

$$F(r,z) = \frac{\hat{z}_0 m}{4\pi} \int_0^\infty \left[e^{-u_0|z+h|} + R_{TE} e^{u_0(z+h)} \right] \frac{\lambda}{u_0} J_0(\lambda r) d\lambda, \tag{1.64}$$

sendo R_{TE} o coeficiente de reflexão, J_0 a função de Bessel de ordem zero e m o momento magnético do dipolo. A variável λ é definida como:

$$\lambda = \sqrt{k_x^2 + k_y^2}, \tag{1.65}$$

sendo K_x e k_y as frequências nas direções x e y. O outro termo u_n pode ser indicado como:

$$u_n = \sqrt{\lambda^2 + k^n}, \tag{1.66}$$

onde k_n^2 é o número de onda da n-ésima e r é a distância radial da fonte ao receptor, indicado como:

$$r = \sqrt{x^2 + y^2}. \tag{1.67}$$

Integrando a equação 1.64 em uma bobina circular de raio a e corrente I, obtém-se a expressão:

$$F(r,z) = \frac{\hat{z}_0 I a}{2} \int_0^\infty \left[e^{-u_0|z+h|} + R_{TE} e^{u_0(z+h)} \right] J_1(\lambda a) J_0(\lambda r) d\lambda, \tag{1.68}$$

em que J_1 é a função de Bessel de primiera ordem. Para obter a expressão do campo magnético vertical na situação do *loop* central se subtitui a equação 1.68 na equação 1.57 e se obtém a integral de Hankel no domínio da frequência:

$$H_z = \frac{Ia}{2}\int_0^{\infty}\left[e^{-u_0|z+h|} + R_{TE}e^{u_0(z+h)}\right]\frac{\lambda^2}{u_0}J_1(\lambda a)d\lambda. \tag{1.69}$$

Para obter a resposta transiente no domínio do tempo, faz-se a Transformada inversa de Laplace o Fourier. Esta integral não pode ser ressolvida analiticamente, por tal motivo R_{TE} se torna:

$$R_{TE} = \frac{\lambda - u}{\lambda + u}, \tag{1.70}$$

assim a equação 1.69 se reduz para:

$$H_z = \frac{Ia}{2}\int_0^{\infty}\frac{\lambda^2}{\lambda - u}J_1(\lambda a)d\lambda. \tag{1.71}$$

A equação acima pode ser ressolvida para o campo $\mathbf{B}$, usando a expressão $\mathbf{B} = \mu_0\mathbf{H}$ e aplicando a transformada inversa de laplace, tem-se:

$$B_z = \frac{\mu_0 I}{2a}\left[\frac{3}{\pi^{1/2}\theta a}e^{-\theta^2a^2} + (1 - \frac{3}{2\theta^2a^2})erf(\theta a)\right], \tag{1.72}$$

sendo $\theta = \sqrt{\frac{\mu_0\sigma}{4t}}$ e $erf(\theta a)$ a função erro. Para calcular a resposta do impulso, derivada-se o campo magnético em relação ao tempo, tem-se:

$$\frac{\partial B_z}{\partial t} = -\frac{I}{\sigma a^3}\left[3erf(\theta a) - \frac{2}{\pi^{1/2}}\theta a(3 + 2\theta^2a^2)e^{-\theta^2a^2}\right]. \tag{1.73}$$

A equação acima, representa a resposta do impulso, que é a medição feita no campo. No entanto, as medições do campo secundário só são possíveis usando um intervalo de de tempo pequeno, uma vez a corrente é desligada. Este curto período de tempo é chamado de tempo tardio. Para estes tempos, o impulso pode ser aproximado por:

$$\frac{\partial B_z}{\partial t} = \left(\frac{I\sigma^{3/2}\mu_0^{5/2}a^2}{20\pi^{1/2}}\right)t^{-5/2}. \tag{1.74}$$

A equação acima apresenta uma derivada temporal de B proporcional a $t^{-5/2}$. No entanto, a equação mais representativa para medir os tempos tardios no método TDEM é dada por:

$$\rho_a = \left(\frac{Ia^2}{20\partial\frac{B_z}{\partial t}^{1/2}}\right)\frac{\mu_0^{5/3}}{\pi^{1/3}}t^{-5/3}. \tag{1.75}$$

Inversão 1.5D de dados TDEM

A inversão de dados eletromagnéticos no domínio do tempo (TDEM) tem sido usada nos ultimos 30 anos para o mapeamento de resistividade na subsuperfície. Seu uso tem ficado bastante restrito para problemas de inversão 2D e 3D, devido ao seu alto custo computacional. Neste trabalho, usou-se um esquema de inversão de quase 2D, no qual são usados conjuntos de sondagens eletromagnéticas 1D fazendo uma transversa. A figura 1.5 mostra a configuração de sondagens paralelas equiespaçadas conformando um meio de camadas plano-paralelas.

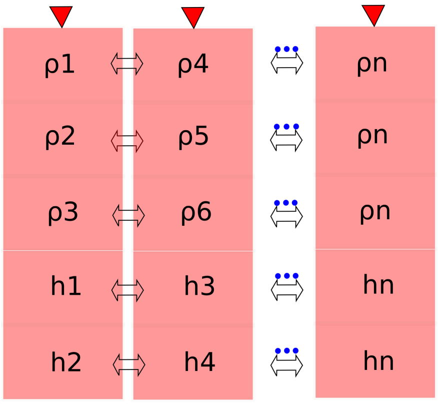

Figura 1.5: Configuração de sondagens eletromagnpeticas 1D (triângulos vermelhos), dispostos ao longo de uma transversa.

Neste trabalho foi implementada a estratégia de inversão lateralmente vinculada, (e.g., Auken e Christiansen (2004) and Santos (2004)), mas seguindo modelagens diretas 1D na forma de modelos de camadas, tal como Auken et al. (2005). No entanto, estas metodologias de inversão foram desenvolvidas usando métodos locais aplicados só a problemas de resistividade quase 2D. Neste trabalho, usaram-se novos métodos de otimização global (MOGs) para caraterizar a resistividade na subsuperfície, aplicando sondagens eletromagnéticas no domínio do tempo (TDEM). Além disso, foi usado o vínculo de suavidade global (SG), aplicado no trabalho de Constable et al. (1987) e o vínculo de variação total (VT), definido em Rudin et al. (1992) para estabilizar o problema inverso de natureza instável. Finalmente, foram implementadas versões paralelizadas dos métodos de otimização global usando o paradigma de programação MPI para acelerar o tempo de computação.

1.1.4 Contextualização sobre a computação em paralelo

Nesta sub-seção é descrita a teoria principal sobre paralelismo, que inclui as técnicas de processamento paralelo por memória compartilhada *Open Multi-Processing* (OpenMP) e troca de mensagens *Message Passage Interface* (MPI).

Computação por memória compartilhada OpenMP

A interface OpenMP permite desenvolver programas em arquiteturas de multiples processos em C/C++ e Fortran. Seu fácil e flexível uso, possibilita desenvolver aplicações paralelas basadas em memória compartilhada. Todos os processos OpenMP começam com um único processo chamado de $master{-}thread$, dentro de uma região paralela encontrada. O $master-thread$ cria um conjunto de $worker - threads$ que executam instruções em paralelo, onde finalmente os *threads* sincronizam e juntam todas as instruções novamente no $master - thread$.

As principal diretiva usada no OpenMP inclui a notação `!$omp ...` que atua sobre um bloco do código e constrói um conjunto de *threads* que executam este bloco em paralelo

```
!$omp parallel [clause ...]
Block ...
IF
PRIVATE
SHARED
REDUCTION
NUM_THREADS
!$omp end parallel
```

Algumas das diretivas mais importantes no interior do bloco acima apresentado podem ser descritas como:

- `!$omp parallel IF` Executa a operação se a condição é verdadeira.
- `!$omp parallel PRIVATE (lista)` Duplica as variáveis em cada *thread*

- `!$omp parallel SHARED (lista)` As variáveis definidas na lista são distribuídas em todos os *threads*, ficando a critério do programador seu acesso.
- `!$omp parallel REDUCTION (lista)` Diretiva que define as variáveis usadas em operações de redução. Estas variáveis são duplicadas em cada *thread* e seu acesso é privado.
- `!$omp parallel NUM_THREADS (expressao escalar inteira)` Realiza em paralelo junto com o número de *threads* uma expressão determinada.

Finalmente a diretiva `!$OMP DO` especifica que as operações do *LOOP* deven ser excutadas em paralelo pelo conjunto de *threads*. Sua notação pode ser escrita como segue:

```
!$omp do [clause ...]
.
.
.
Bloco de operações
.
.
.
!$omp end do
```

Computação por memória distribuída MPI

Na década de 90, um grupo de cientistas e membros da indústria de computadores desenvolveu uma interface de passagem e recepção de mensagens. Esta tarefa, foi implementada em diferentes linguagens de programação e diferentes arquiteturas computacionais. No caso particular da interface MPI, um mesmo programa em paralelo executa uma computação sob um subconjunto de uma estrutura de dados distribuída. Cada processo idenficado como *rank*, varia segundo um tamanho *size* ou número de processos disponíveis. A estrutura geral MPI em Fortran usada nesta pesquisa é mostrada como segue:

```
program
use mpi
implicit none
integer :: size, rank, ierr
```

```
call MPI_Init(ierr)
call MPI_Comm_size(MPI_COMM_WORLD, size, ierr)
call MPI_Comm_rank(MPI_COMM_WORLD, rank, ierr)

Bloque para troca de mensagens e execucao dos preocessos

call MPI_Finalize(ierr)
end program
```

As comunicações via MPI podem ser feitas mediente dois mecanismos, tais como comunicação ponto a ponto e comunicação coletiva.

A comunicação ponto a ponto, faz uso das diretivas `MPI_Recv` e `MPI_Send`, onde a passagem de mensagens ocorre entre dois e só dois processos MPI diferentes. Um processo faz uma operação de enviar e outro a de receber.

A comunicação coletiva envolve operações onde todos os processos de um grupo participam, com o objetivo de manipular uma parte comum da informação por todos os processos do grupo. Algumas das principais operações coletivas do paradigma MPI incluem:

- `MPI_Barrier`: Esta diretiva permite criar uma barreira de sincronização do grupo. Uma vez a diretiva é invocado, cada processo é bloquedado até que todos os processos atinjam a mesma chamada MPI-Barrier.

- `MPI_Bcast`: Diretiva para trasmitir uma mensagem desde um processo para os outros processos no grupo.

- `MPI_Scatter`: Diretiva que distribui mensagens diferentes desde uma único processo fonte para cada processo no grupo.

- `MPI_Gather`: Diretiva que reúne mensagens diferentes de cada processo no grupo com destino um único processo. Esta directiva é oposta a `MPI_Scatter`.

- `MPI_Allgather`: Diretiva que reúne mensagens diferentes de cada processo no grupo com destino todos os processos.

- `MPI_Allreduce`: Diretiva que usa uma operação de redução e locação de algum resultado em todos os processos do grupo. Esta operação inclui a diretiva `MPI_Reduce` e `MPI_Bcast`.

- `MPI_Reduce_scatter`: Esta diretiva reduz elemento a elemento em um vetor entre todos os processos no grupo. Em seguida esta operação, o vetor resultante é dividido em segmentos e distribuído entre os processos. Envolve a diretiva `MPI_Reduce` e logo a diretiva `MPI_Scatter`.
- `MPI_Alltoall`: Esta diretiva distribui diferentes dados para cada um dos processos receptores.
- `MPI_Scan`: Esta diretiva reduz os dados distribuidos em todo o grupo.

1.1.5 Resultados

Os resultados obtidos neste trabalho de doutorado são apresentados através de dois artigos. No primeiro artigo (capítulo 2), intitulado *A parallel improved PSO algorithm with genetic operators for 2D inversion of resistivity data* (Um algortimo paralelo melhorado com operadores genéticos para inversão 2D de dados de resistvidade), propõe-se um novo algoritmo de inversão de resitividade 2D usando o Particle Swarm Optimization e operadores genéticos de elitismo-mutação para melhorar a diversidade da população. Além disso, o algoritmo usa computação em paralelo para acelerar o tempo da inversão associado com o uso de métodos de otimização global. As conclusões do primeiro artigo se descrevem a continuação são:

- A estratégia de elitismo-mutação permitiu melhorar a exploração e diversidade do algoritmo.
- Os vínculos de Suavidade Global (SG) e Variação Total (VT) mostraram bons resultados dos modelos invertidos para dados sintéticos e reais, sendo o vínculo de VT que apresentou melhores taxas de variação da função custo em relação ao vínculo de SG
- O paradigma de programação MPI puro apresentou acelerações de 17 vezes em relação ao tempo de processamento sequencial. Além disso, o paradigma híbrido OpenMP-MPI foi ligeramente inferior ao tempo de inversão usado com o MPI puro.

O segundo artigo (capítulo 3) foi intitulado *1,5D time domain electromagnetic inversion using global optimization methods and parallel processing* (Inversão 1.5D de dados eletromagnéticos no domínio do tempo usando metodos de otimização global e processamento em paralelo). Aborda um esquema de inversão quase 2D usando vínculos tais como Suavidade Global (SG) e Variação Total (VT), através do da configuração de sondagens eletromagnéticas no domínio do tempo 1D dispostas em transversas. A inversão é conduzida usando dois novos métodos de otimização global tais como o algoritmo Whale Optimization Algorithm

(WOA) e o Grey Wolf Optimizer (GWO). Além disso, é usada a computação em paralelo para acelerar o tempo da inversão. Os resultados deste artigo são:

- A inversão 1.5D TDEM obteve uma boa recuperação dos modelos da subsuperfície usando os vínculos de SG e VT. No entanto, o último mostrou-se mais robusto na identificação de estruturas com altos constrastes de resistividade em relação ao SG.
- O algoritmo WOA e PSO apresentaram melhores ajustes das taxas de variação da função custo e taxas de convergência, necessitando menor quantidade de modelagens diretas do que o algortimo GWO.
- O uso de técnicas de paralelismo MPI permitu acelerar 17 vezes o tempo da inversão em relação ao tempo de processamento sequêncial para os dados sintéticos e reais.

2

Artigo 1

Este manuscrito publicado na revista Acta Geophysica de Qualis B1, intitulado: "*A parallel improved PSO algorithm with genetic operators for 2D inversion of resistivity data*".

Nas seguintes páginas encontra-se o manuscrito desenvolvido nesta pesquisa.

A parallel improved PSO algorithm with genetic operators for 2D inversion of resistivity data.

Jorge L. Abril[1*], Marcos A. Vasconcelos[1†], Francisco M. Barboza[2†] and Oscar F. Mojica[3†]

[1*]Center for Research in Geophysics and Geology (CPGG), Federal University of Bahia (UFBA), Rua Barão de Jeremoabo-Ondina, Salvador, 100190, Bahia, Brazil.
[2]Computing and Technology Department (DCT), Federal University of do Rio G do Norte, Street, Caico, 10587, Rio Grande do Norte, Brazil.
[3]Supercomputing Center for Industrial Innovation, SENAI CIMATEC, Av. Orlando Gomes, Salvador, 610101, Bahia, Brazil.

*Corresponding author(s). E-mail(s): jlab66@gmail.com;
Contributing authors: marcos.vasconcelos@ufba.br; marcio.barboza@ufrn.br; o_mojical@hotmail.com;
†These authors contributed equally to this work.

Abstract

In this paper an improved particle swarm optimization technique known as elitist-mutated particle swarm optimization (EMPSO) was applied in the 2D electrical resistivity imaging; a complex and highly non-linear optimization problem. The EMPSO enables better exploration of the search space, by replacing particles with a worse performance by the best particle of the swarm mutated in random positions. Nevertheless, this technique, as any other based on a population of models, costs much computation time in solving optimization problems with a large number of unknown parameters. We addressed this problem by developing a parallel version of the EMPSO that supports pure MPI and hybrid MPI-OpenMP modes, and we named as Parallel elitist-mutated PSO (PEMPSO). The solution to the inverse problem is based on minimizing an objective function with a regularization term to create a mathematically stable solution. Total variation and global smoothness regularizations were used in the inversion of synthetic data obtained from simple models and a set of real data of a highly complex geological/geotechnical nature. By virtue of the features of the synthetic models and the geology of the local where the data were acquired, the inversions with Total variation regularization provided the best outcomes. Additionally, we have improved the execution time significantly with our parallel solution (the pure MPI model turned out to be better than the hybrid model) in comparison with the sequential version. Cumulative frequency distribution of errors between modeled and observed apparent resistivity data for all experiments was used to validate the PEMPSO technique for estimating resistivity.

Keywords: constrained inversion, electrical resistivity imaging, particle swarm optimization, parallel computing

1 Introduction

Direct current (DC) resistivity imaging techniques have been used for more than 80 years in mapping of the distribution of resistivity in the subsurface. The widespread adoption of such tecnhiques in several environmental, engineering and hydrological studies is mainly due to their low cost and simplicity of data collection. Once the data are acquired, a processing step come before they are used in the inversion process. Then the task is to interpret the results. Existing approaches for solving the inverse problem use local gradient-based or global search methods, both of which can involve the use of constraints to stabilize its ill-possed nature (Chambers et al, 2006; Loke et al, 2013). In the first case, starting from a reasonable point in the model space, the methods based on the gradient allow a quick descend to the nearest minimum. The disadvantage of this method is that it depends on a good starting point (Gill et al, 2019). In the second case, the model space is searched by trajectories chosen randomly until a satisfactory minimum is reached. In this scheme, entrapment around local minima is avoided by owned strategies of the global search methods, also called global optimization methods (GOMs). Because of their way to explore the model space, GOMs are quite time-consuming compared to gradient-based search methods, but on the other hand, they have no restrictions in relation to the starting model. Methods as simulated annealing (SA)(Kirkpatrick et al, 1983) and the variants of its base version, fast simulated annealing (FSA)(Szu and Hartley, 1987) and very fast simulated annealing (VFSA)(Ingber, 1989) along with the genetic algorithm (GA)(Holland, 1975) are included into this group.

Despite their high computational costs, GOMs have been used successfully in the inversion of DC resistivity data. Chunduru et al (1995) and Jha et al (2008) used GA in the inversion of vertical electrical sounding (VES) data, while Sen et al (1993) and Sharma (2012) used SA and VFSA for the same purpose, Ferreira et al (2003) developed and hybrid scheme which combines GA and a local gradient-based search method to improve the convergence and the quality of the solution in a 2D inversion of resistivity data, Schwarzbach et al (2005) and Başokur and Akca (2011) also applied GA for 2D DC resistivity inversion and Liu et al (2012) introduced an improved GA to solve the 3D inversion problem. These many examples show that SA (or a variant) and GA are by far the methods most used. This is the case not only for resistivity data, but other geophysical data as well (Pace et al, 2021).

Apart from these traditional methods, the Particle Swarm Optimization (PSO) is becoming increasingly popular as an alternative among geophysicists to solve ill-posed inversion problems. PSO algorithm is based on the behavior of animals in swarm to achieve a common goal with a collaborative manner. It has been brought into applied geophysics by Shaw and Srivastava (2007), and since then has been widely adopted in the inversion of several geophysical data. However, works with PSO involving the inversion of resistivity data are practically limited to the 1D case (Pace et al, 2021). As examples where PSO has been used to solve the 2D resistivity inversion problem, we can cite the studies by Barboza et al (2019) and Arboleda-Zapata et al (2022). In the former case, the study focused on the incorporation of different constraints on the functional to be minimized. The authors emphasize that their approach allows the interpreter to assign various restrictions in different parts of the grid used in the resistivity discretization in the subsurface. Nevertheless, their approach is associated with the problem of high computational cost. In the latter case, the study was concentrated on developing a workflow that analyzes an ensemble of models generated from a PSO-based inversion approach to assess uncertainties related to the non-uniqueness of the inverse problem.

This paper proposes to use an improved version of the PSO, which is called EMPSO, (Nagesh Kumar and Janga Reddy, 2007) for 2D inversion of DC resistivity data. EMPSO improves the weaknesses in the PSO by using GA features as elitism and mutation. Besides its use, we introduce a parallel version of it to overcome the natural problem of the computing cost associated with the high number of evaluations of the objective function (Sen and Stoffa, 2013). We implemented the PEMPSO algorithm with 1) an MPI model and 2) a hybrid MPI-OpenMP model build on the basis of the MPI implementation. Through the parallel MPI implementation, assessments of the objective function are performed simultaneously using multiple MPI processes. In the hybrid implementation, computationally intensive parts of the code (.e.g., update of particle positions and velocities) are executed in parallel by using OpenMP work-sharing directives. The potential of the PEMPSO algorithm was inspected by means of tests with synthetic data obtained from known resistivity distributions and real data from a field survey carried out in the city of

Salvador-Bahia, Brazil. In both cases, the PEMPSO was found to be efficient in obtaining a resistivity model of the subsurface.

2 Mathematical approach of the 2D resistivity inversion

The parameterization of the medium consists of its partition in rectangular and homogeneous cells, considering the resistivity of each cell a model parameter. The subsurface resistivity $\rho(x, z)$ distribution, can then be arranged in an $N_x N_z$-size vector $\mathbf{m}$, where N_x and N_z represent the number of cells along $x-$ and $z-$ directions of the cell based 2D model, respectively. A forward modeling algorithm based on the iterative solution of the electric potential equation through the finite-difference technique (Dey and Morrison, 1979; Medeiros, 1987) allow us to estimate the apparent resistivity data $\mathbf{d}^{cal}(\mathbf{m})$. This vector might reproduce roughly the set of observed data $\mathbf{d}^{obs}$, that is:

$$\mathbf{d}^{obs} = \mathbf{d}^{cal} + \eta \tag{1}$$

in which η it is a vector associated with discrepancies. Minimizing of the norm of η to recover the model parameters $\mathbf{m}$, is an ill-posed problem with nonunique solution. In order to obtain stable solution for this ill-posed problem, it is necessary to restrict the solutions to a particular class of models which may meet additional constraints. In our study, we consider a minimization of two-term objective function as follows (Barboza et al, 2019):

$$\mathcal{L}[\mathbf{d}^{cal}(\mathbf{m}), \mathbf{d}^{obs}(\mathbf{m});\ \mathbf{m}] = \Phi_a[\mathbf{d}^{cal}, \mathbf{d}^{obs}] + \lambda\mathcal{R}(\mathbf{m}) \tag{2}$$

In the equation above, the first term Φ_a of $\mathcal{L}$, requires an adjustment between predicted $\mathbf{d}^{cal}$ and observed $\mathbf{d}^{obs}$ data, in the least-squares sense, that is

$$\Phi_a = \frac{||log(\mathbf{d}^{cal}(\mathbf{m})) - log(\mathbf{d}^{obs}(\mathbf{m}))||_2^2}{N_{obs}}, \tag{3}$$

where N_{obs} is the total number of measurements and $||\cdot||_2^2$ denotes the squared $l2$-norm. On the other hand, the second term of equation (2), is composed of λ and $\mathcal{R}(\mathbf{m})$. These terms correspond to the regularization parameter and the constraint used to stabilize the solution of the inverse problem, respectively. We have used two constraints for simulations with synthetic and field data: (i) the global smoothness (GS) (deGroot Hedlin and Constable, 1990) which makes the inversion process converge to a solution without abrupt variations between the model parameter values and; (ii) the total variation (TV) constraint, which has the opposite effect otherwise and leads to discontinued solutions (Farquharson and Oldenburg, 1998; Bertete-Aguirre et al, 2002). The following section will describe the mathematical expressions for both constraints.

3 Mathematical definition of the constraints

Different classes of stabilizing constraints can be used in the 2D resistivity problem (Barboza et al, 2019). Such constraints can be used separately or in combination. In the first case, a single constraint is applied in the whole inversion mesh to obtain the solution. In the latter, different constraints can be applied to different parts of the mesh, following the interpreter's choice. Here, we chose the first option, that is, the constrainst were independently applied in the inversion examples, using both synthetic and field data.

3.1 Global Smoothness Constraint (GS)

This constraint, also called the Spatial Continuity Constraint (Constable et al, 1987; Tikhonov and Arsenin, 1977), imposes a smoothing in the variation of the physical property. It favors solutions in which the discrepancy between the resistivity values of neighboring inversion cells in both directions of the grid is minimal. In other words, the constraint will choose low contrasts in physical property. The mathematical expression is given by:

$$\mathcal{R}_{GS} = ||\mathbf{Dm}||_2^2 \tag{4}$$

where $\mathbf{D}$ represents the matrix of discrete first derivatives along the horizontal and vertical directions, $\mathbf{m}$ represents the vector of the model parameters and $|| \cdot ||_2^2$ denotes the squared $l2$-norm.

3.2 Total Variation Constraint (TV)

This constraint can be used to mapping high contrasts (discontinuities) in the distribution of the model parameters, as demonstrated by Rudin et al (1992) and Portniaguine and Zhdanov (1999). It is defined by equation:

$$\mathcal{R}_{TV} = ||\mathbf{Dm}||_1, \tag{5}$$

where, as in the previous case $\mathbf{D}$ and $\mathbf{m}$ represent the matrix of discrete first derivatives along the horizontal and vertical directions and the vector of the model parameters, respectively, and the term $|| \cdot ||_1$ denotes the $l1$-norm.

4 Inversion of resistivity data using the improved PSO

In the past two decades, PSO has been helpful in solving optimization problems in different branches of science, including geophysics (e.g., Chen et al 2017; Pace et al 2019; Moura et al 2020). This technique, developed by Kennedy and Eberhart (1995), is based on the group behavior of several animals, who work together seeking a shared goal (e.g. to find food). PSO algorithm emulates the interaction between individuals in a group to share information. In pursuit of the shared goal, each individual in the group use its experience and the group experience as well or, stated another way, the individual decision making process is not based solely on its experience, but also in the experience of others individuals (Kennedy, 2006).

In the PSO, the individuals are called particles and represent the potential solutions. They move in the space of the problem $\mathbb{R}^n$, where n is the size of space. A particle i is defined by its position vector, $\mathbf{m}_i = m_{i1}, m_{i2}, \ldots, m_{in}$, and its velocity vector, $\mathbf{v}_i = v_{i1}, v_{i2}, \ldots, v_{in}$. In the 2D resistivity problem, then, the position vector represents a resistivity model, that is, a vector whose elements are the resistivity values of the n cells of the 2D mesh. The performance of each particle for our specific problem is assessed using equation (2), which reflects the quality of the particle's position. The best location attained by the i-th particle (known as the personal best) is represented by $\mathbf{p}_i = p_{i1}, p_{i2}, \ldots, p_{in}$, while the best location achieved by neighboring particles (known as the local best) is represented by $\mathbf{g}_i = g_{i1}, g_{i2}, \ldots, g_{in}$. At each iteration k, each particle changes its position according to the new velocity as in

$$\mathbf{v_i}^{k+1} = w^k \mathbf{v_i}^k + c_1 \otimes \mathbf{u}_1(\mathbf{p}_i - \mathbf{m}_i{}^k) + c_2 \otimes \mathbf{u}_2(\mathbf{g}_i - \mathbf{m}_i{}^k) \tag{6}$$

$$\mathbf{m_i}^{k+1} = \mathbf{m_i}^{k} + \mathbf{v_i}^{k+1} \tag{7}$$

where $i = 1, 2, \ldots, n_{part}$, $\mathbf{p}_i$ and $\mathbf{g}_i$ denote the best particle position and the best neighborhood position and the parameters w, $c1$, $c2$, $\mathbf{u}_1$ and $\mathbf{u}_2$ are the inertia factor, two positive constants and two vectors of random numbers within [0, 1], respectively ($\otimes$ denotes element-by-element vector multiplication). Equations (6) and (7) constitute the classic PSO, whose flow chart is shown in Figure 1.

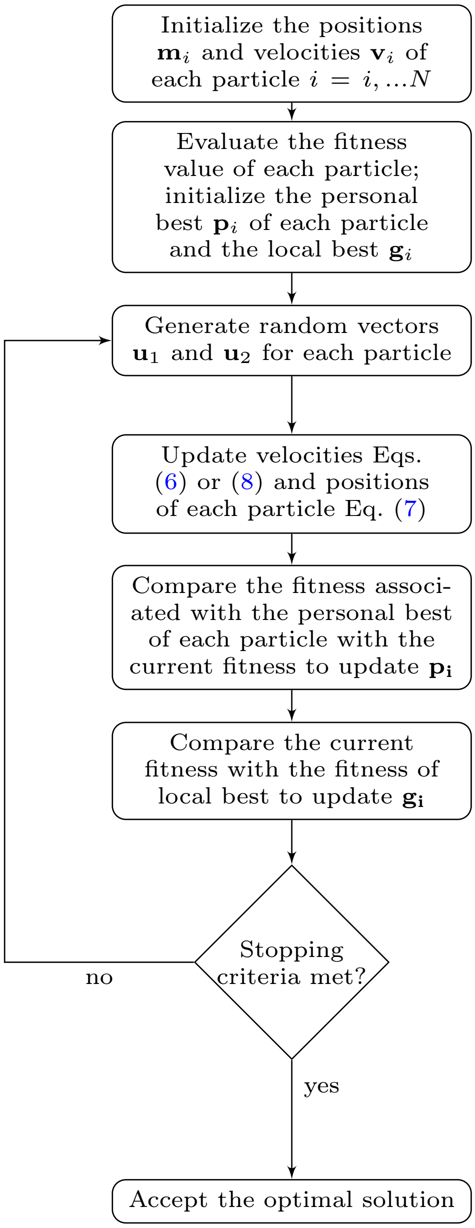
Initialize the positions $\mathbf{m}_i$ and velocities $\mathbf{v}_i$ of each particle $i = i, \ldots N$
Evaluate the fitness value of each particle; initialize the personal best $\mathbf{p}_i$ of each particle and the local best $\mathbf{g}_i$
Generate random vectors $\mathbf{u}_1$ and $\mathbf{u}_2$ for each particle
Update velocities Eqs. (6) or (8) and positions of each particle Eq. (7)
Compare the fitness associated with the personal best of each particle with the current fitness to update $\mathbf{p_i}$
Compare the current fitness with the fitness of local best to update $\mathbf{g_i}$
Stopping criteria met?
no
yes
Accept the optimal solution

Figure 1: The flowchart of the canonical PSO algorithm

Another version was proposed by Clerc (1999), who modified the equation to calculate the new velocity including a constriction factor χ, in order to improve the convergence of the PSO

$$\mathbf{v_i}^{k+1} = \chi[\mathbf{v_i}^k + c_1 \otimes \mathbf{u}_1(\mathbf{p}_i - \mathbf{m}_i{}^k) + c_2 \otimes \mathbf{u}_2(\mathbf{g}_i - \mathbf{m}_i{}^k)]. \tag{8}$$

The constriction factor is given by:

$$\chi = \frac{2\kappa}{2 - \varphi - \sqrt{(\varphi^2 - 4\varphi)}} \tag{9}$$

where $\kappa \in [0, 1]$ and $\varphi = c1 + c2 \geq 4$, with typical values of $\kappa = 1$, $c1 = c2 = 2.05$ and $\varphi = 4.1$. The inclusion of the factor of inertia (w), as well as the constriction coefficient (χ) of equations (6) and (8) helps prevent the explosion of the swarm, which means an uncontrollable increase in the magnitude of the velocity (displacement of the particles). Moreover, Eberhart and Shi (2000) suggested to limit the velocity of the particles between a maximum and a minimum range $[-\mathbf{v}_{max}, \mathbf{v}_{max}]$ with $\mathbf{v}_{max} = \mathbf{x}_{max}$ and $\mathbf{x}_{max}$ as the limits of the search space, i.e. $\mathbf{x}_{upper} - \mathbf{x}_{lower}$. If the constraint is violated, the d-th component of the velocity for the i-th particle it is adjusted to the nearest velocity limit, like this:

$$v_{id} = {v_d}^{max}, \text{ if } \; v_{id} > {v_d}^{max}, \tag{10}$$

$$v_{id} = -{v_d}^{max}, \text{ if } \; v_{id} < -{v_d}^{max}. \tag{11}$$

Since its proposal in the mid-nineties, the PSO has been the subject of extensive research and several advances and/or variants have already been proposed to help it become more robust. In this study, we have adopted a variant that imports characteristics of GA in the PSO, called EMPSO (Nagesh Kumar and Janga Reddy, 2007). The EMPSO adds an elitist-mutation (EM) exploration strategy which proceeds as follows: first, all particles are sorted in ascending order based on their objective function and the index numbers for the respective particles are obtained, that is, the index of the classified swarm is recorded in $ASF[i], i = 1, \ldots, n_{part}$; then, the elitist mutation is performed on a predetermined number NM of the worst particles (selected according to the classification) and the respective particle positions vectors are replaced by the new mutated position vectors. These latter are obtained after the execution of the mutation in some components of the global best position vector, which are chosen according to the preestablished probability of mutation P_{em}. Nonetheless, the velocity vectors of the chosen NM particles remain the same. Figure 2 shows the pseudo-code implementation of the EM scheme.

```
for $i \leftarrow 1$ to $NM$ do
    $l \leftarrow ASF[i]$
    for $d \leftarrow 1$ to $n$ do
        if rand $< p_{em}$ then
            $x_{ld} = g_d + 0.1 \times VR_d \times$ randn
        else
            $x_{ld} = g_d$
        end if
    end for
end for
```

Figure 2: Pseudo-code of EM operator (Nagesh Kumar and Janga Reddy, 2007), where NM is the number of particles to be elitist-mutated; p_{em} is the probability of mutation; g_d is the d-th component of global best particle; ASF is the index of sorted population; `randn` is a Gaussian random number $N(0, 1)$; and VR_d is the range of decision variable d.

In addition to the elitist-mutation strategy, we included parallel processing to increase the efficiency of the PSO algorithm, since GOMs are very time consuming due to the large number of iterations necessary for their convergence. A situation that can get even worse in problems involving a large number of unknown parameters, as this leads directly to the increase of the computational cost in the evaluation of the objective function. Thanks to its inherent parallelism, and to the astonishing recent growth of computational power, many studies have been carried out on the parallelization of the PSO algorithm (Lalwani et al, 2019). Our parallel code uses data parallelism with division of tasks on distributed systems, including multi-core processors. The following section provides the details of the parallel EMPSO implementation.

5 Parallel implementation of the PSO

Given that the GOMs require a high number of evaluations of the objective function, it is clear that serial versions of the PSO would require a high computational cost. To reduce the processing time, we paid particular attention to the code implementation and incorporated the use of the MPI library to expand computing across multiple processes. In addition, a hybrid MPI + OpenMP version of the code was implemented, adding OpenMP directives in time-consuming parts of the code (specific loops). The parallel code using MPI is based on the GA implementation of Metcalfe and Charbonneau (2003) for applications in astrophysics, which uses a dynamic partition of tasks, assigning tasks to MPI processes on demand.

5.1 General considerations about the parallelization

In a large majority of inversion problems in geophysics, the evaluation of the objective function is the step that requires the greatest processing time. Our case is not exempt from this situation, spending only a small part of the overall time for operations other than those designed to compute the objective function values, such as the updating of the positions and velocities of the particles, and the input and output (I/O) operations. Taking this into account, we have implemented a version of the PSO that is based on the parallel execution of the evaluation of the objective function for each particle. The parallel PSO algorithm was implemented using a master/worker programming paradigm with MPI. In particular, a PSO subroutine (pseudocode in Figure 3) that takes care of all tasks related to PSO (initialization of particle positions, application of the EM operator) is called in the main program by the master process. Once the particles are created, they are distributed among the available worker processes through a subroutine that computes the objective function in parallel (mpi-fitness), sending a particle (an array containing the parameter values) to each worker process. The subroutine then waits for responses of workers and sends new particles to each worker since the previous computation has finished (objective function evaluation). A given worker process will perform a two-dimensional forward modeling associated with the received particle and will also evaluate its corresponding objective function according to equation (2). The objective function value returns to the master process, and the worker process is then ready to receive a new particle from him. In practice, this implementation allows the master to keep all workers busy if there are enough particles to distribute. When the objective function value is assigned to the particles in each iteration, the mpi-fitness subroutine returns to the PSO subroutine that performs other PSO operations.

5.2 Code structure

Figure 3 shows, in the form of a pseudocode, the structure of PSO.

```
p(:,:),v(:,:)←initialize(n,npart,lb,ub)          ▷ Initialize all particle's position and velocity
for j = 1 to npart do
    v(:,j)←call_velocity_boundary_treatment(n,vmax)      ▷ Constrains the components of velocity
end for
pBest(:,:)←particle(:,:)                                ▷ Initialize particle's best position
while iter .le. Maxiter do                                        ▷ Main optimization loop
    particlefit(:)←call_mpi_fitness(n,npart,p)       ▷ Parallel fitness evaluation with MPI
  ▷ Updates the values of pbest and pbestfit for all the swarm using syncronous strategy
    pbest(:,:),pbestfit(:)←sync_pgbest(n,npart,p,particlefit,minmax)
  ▷ Updates the best-known position for the neighbouring particles
    localbestpos(:),localbestfit(:)←updatelocalbest(npart,pbestfit,pneigbours,minmax)
    !$omp parallel if (npart .ge. threshold) private(j)
    !$omp do
    for j = 1 to npart do
    ▷ Calculate a new particle's velocity
        v(:,j)←compute_vel(n,p(:,j),w,c1,c2,cf,pbest(:,j),pbest(:,localbestpos(j)))
    ▷ Constraints the components of the velocity
        v(:,j)←velocity_boundary_treatment(n,vmax)
    ▷ Updates the position of the particles and keeps them in the research space
        p(:,j)←position_boundary_treatment(n,v(:,j),lb,ub,boundary_wall)
    end for
    !$omp end do
    !$omp end parallel
    iter= iter +1                                                 ▷ Advance in iteration
end while
```

Figure 3: A parallel hybrid OpenMP-MPI PSO pseudo-code

The various functions and subroutines that appear implement the concepts defined in PSO description above and their operation is only outlined in the following part. The code set a population of *npart* particles in the *n*-dimensional search space, stored in the array `particle(1:n, 1:npart)`. It is initialized randomly in a specific interval defined by the arrays `lb(1:n)` and `ub(1:n)`. That way the range of parameters to be explored remains limited. The maximum possible velocity for each dimension is controlled by the `vmax (1:n)` array, which determines the maximum change that a particle can undergo during an iteration. This is done with the subroutine `velocity-boundary-treatment`. In each iteration, the objective function value of the particles is calculated using the subroutine `mpi-fitness` and stored in the *npart*-dimensional `particlefit` array. Then, each particle is updated following two "best" values. The first is the best position achieved by each particle heretofore. This value is called *pbest* and is stored in the `pbest (1:n, 1:npart)` array. The fitness value associated with `pbest` is also stored in the `pbestfit` array of the *npart* dimension. Both arrays are updated via the `syncpgbest` subroutine. Another "best" value tracked by the PSO is the position of best solution found by particle i's neighborhood (array `pneigbours`). This best value is local and is called *lbest*. The best fitness and the index of the particle with that best fitness value in the vicinity of each particle are stored in the `localbestpos` and `localbestfit` arrays of the *npart* dimension. This is accomplished by the `updatelocalbest` subroutine. When the entire swarm acts as the neighborhood of any particle, the best value tracked by the PSO becomes the best general position obtained so far by any particle of the population, which is known as *gbest*. This configuration is called topology of global neighborhood. Note that the version (without the EM operator) presented in section 4 uses the local topology, however, the code implemented supports the use of both types of topology, since the global topology is an instance of the local one. The velocities for updating the position of the particles are calculated through the `compute-vel` subroutine, which takes as input the parameters `c1`, `c2`, `w` and `cf` (constriction factor), among others. The position of the particles in the optimization space is calculated with the subroutine `position-update-and-boundary-treatment` after velocities are set to stay within the limits imposed in accordance with the boundary treatment `boundary_wall`. This concludes an iteration and the steps above are repeated again, until a predetermined number of iterations `Maxiter` is reached.

5.3 MPI-OpenMP hybrid strategy

We also coded a hybrid implementation, which is based on the code described above, but with OpenMP directives inserted in the loop to update the velocities and positions of the particles (because it is the loop that takes more time). The final program is flexible enough to run only in MPI or in hybrid MPI-OpenMP mode, depending on the form of compilation. In the pseudocode (Figure 3), it can be seen in the loop of the line 12, that both the update and the treatment of the boundary conditions of each particle are independent. This is where the directives `omp parallel if (npart <= threshold) private(j)` and `omp do` are inserted to conditionally enable parallelism i.e. if the number of particles is greater than a certain threshold. We tested both versions of PEMPSO on a high-performance cluster for academic research. Each CPU model of a cluster node is a 2.3 GHz Intel Xeon Sixteen-Core E5-2698v3 with 128 GB RAM. While several combinations of MPI processes and OpenMP threads for each node of the cluster are possible, we limit ourselves to working with two scenarios: 16 *MPI* processes × 2 OpenMP threads per node and 8 *MPI* processes × 4 OpenMP threads per node. We observed that the pure MPI programming model (which maps particles to one core) with 32 processes per node provided the best performance, reducing the computational time significantly in comparison to the serial version (see Table 1).

Model	1 Node		2 Nodes		Pure MPI	Serial
	16 MPI X 2 threads	8 MPI X 4 threads	16 MPI X 4 threads	32 MPI X 2 threads	32 MPI	
Synthetic model (Plume)	time= 1.66 hr	time= 3.13 hr	time= 1.47 hr	time= 0.74 hr	time= 1.28 hr	time= 22.5 hr
Real data (LEM 1)	time= 0.87 hr	time= 1.49 hr	time= 0.71 hr	time= 0.36 hr	time= 0.63 hr	time= 10.68 hr

Table 1: Total execution time comparison between the MPI, MPI+OpenMP and sequential inversions.

6 Application on synthetic data

In order to evaluate the PEMPSO, we have designed two synthetic models (Figures 4a and 5a), referred to as the dipping dike model and the contaminant plume model, respectively.

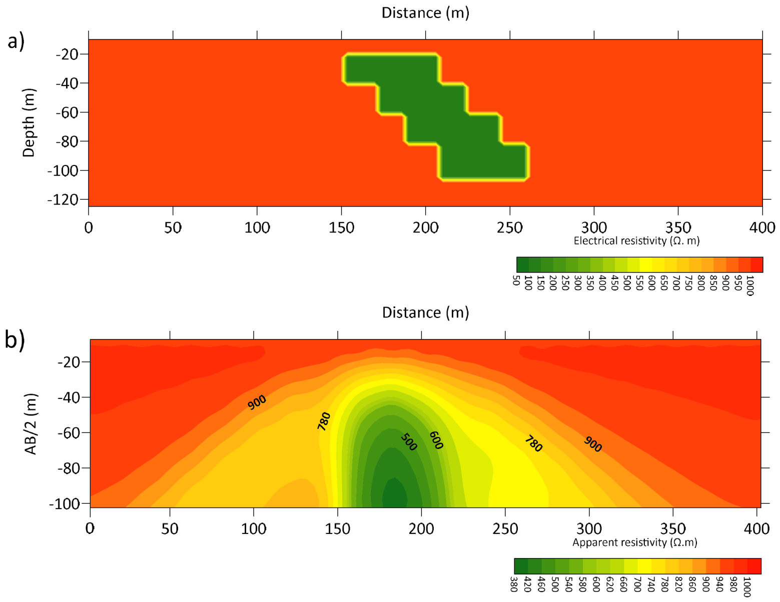

Figure 4: Synthetic data - Dike model: (a) The subsurface resistivity $\rho(x, z)$ distribution. (b) Apparent resistivity section modeled with 21 vertical electrical soundings equally spaced (20 m apart), the first and last centers being located at positions 0 and 400 m, respectively.

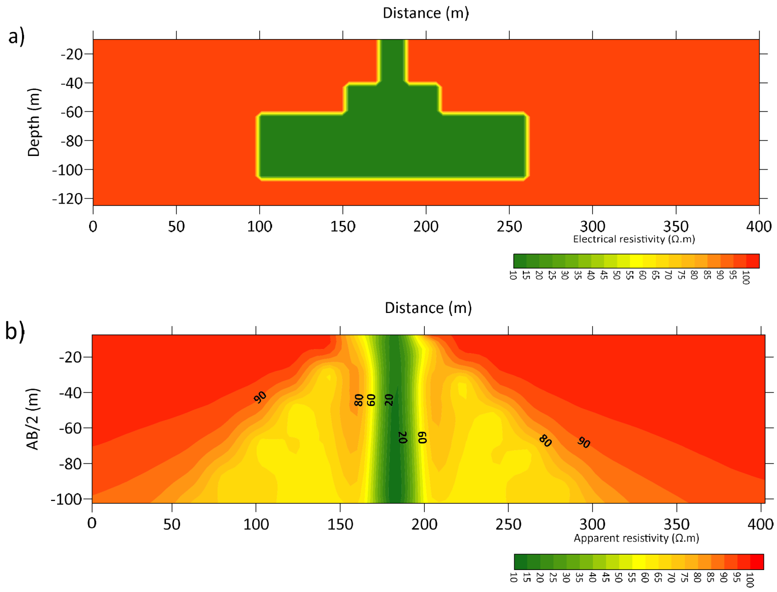

Figure 5: Synthetic data - Plume model: (a) The subsurface resistivity $\rho(x, z)$ distribution. (b) Apparent resistivity section modeled with 21 vertical electrical soundings equally spaced (20 m apart), the first and last centers being located at positions 0 and 400 m, respectively.

Both models have bodies of tabular geometry. In the plume model, a resistive half-space of 100 $\Omega.m$ involves a conductive body with resistivity of 10 $\Omega.m$ that simulates the infiltration of the contaminant underground, which spreads over the whole terrain to a depth of 100 m. The dike model is constituted of a dipping slab of resistivity 100 $\Omega.m$ in one highly resistant homogeneous half-space of 1000 $\Omega.m$. Such configuration makes the interpretation of electroresistivity data complicated. We used the finite-difference approach presented in Dey and Morrison (1979), applying the reciprocity theorem. The mesh for these models it is composed of 23 cells along the x direction and 6 cells along the z direction, thus constituting a 138 cells model. Figures 4b and 5b show pseudo-sections of apparent resistivity, for the dike model and the plume model, respectively. In each case 21 vertical electrical soundings were used, where the values of AB/2 constitute an arithmetic progression with 20 terms, where the first and the last ones are equivalent to 7.5 and 102.5 m, respectively, and the difference corresponds to the size of the mesh in x direction (5 m).

An important topic in the inversion of 2D resistivity data has to do with setting the search space for the model parameters, that is, the upper and lower limits in which the PSO random search takes place. To solve this problem, we have performed 1D inversions of the vertical electrical soundings with the Schlumberger arrangement, to establish the best range of the search space. The value of the regularization parameter (λ) in equation (2) is also an important matter in the inversion. Several authors have used different techniques to calculate this magnitude, such as Generalized cross validation-GCV (Wahba, 1990), Regisnka method (Regińska, 1996) and the most popular, the L curve criterion (Farquharson and Oldenburg, 2004). However, these approaches become computationally expensive and for this reason, the regularization parameter was adjusted using a trial and error method. Thus, after conducting several tests, it was estimated at a value of 1.d-3. Finally, to evaluate the robustness of the inversion with noisy data, we have added random Gaussian noise with zero mean and standard deviation of 2.5% to the data.

The application of the classic PSO algorithm, given by equations (7) and (8) was not effective to solve the inverse problem of 2D resistivity, once it was not able to diminish the value of the objective function in both synthetic and real data cases. Barboza et al (2019) also found convergence problems with the classic PSO and attribute them to the utilization of a generator of uniform (pseudo)random numbers for creating the initial positions of the particles. To address this issue, they used a triangular probability density function (pdf) to initialize directly the positions of the particles with log-resistivity values. In our experiments with the classic PSO, we maintained the uniform pdf and this is apparently the reason for the bad results. Hence, it is important to emphasize that all results presented below were obtained using PEMPSO.

Since the proposed PEMPSO allow us to run the inversions in an efficient manner, we performed 10 independent simulations. The average value of those simulations results is chosen as the solution. To configure the number of particles of the swarm, we have initially proved the strategies proposed by Engelbrecht (2007) for the genetic algorithm and Juan et al (2010) for the PSO. Following these authors, we tested using factors of 6, 8, 10 and 12 times, the number of the model parameters for setting the number of individuals in the swarm. However, to ensure a better exploration in the search space, we have chosen to use a constant value of 2000 particles, for the experiments with both data types. On the other hand, cognitive c_1 and social c_2 coefficients were set to 2.9 and 1.2, respectively. Other parameters of the EMPSO use the following settings: constriction factor=0.73, mutation probability $(P_m) = 0.1$ and the size of the elitist-mutated particles NM=400, i.e., 20% of the swarm population. Finally, the EM operator is applied as soon the optimization process reaches 10% of the maximum number of iterations. This is intended to preserve the diversity of the swarm in the first iterations. Below, are the results obtained for the two synthetic models used for testing the PEMPSO. In both cases, we used the maximum number of iterations (kept to be 2000) as stopping criterion.

6.1 Contaminant Plume Model

The Figure 6a, shows the convergence rate of PEMPSO for both sort of constraints used in this research (for one of the 10 simulations). From the analysis of the figure it would appear that the PEMPSO began to stabilize after about 1100 iterations, where is reached an objective function value of 0.06.

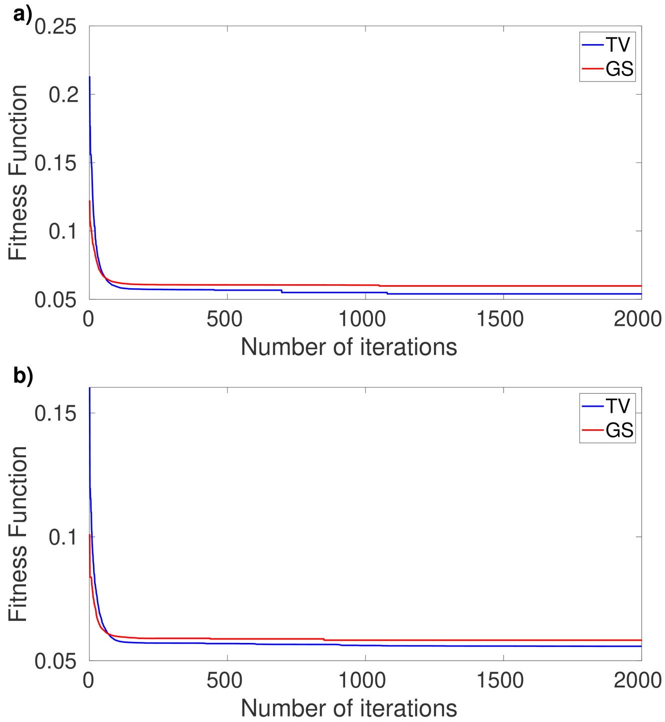

Figure 6: Evolution of the objective function in the inversion of synthetic data obtained from the (a) plume and (b) dike models.

Figures 7a and 7b show the PEMPSO inverted resistivity models using the GS and TV constraints, respectively. The results for both cases exhibited an accurate reconstruction of the conductive body.

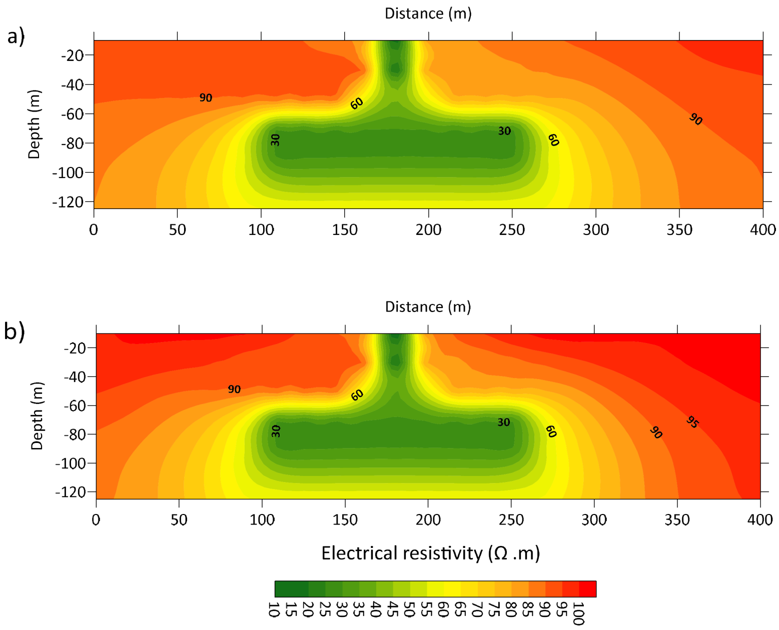

Figure 7: Synthetic data - Plume model: estimates obtained from inversion with the (a) GS constraint and (b) TV constraint.

To check the quality of both the calculated data and the best model obtained by PEMPSO (an average of the inversion results), we have performed an analysis of the cumulative frequency distribution of the relative errors, RE (%), of the apparent resistivity data and the inverted model cells. The analysis shows that almost 90 % of the data is reproduced with a value of RE=15 % (Figure 8a). It is also possible to conclude that almost 85 % of the inverted model cells reached RE values lower than 10% (Figure 8b). It is noted that the inverted model using TV regularization, not surprisingly, appears to be the best solution, due to the fact that the synthetic plume model is not really smooth.

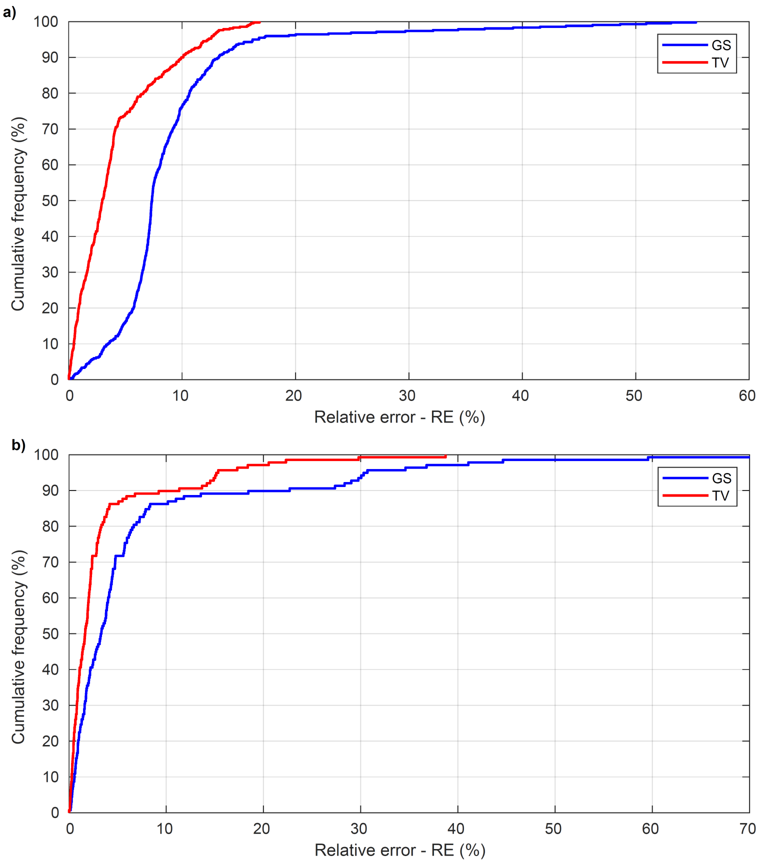

Figure 8: Synthetic data - Plume model: (a) Cumulative distribution for the percentage discrepancy between observed and modeled apparent resistivity data. (b) Cumulative distribution for the percentage discrepancy between true model and the estimated model parameter values.

6.2 Dipping Dike Model

Figures 9a and 9b show the inverted resistivity models using the GS and TV constraints, respectively.

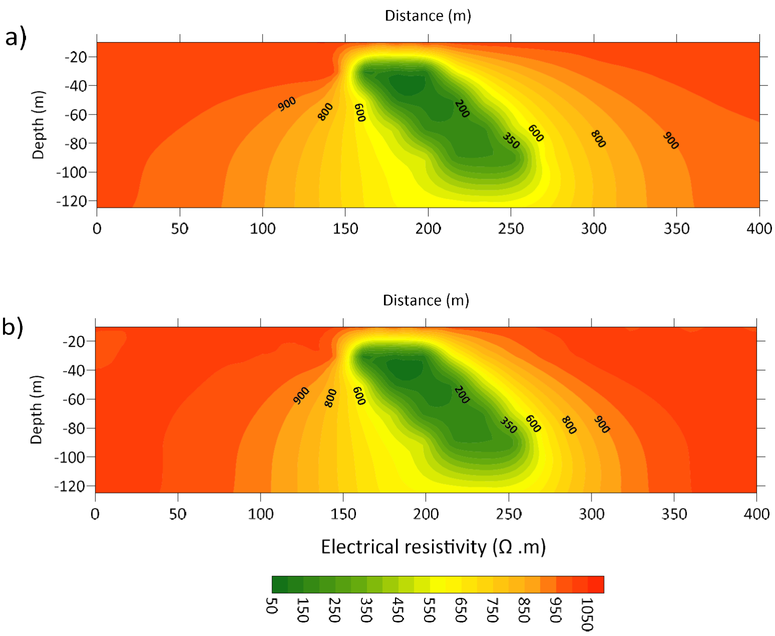

Figure 9: Synthetic data - Dike model: estimates obtained from inversion with the (a) GS constraint and (b) TV constraint.

As in the previous case, both estimates reproduce very well the true model (Figure 4a). Figure 6b, shows the convergence curves of the objective functions for the two constraints (for one of the 10 simulations). Similarly to what happens for plume model, the objective function value drops steeply in the first iterations until it achieves a plateau at 0.056, close to the 900 iterations. From the analysis of the cumulative frequency distribution of the relative errors, RE (%), of the apparent resistivity data and the average inverted model cells (Figures 10a and 10b), it is possible to establish that more than 95% of the data presented a relative error of less than 10% and nearly 90 % of the inverted model cells, reached RE values lower than 15 %. Since the dike model also has blocky features the same reasoning above can be used to explain the smaller discrepancies between true and estimated model parameter values with the TV regularization.

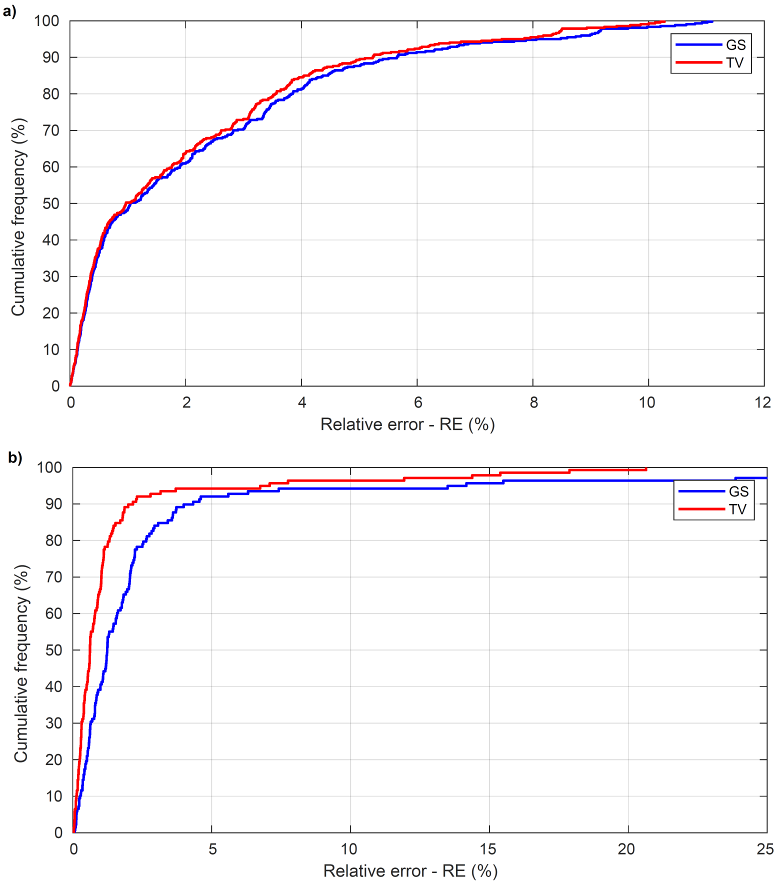

Figure 10: Synthetic data - Dike model: (a) Cumulative distribution for the percentage discrepancy between observed and modeled apparent resistivity data. (b) Cumulative distribution for the percentage discrepancy between true model and the estimated model parameter values.

7 Application on field data

The electroresistivity methods are used in investigating engineering problems as landslides, mostly due to spatial variations that can suffer the resistivity, affecting particle agglomeration, groundwater content, nature of the electrolytes, among other factors (Archie et al, 1942). Bortolozo et al (2019) employed electrical and electromagnetic methods in the geotechnical evaluation of soil stability conditions, to establish the design of foundations of an important structure in the region of Campos de Jordão, Brazil. Sigdel and Adhikari (2020), used electrical methods to evaluate the slip surface and soil conditions of a landslide in Nepal. Here, we also exploited the DC resistivity imaging technique for a landslide site characterization in the city of Salvador-BA, Brazil, with a view to assessing the usefulness of the PEMPSO in challenging scenarios.

The electroresistivity study was carried out on landslides located at the Luiz Eduardo Magalhães Avenue (LEM Avenue) that severely damaged the structure on which a road was built to improve the flow of vehicles. Many evidences in the displaced material suggests that an active fault known as Morro de Águia gave rise to the landslide.

In addition, percussion drilling operations with execution of SPTs (standard penetration tests) were carried out in the slope area of LEM Avenue, and soil samples were collected in 19 locations of the study area, aiming to analyze the loss of cohesion of the material, as well as identify the presence of the groundwater. Figure 11a shows the layout of the VES locations at the study area.

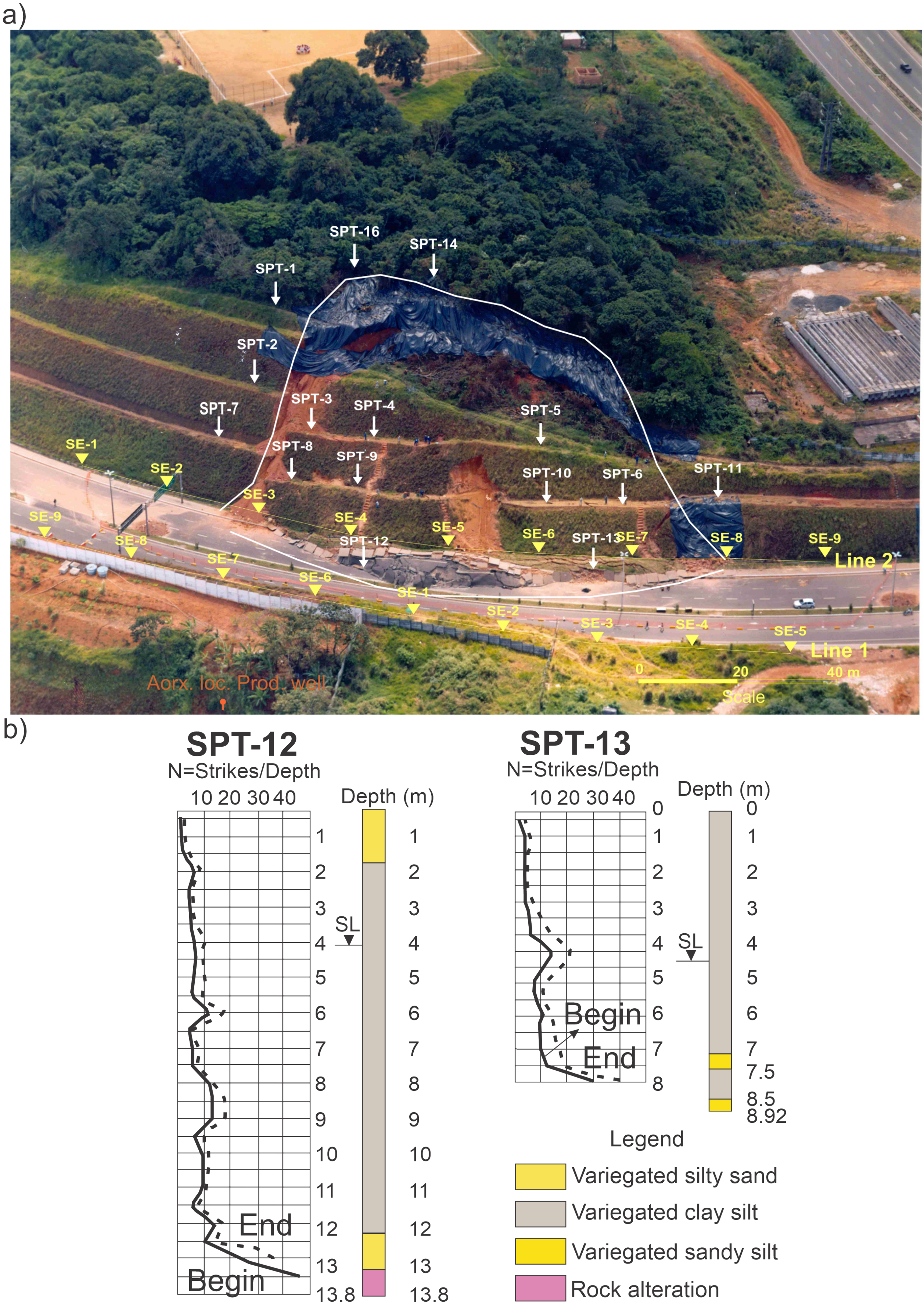

Figure 11: LEM Avenue (a) Location of study area. In yellow are the vertical electrical soundings. In white are represented the standard penetration tests surveyed in affected area. (b) Standard Penetration Tests near the Line 2. These soundings show a low cohesion of remobilized material - as evidenced by intervals with an N equal to or less than 10 strikes/25 cm of penetration.

Two pseudo-sections were obtained from Schlumberger array soundings located along traverses situated opposite each other, on either side of the LEM Avenue (hereinafter referred to as Line 1 and Line 2). The traverses are composed of 9 vertical electrical soundings, equally spaced, with stations 20 m apart from each other. In all soundings 18 measurements were performed and the maximum current electrode distance (AB/2) was set to 100 m. The total length of each traverse was 180 m, totaling 162 measurements of apparent resistivity, as shown in the pseudo-sections Line 1 (Figure 12a) and Line 2 (Figure 13a).

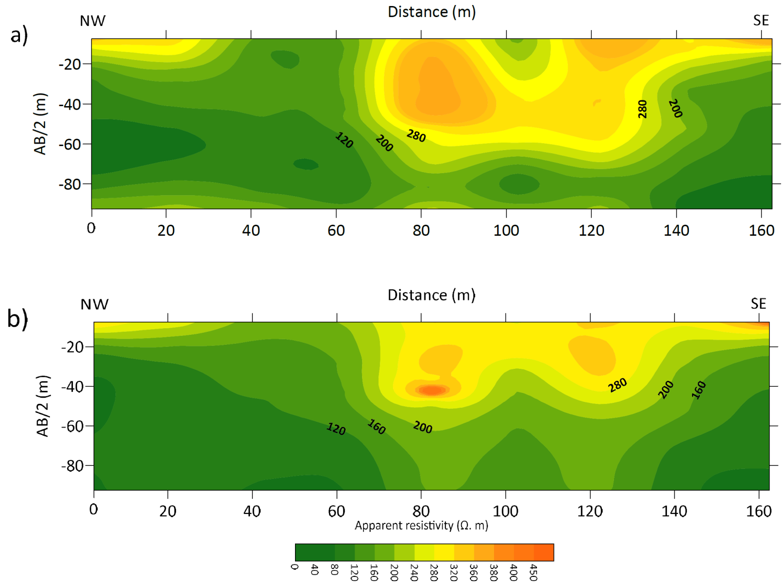

Figure 12: Field case - Line 1: (a) observed and (b) modeled apparent resistivity sections with Schlumberger array. The modeled section (b) was generated from the inversion result with the TV regularization scheme, shown in Figure 15b.

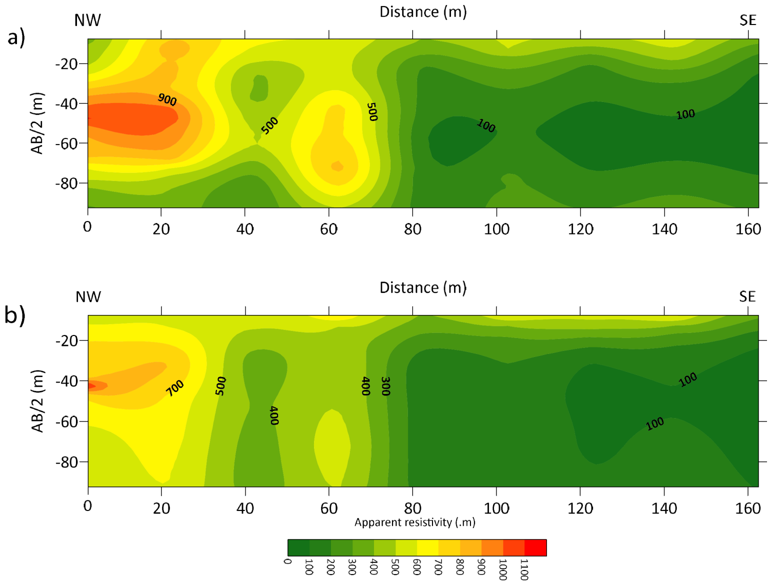

Figure 13: Field case - Line 2: (a) observed and (b) modeled apparent resistivity sections with Schlumberger array. The modeled section (b) was generated from the inversion result with the TV regularization scheme, shown in Figure 16b.

The inversions were executed on a mesh composed of 11 cells in the x direction and 8 cells in the z direction. The mesh is equally spaced along x with mesh size equal to 5 m, but is nonuniform along z, being the available depths equal to 0, 2, 4, 6, 8, 10, 12, 14 and 18 m. As with the synthetic cases, the inversion for each traverse was run 10 times and the average solution was chosen.

All the PEMPSO settings from the synthetic tests were preserved, and the regularization parameter was established as 1.d-2 for Line 1 and 1.d-3 for Line 2, after executing 10 previous inversions. Convergence curves (selected randomly from 10 inversions) for both traverses Line 1 and Line 2 (Figures 14a and 14b), followed a similar pattern as those of the synthetic tests, differing only in that the objective function values of the inversions with the TV constraint were slightly lower when compared to those ones obtained from the inversion with the GS constraint.

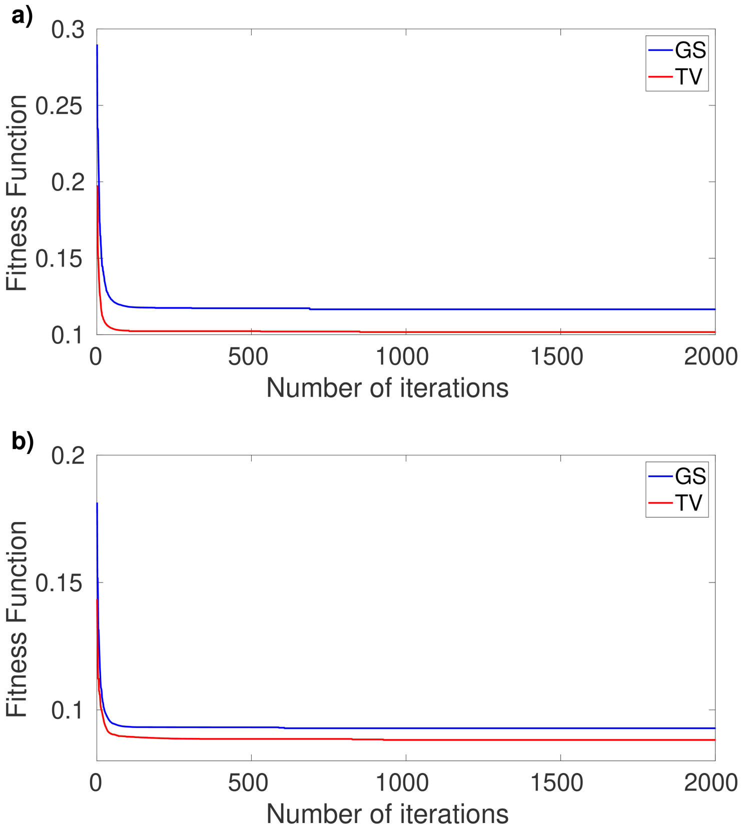

Figure 14: Evolution of the objective function in the inversion of field data from the (a) Line 1 and (b) Line 2 traverses.

7.1 Geological interpretation of the Line 1

For the Line 1, the results of the inversion with the GS and TV constraints are shown in Figures 15a and 15b, respectively.

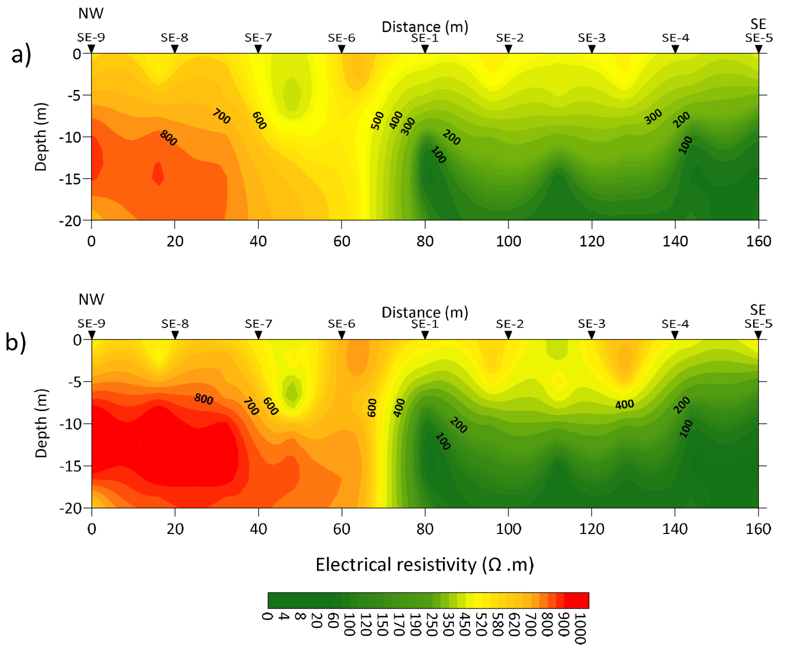

Figure 15: Field case - Line 1: Estimates obtained with the (a) GS and (b) TV constraints.

Both results are pretty similar. The soundings on the preserved area of the Avenue (SE-06 to SE-09) show relatively medium ρ values, underneath the soil, which increase progressively with depth. On the other hand, in the frontal area to the slide (SE-01 to SE-05), ρ values become higher, making it possible to delineate two different materials. In this area, there was a high loss of soil cohesion and air intake, indicated by higher resistivities of 200 $\Omega.m$ until 5 m-depth. Other qualitative results in the estimated resistivity section reflect a zone of crystalline rocks between (SE-01 and SE-02), at a depth of 10 m, ($\rho < 900\ \Omega.m$). The area between the SE-05 and SE-06 soundings, where resistivities below $\rho < 600\ \Omega.m$ appeared, could be linked with the Morro de Águia Fault.

Statistical analysis of the cumulative frequency distribution of the relative errors for all the results of the inversions with Line 1 data, shows, as expected, a worse data fitting than in synthetic case studies. Around 80% of the observations are reproduced with a relative error of up to 30% (Figure 17a). It also can be seen from such analysis that the best data fitting is provided by the solution obtained from the inversion with the TV constraint (Figure 15b).

7.2 Geological interpretation of the Line 2

For the Line 2, the results of the inversion are shown in Figure 16a and Figure 16b.

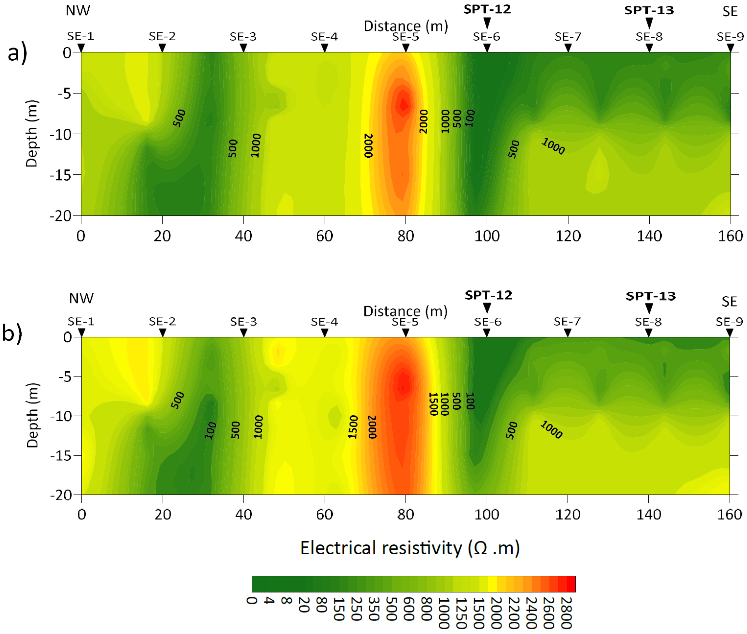

Figure 16: Field case - Line 2: Estimates obtained with the (a) GS and (b) TV constraints.

As in the case of the Line 1, the results of the inversion are similar for both constraints. The soundings centered on the preserved portion of Avenue (SE-1 to SE-2), unlike Line 1, showed relatively high resistivity values underneath the soil, decreasing progressively with depth. The SE-3 station, however, shows relatively low resistivity values ($\rho < 250\Omega.m$), increasing in depth, something that can be interpreted as sub-vertical linings of the Morro do Aguia Fault zone. Underneath the SE-5 station, there is a zone with very high resistivity values ($\rho > 2000\Omega.m$), which is probably associated with the Morro do Aguia Fault.

In the area encompassed by the stations SE-5 to SE-9, the high values of ρ, suggest that the material is remobilized (until 10 meters). To crosscheck our interpretation, we compared the soil samples from SPT measures (SPT-12 and SPT-13) of two percussion soundings, carried out in the affected area. The samples showed different lithologies, mainly variegated sandy silt with mixtures of variegated silty clay and variegated sandy silt. In the SPT-12 sounding, it is noteworthy the altered crystalline rock after 13 m -depth. Both soundings reflect remobilized material with low consistency and cohesion, as indicated by the geotechnical parameter N(strikes/depth), which is roughly equal to 10 hits/25 cm of penetration (Figure 11b).

From Figure 17b, can be depicted that the computed resistivity showed a fairly good match with observed resistivity. The results are similar to those arising from the inversion of Line 1 data, approximately 90% of the observations are reproduced with a relative error of less than 30%.

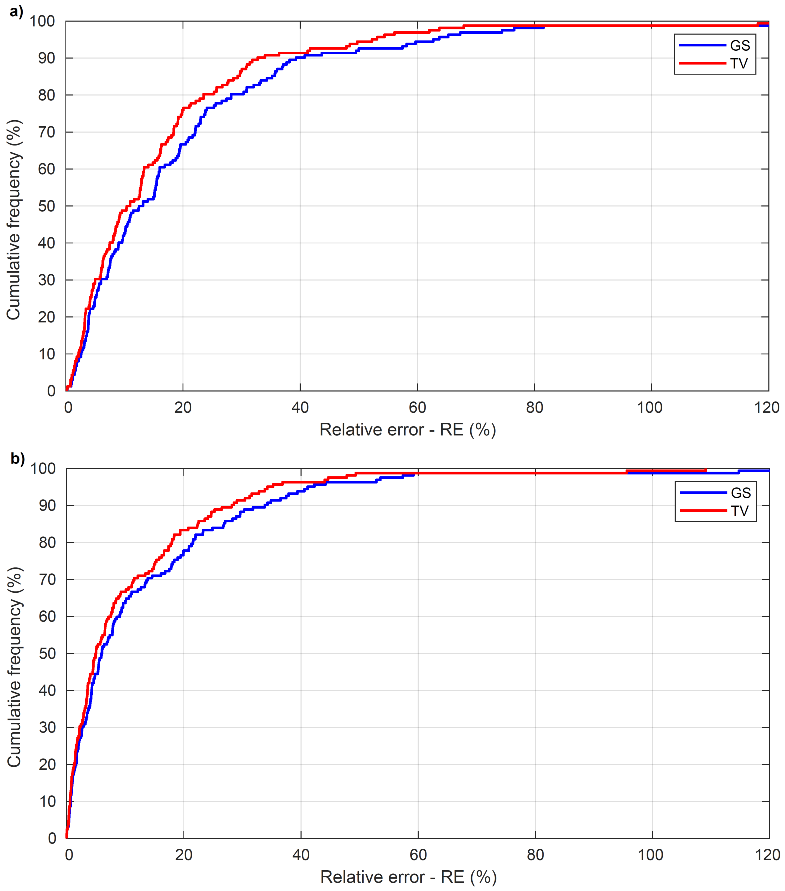

Figure 17: Cumulative distribution for the percentage discrepancy between observed and modeled apparent resistivity data for traverses (a) Line 1 and (b) Line 2

As well as the case above, the best data fitting is provided by the solution obtained from the inversion with the TV constraint (Figure 16b).

8 DISCUSSIONS

The PSO algorithm, like any GOM, allows us easily to incorporate different classes of constraints and data misfit measurements as terms of the objective function. In this work, for example, we explore the sum of squared deviations between observed and calculated resistivity with two types of constraints: GS and TV. However, GOMs can be trapped in local minimums, when a proper exploration on the whole search space is not guaranteed (Juan et al, 2010).

Also, the use of global optimization techniques requires a high computational cost as these techniques generally need a large number of objective function evaluations (Sen et al, 1993). In an effort to overcome such drawback, different approaches using different computing languages and parallel architectures, has been proposed over recent years. Based on a review of those approaches for the specific case of PSO (Lalwani et al, 2019) we have implemented the EMPSO using hybrid parallelism (MPI + OpenMP).

The improvements in the EMPSO allow a better exploration and enhancing of the swarm diversity, whereas the parallel implementation reduces the inversion time. Table 1, presents a quantification of such a reduction in time, taking as a reference the serial time. For performance tests, we have chosen to run inversions with data from an unique synthetic model (plume model) and the field data from Line 1 survey. Tests for the hybrid version used up to two nodes, but those for the pure MPI version, only one. In our analysis, the pure MPI surpasses the hybrid model MPI-OpenMP in both scenarios tested (16×2 and 8×4). The Pure MPI version brought a 17x speedup (ratio between sequential execution time and parallel execution time), for the synthetic model and for the real field data. The performance of the hybrid version depends on the number of MPI processes and OpenMP threads. The two MPI + OpenMP scenarios have different runtimes because their workloads are different, and although in the second case (8×4) the load is greater for each MPI process, the parallelization in the second level with OpenMP does not use 100% of the 4 cores, which justifies the difference of time in favor of the first case (16×2). Speedups for the first and second scenarios (using 32 cores - 1 node) in tests with synthetic and real data were practically equal, 12.3x and 7.1x, respectively. The times in the hybrid version with twice the MPI processes (that is, 2 nodes), were reduced by half, even involving the MPI communication between the two nodes. Obviously, speedups were about twice as high, 30x and 15x for both configurations, respectively, for both data sets. Therefore, it is easy to deduce that for the sizes of model tested, the increase in the number of nodes does not bring many benefits in terms of performance.

In general, in the inversions with both synthetic and real data, the objective function had the least value when the TV constraint was used. This reason is likely attributed to the fact that the GS constraint is very sensitive to the model, especially in models with high/low contrast of resistivity (Menke, 2018). Figures 15 and 16 are good examples of that, where the inverted models exhibit sharp contacts, like faults, sub-vertical structures, alternation of in-situ materials and remobilized material resulting from landslides in the study area. In summary, the PEMPSO proved to be efficient to identify complex geological structures as bodies of simple tabular geometry (Figures 7 and 9).

9 CONCLUSIONS

In this study, a parallel version of the elitist-mutated PSO, which we named PEMPSO is presented to deal with the 2D resistivity imaging problem. GOMs like PSO offer advantages in incorporating different stabilizers in the objective function, but, present some disadvantages, such as slow convergence, risk of trapping in local minima and high computational cost. To alleviate these limitations, in this paper we have proposed a parallelisation of the EMPSO, which incorporate an elitism strategy combined with mutation that help to prevent it from getting stuck in local minima. The proposed hybrid MPI + OpenMP implementation is based on a master-worker communication model, in which the master process exclusively performs PSO-related operations and the worker processes perform the evaluation of the particles. The operations to update the position and velocities of the particles are parallelized via OpenMP. Given the flexibility of the hybrid model, there is always the option of using pure MPI, which for the studied cases proved to be superior. We check the validity of the PEMPSO using two synthetic case studies as well as two field datasets. In both cases, the proposed PEMPSO provides good solutions in an efficient way and turned into a good alternative to traditional techniques generally used for apparent resistivity data inversion.

10 ACKNOWLEDGMENTS

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001. MSc. Abril is also grateful to CAPES for his PhD. fellowship. We thank the Federal University's High Performance Processing Nucleus (NPAD) Rio Grande do Norte (UFRN) for the computational resource used in this research.

11 Conflict of interest

The authors that developed this research notify that there is no conflict of interests by a third party.

References

Arboleda-Zapata M, Guillemoteau J, Tronicke J (2022) A comprehensive workflow to analyze ensembles of globally inverted 2d electrical resistivity models. Journal of Applied Geophysics 196:104,512. https://doi.org/https://doi.org/10.1016/j.jappgeo.2021.104512, URL https://www.sciencedirect.com/science/article/pii/S0926985121002603

Archie GE, et al (1942) The electrical resistivity log as an aid in determining some reservoir characteristics. Transactions of the AIME 146(01):54–62

Barboza FM, Medeiros WE, Santana JM (2019) A user-driven feedback approach for 2d direct current resistivity inversion based on particle swarm optimization. Geophysics 84(2):E105–E124

Başokur AT, Akca I (2011) Object-based model verification by a genetic algorithm approach: Application in archeological targets. Journal of Applied Geophysics 74(4):167–174

Bertete-Aguirre H, Cherkaev E, Oristaglio M (2002) Non-smooth gravity problem with total variation penalization functional. Geophysical Journal International 149(2):499–507

Bortolozo CA, Motta MFB, de Andrade MRM, et al (2019) Combined analysis of electrical and electromagnetic methods with geotechnical soundings and soil characterization as applied to a landslide study in campos do jordão city, brazil. Journal of Applied Geophysics 161:1–14

Chambers JE, Kuras O, Meldrum PI, et al (2006) Electrical resistivity tomography applied to geologic, hydrogeologic, and engineering investigations at a former waste-disposal site. Geophysics 71(6):B231–B239

Chen Z, Zhu X, Liu H, et al (2017) Sequential and simultaneous joint inversion of resistivity and ip sounding data using particle swarm optimization. Journal of Earth Science 28(4):709–718

Chunduru RK, Sen MK, Stoffa PL, et al (1995) Non-linear inversion of resistivity profiling data for some regular geometrical bodies1. Geophysical Prospecting 43(8):979–1003

Clerc M (1999) The swarm and the queen: towards a deterministic and adaptive particle swarm optimization. In: Proceedings of the 1999 congress on evolutionary computation-CEC99 (Cat. No. 99TH8406), IEEE, pp 1951–1957

Constable SC, Parker RL, Constable CG (1987) Occam's inversion: A practical algorithm for generating smooth models from electromagnetic sounding data. Geophysics 52(3):289–300

Dey A, Morrison H (1979) Resistivity modeling for arbitrarily shaped two dimensional structures, part i: Theoretical formulation. Lawrence Berkeley Laboratory Report (LBL-5223)

Eberhart RC, Shi Y (2000) Comparing inertia weights and constriction factors in particle swarm optimization. In: Proceedings of the 2000 congress on evolutionary computation. CEC00 (Cat. No. 00TH8512), IEEE, pp 84–88

Engelbrecht AP (2007) Computational intelligence: an introduction. John Wiley & Sons

Farquharson CG, Oldenburg DW (1998) Non-linear inversion using general measures of data misfit and model structure. Geophysical Journal International 134(1):213–227

Farquharson CG, Oldenburg DW (2004) A comparison of automatic techniques for estimating the regularization parameter in non-linear inverse problems. Geophysical Journal International 156(3):411–425

Ferreira NR, Porsani MJ, Oliveira SPd (2003) A hybrid genetic-linear algorithm for 2d inversion of sets of vertical electrical sounding. Revista Brasileira de Geofísica 21(3):235–248

Gill PE, Murray W, Wright MH (2019) Practical optimization. SIAM deGroot Hedlin C, Constable S (1990) Occam's inversion to generate smooth, two-dimensional models from magnetotelluric data. Geophysics 55(12):1613–1624

Holland JH (1975) Adaptation in Natural and Artificial Systems. Ann Arbor: University of Michigan Press

Ingber L (1989) Very fast simulated re-annealing. Mathematical and computer modelling 12(8):967–973

Jha MK, Kumar S, Chowdhury A (2008) Vertical electrical sounding survey and resistivity inversion using genetic algorithm optimization technique. Journal of Hydrology 359(1-2):71–87

Juan L, Esperanza G, José P, et al (2010) Pso: a powerful algorithm to solve geophysical inverse problems: application to a 1d-dc resistivity case. J Appl Geophys 71:13–25

Kennedy J (2006) Swarm intelligence. In: Handbook of nature-inspired and innovative computing. Springer, p 187–219

Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of ICNN'95-international conference on neural networks, IEEE, pp 1942–1948

Kirkpatrick S, Gelatt CD, Vecchi MP (1983) Optimization by simulated annealing. science 220(4598):671–680

Lalwani S, Sharma H, Satapathy SC, et al (2019) A survey on parallel particle swarm optimization algorithms. Arabian Journal for Science and Engineering 44(4):2899–2923

Liu B, Li S, Nie L, et al (2012) 3d resistivity inversion using an improved genetic algorithm based on control method of mutation direction. Journal of Applied Geophysics 87:1–8. https://doi.org/https://doi.org/10.1016/j.jappgeo.2012.08.002, URL https://www.sciencedirect.com/science/article/pii/S0926985112001425

Loke M, Chambers J, Rucker D, et al (2013) Recent developments in the direct-current geoelectrical imaging method. Journal of applied geophysics 95:135–156

Medeiros W (1987) The resistivity method applied to the hydrogeology of crystalline terrains: A problem of two-dimensional modelling. Master's thesis, Universidade Federal da Bahia

Menke W (2018) Geophysical data analysis: Discrete inverse theory. Academic press

Metcalfe TS, Charbonneau P (2003) Stellar structure modeling using a parallel genetic algorithm for objective global optimization. Journal of Computational Physics 185(1):176–193

Moura FA, Silva SA, de Araújo JM, et al (2020) Progressive matching optimisation method for fwi. Journal of Geophysics and Engineering 17(2):357–364

Nagesh Kumar D, Janga Reddy M (2007) Multipurpose reservoir operation using particle swarm optimization. Journal of Water Resources Planning and Management 133(3):192–201

Pace F, Santilano A, Godio A (2019) Particle swarm optimization of 2d magnetotelluric data. Geophysics 84(3):E125–E141

Pace F, Santilano A, Godio A (2021) A review of geophysical modeling based on particle swarm optimization. Surveys in Geophysics pp 1–45

Portniaguine O, Zhdanov MS (1999) Focusing geophysical inversion images. Geophysics 64(3):874–887

Regińska T (1996) A regularization parameter in discrete ill-posed problems. SIAM Journal on Scientific Computing 17(3):740–749

Rudin LI, Osher S, Fatemi E (1992) Nonlinear total variation based noise removal algorithms. Physica D: nonlinear phenomena 60(1-4):259–268

Schwarzbach C, Börner RU, Spitzer K (2005) Two-dimensional inversion of direct current resistivity data using a parallel, multi-objective genetic algorithm. Geophysical Journal International 162(3):685–695

Sen MK, Stoffa PL (2013) Global optimization methods in geophysical inversion. Cambridge University Press

Sen MK, Bhattacharya BB, Stoffa PL (1993) Nonlinear inversion of resistivity sounding data. Geophysics 58(4):496–507

Sharma SP (2012) VfsaresâĂŤa very fast simulated annealing fortran program for interpretation of 1-d dc resistivity sounding data from various electrode arrays. Computers & Geosciences 42:177–188

Shaw R, Srivastava S (2007) Particle swarm optimization: A new tool to invert geophysical data. GEOPHYSICS 72(2):F75–F83. https://doi.org/10.1190/1.2432481, URL https://doi.org/10.1190/1.2432481, https://arxiv.org/abs/https://doi.org/10.1190/1.2432481

Sigdel A, Adhikari RK (2020) Application of electrical resistivity tomography (ert) survey for investigation of the landslide: a case study from taprang landslide, kaski district, west-central nepal. Journal of Nepal Geological Society 60:103–115

Szu H, Hartley R (1987) Fast simulated annealing. Physics letters A 122(3-4):157–162

Tikhonov AN, Arsenin VY (1977) Solutions of ill-posed problems. New York 1:30

Wahba G (1990) Spline models for observational data, siam, philadelphia, 1990. MR 91g 62028

3

Artigo 2

Este manuscrito foi submetido na revista Acta Geophysica de Qualis B1, e foi intitulado: "*1.5D time domain electromagnetic inversion using global optimization methods and parallel processing*".

Nas seguintes páginas encontra-se o manuscrito desenvolvido nesta pesquisa.

1.5D time domain electromagnetic inversion using global optimization methods and parallel processing

Jorge L. Abril[1*], Marcos A. Vasconcelos[1†], Francisco M. Barboza[2†] and Cassiano. Bortolozo[3†]

[1*]Center for Research in Geophysics and Geology (CPGG), Federal University of Bahia (UFBA), Rua Barão de Jeremoabo-Ondina, Salvador, 40170-115, Bahia, Brazil.
[2]Computing and Technology Department (DCT), Federal University of do Rio G do Norte, Street, Caico, 59300-000, Rio Grande do Norte, Brazil.
[3]National Center for Monitoring and Early Warning of Natural Disasters, CEMADEN, Estrada Doutor Altino Bondensan, 500, São José dos Campos, 12247-016, São Paulo, Brazil.

*Corresponding author(s). E-mail(s): jlab66@gmail.com; Contributing authors: marcos.vasconcelos@ufba.br; marcio.barboza@ufrn.br; cassianoab@gmail.com;
†These authors contributed equally to this work.

Abstract

We applied three Global Optimization Methods (GOMs) to invert time-domain electromagnetic data (TDEM) on the 1.5D imaging of resistivity in the subsurface. These stochastic methods incorporate different kinds of stabilizers in the objective function due to the easiness of implementation in their algorithms and being computationally efficient. Nevertheless, global optimization methods, like any other based on swarm intelligence, cost much computational time, including problems with a high number of unknown parameters. Furthermore, time is easily increased when the forward modeling involves calculations that are time-consuming such as Gaver-Stehfest inverse transforms, Hankel transforms, and others. To overcome this difficulty, we developed a parallel pure MPI version of each GOM, allowing us to distribute the computation among several cores of a cluster. The performances of the classic version of Particle Swarm Optimization (PSO), Grey Wolf Optimizer (GWO), and Whale Optimization Algorithm (WOA) using MPI parallelism for solving 1.5D TDEM inverse problems were compared to invert a set of synthetic and real data. The principal outcomes show that these GOMs reproduce a coherent distribution of subsurface resistivity in either synthetic models or real data. WOA and PSO exhibit better computational performances, converging first than GWO. WOA provided better performance in the final value achieved of the cost function than PSO and GWO, and pure MPI parallelism provided a 17x speedup in the computation time for both synthetic models and a 50x speedup in the computing time for the real data. To evaluate better this comparison, we analyzed the solutions using L1 and L2 norms to identify smooth and sharp structures. Additionally, we have diminished the computational time execution with the parallel solution (MPI version of each traditional stochastic method) against the sequential processing.

Keywords: constrained inversion, TDEM, global optimization methods, parallel computing

1 Introduction

The time-domain electromagnetic method (TDEM) or transient electromagnetic (TEM) is a technique used over 40 years in imaging the distribution of resistivity in the subsurface. This technique involves the induction of an electric current into the ground, generated in most cases by a big transmitter loop sized from tens to hundreds of meters in length. The easiness of data collection and the low-cost acquisition in field campaigns allowed it to become a valuable tool in several hydrogeological and environmental investigations.

Once data are collected, the user needs to apply some processing techniques before using them in the inversion process. After that, an important step is required to begin the interpretation of results that is only possible through flexible inversion processes. Existing different approaches to solving nonlinear inverse problems, both, the local gradient-based and global search methods, involve the use of stabilizers in the cost function to reduce its ill-posed nature (Silva et al, 2001). For the first aforementioned case, local methods can converge fast but are highly dependent on an good initial model (Gill et al, 2019). In the last case, global methods do not need a good starting point because the search space is made by trajectories selected randomly until a minimum global is attained. These stochastic, Global Optimization Methods (GOMs) have developed their techniques to avoid getting stuck in local minima.

Compared to gradient-descent algorithms, GOMs have no restrictions concerning initial models due to the random nature of search space, but they are quite time-consuming because several runs are needed to evaluate the cost function (Sen and Stoffa, 1995). Algorithms as PSO (Kennedy and Eberhart, 1995), Grey Wolf Optimizer (GWO) (Mirjalili et al, 2014), Simulated Annealing (SA) (Kirkpatrick et al, 1983) and Genetic Algorithm (GA) (Goldberg, 1989) are classified into this group of meta-heuristic algorithms.

In spite of the computational cost, over the last 30 years, GOMs have been employed in the inversion of TDEM data to determine the distribution of resistivity in the subsurface. Monteiro and El-Kaliouby (2010) used SA, PSO, and a local optimization technique in the individual and joint inversion of vertical electrical soundings (VES) and TEM data. Similarly, Cheng et al (2015) applied joint inversion of DC resistivity and TDEM data using PSO to detect roadways in a coal mine in the east of China, and for the same purpose Chandra et al (2017) and Agarwal et al (2018) used a GWO to an electrical and gravity-magnetic data set, respectively. Li et al (2018) introduced an improved GWO applied to geophysics inversion, and Godio and Santilano (2018) used the PSO for the inversion of electromagnetic soundings, applied to audio-magnetotelluric (AMT) and magnetotelluric (MT) data. These studies show that PSO and GWO are up to date, the most well-known methods not only in the TEM data but also in other geophysical methods.

In addition to these traditional methods, the Whale Optimization Algorithm (WOA) created by Mirjalili and Lewis (2016) is a recent meta-heuristic method and underused in Earth Sciences. Just a couple of papers were published and are restricted to 1D inverse problems such as Abdelazeem et al (2019) and Liang et al (2022). Apart from the 1D works mentioned above, 1.5D and 2D inversions of TEM data using GOMs in geophysics are quite rare and computationally demanding, limiting their use to local optimization methods that are faster in converging, but resort to derivative calculations to achieve the minimum of cost function. Keeping in mind this limitation, it is clear that the use of GOMs would avoid the calculate of complex derivatives and allow to incorporate with easiness several stabilizers in the cost function. Furthermore, supported in the use of scientific computing clusters, the challenge can be overcome. Taking care of the aforementioned vantages that GOMs provide in geophysics and based on Barboza et al (2018), we developed a new comparative study analyzing the performance of parallel canonical versions of WOA, PSO, and GWO to solve the 1.5D TEM nonlinear inversion problem, imposing lateral continuity stabilizers on the parameters model (e.g., Auken and Christiansen (2004) and Santos (2004)), but following 1D forward modeling in the form of the layered model, such as (Auken et al, 2005). Particularly, we analyzed the L_1 and L_2 norms as a way to infer small discontinuities in the lateral continuity associated with sharp structures and faults (Loke et al, 2003). To reduce the high computing cost related to the high number of evaluations of the cost function to achieve a global minimum (Sen and Stoffa, 2013), we developed a parallel MPI implementation of each GOM where assessments are carried out parallelly using numerous MPI workers. We approached the efficiency of each algorithm through tests with synthetic and real data. In both cases, the outcomes demonstrated to be efficient in recovering the resistivity model. We organized the manuscript with a discussion of the 1.5D inversion of TEM, and then we focused on addressing some general considerations about the parallel processing scheme used in WOA, PSO, and GWO.

2 Nonlinear inversion of 1.5D TEM data

To clarify the 1.5D inversion scheme used in this research, we set a couple of TDEM soundings (ns) disposed in a parallel line forming transverses. Each sounding was determined by a certain number of layers (nl), that involve ns×nl resistivity parameters (organized in an array $\boldsymbol{\rho}$) and ns×(nl-1) thickness parameters (organized in an array $\mathbf{h}$) for each layer, forming a total number of parameters $N_{par} = ns(nl - 1)$. In addition, we set the N_{par} as an array that contains all model parameters ($\mathbf{m}$) and ($\mathbf{d}^o$) as a vector that contains the total number of observed apparent resistivity measurements (N_{obs}) along the transverse.

Assuming it, we can represent the set of observed apparent resistivity data, that is:

$$\mathbf{d}^o = \mathbf{d}^p(\mathbf{m}) + \xi \tag{1}$$

where $\mathbf{d}^p(\mathbf{m})$ is a vector of apparent resistivity data predicted, obtained through the TDEM responses of 1D layered models (Anderson, 1982; Sandberg, 1988) and ξ is an array of discrepancies. Minimizing the norm of ξ is possible to recover the model parameters ($\mathbf{m}$). However, as the solution becomes unstable, it is necessary to incorporate stabilizers. For solution stabilization, we defined a three-terms parametric functional as follows (Barboza et al, 2018):

$$P[\mathbf{d}^p(\mathbf{m}), \mathbf{d}^o(\mathbf{m});\ \mathbf{m}] = \Phi_d[\mathbf{d}^p, \mathbf{d}^o] + \lambda_\rho S_\rho(\mathbf{m}) + \lambda_h S_h(\mathbf{m}) \tag{2}$$

The equation above details three terms on the right side of parametric functional P. The first term Φ_d corresponds to the vector of discrepancies between observed and calculated data, and the second and third ones are the stabilizers for resistivities $S_\rho(m)$ and thickness layers $S_h(m)$ of the same layer, below neighbors pairs of soundings, respectively. Also, in equation (2), λ_ρ and λ_h are the pair of Lagrange multipliers used for mismatch compensation between data and the stabilizers incorporated (Hansen, 1992). In this investigation, we used two stabilizers for computational modeling with synthetic and field data: (i) the L_2 norm lateral continuity stabilizer used to smooth high heterogeneities between model parameters (DeGroot Hedlin and Constable, 1990) and (ii) the L_1 norm lateral continuity stabilizer that has the opposite effect and allow discontinuous solutions (Bertete-Aguirre et al, 2002). The following section describes the mathematical equations for both stabilizers in detail.

3 Equations of the lateral stabilizers

Different approaches of inversion have been used for incorporating stabilizers in the cost function. Within these techniques, lateral continuity stabilizers or commonly called Lateral Continuity Constraints (LCC), have been widely used to reduce the effects of high contrasts of 2D and 3D structures (e.g., Auken and Christiansen 2004; Santos 2004). In the particular case of Auken and Christiansen (2004), they only use laterally constrained inversion (LCI) with the L_2 norm over the layers and resistivities of the model, resulting in a low-sensitivity method to detect sharp structures and faults. Keeping in mind this disadvantage, the use of L_1 norm becomes helpful in overcoming this issue. In addition, these inversion approaches cannot give us the easiness of modifying the cost function because they are based on local methods. In contrast, GOMs have the flexibility to modify or incorporate constraints (L_1 or L_2 norm) or any other kind of stabilizers in the cost function. We described below both types of stabilizers.

3.1 L_2 norm stabilizer (SL_2)

Based on equation (2), the Φ_d, and S_ρ and S_h terms are based on the least-squares discrepancies between data and model parameters, respectively. It means a smooth representation of the physical property to be investigated (Constable et al, 1987). The depiction mentioned above by each term is given as follows:

$$\Phi_d = \frac{1}{N_{obs}|\sigma_d^2|}||log((\mathbf{d}^o) - log(\mathbf{d}^p))||_2^2, \tag{3}$$

$$S_\rho = \frac{1}{[(ns-1)nl]|\sigma_\rho^2|}||\mathbf{D}_\rho[log(\boldsymbol{\rho})]||_2^2, \tag{4}$$

$$S_h = \frac{1}{[(ns-1)(nl-1)]|\sigma_h^2|}||\mathbf{D}_h\mathbf{h}||_2^2. \tag{5}$$

where $||\cdot||_2^2$ expresses the squared L_2 norm and the σ_d^2, σ_ρ^2, σ_h^2 terms on the denominator of equations (3)-(5) are the variances applied to normalize the parametric functional P of equation 2, during the inversion process. These values can be adjusted after several attempts of simulations. Furthermore, the $\mathbf{D}_\rho$ and $\mathbf{D}_h$ operators symbolize the matrices of first-order derivatives, whose sizes are described in detail by (Barboza et al, 2018).

3.2 L_1 norm stabilizer (SL_1)

This stabilizer uses the least-absolute discrepancies, which means the opposite behavior for the L_2 norm stabilizer. It promotes high discontinuities between model parameters, favoring the search for sharp structures and faults (Rudin et al, 1992). The equations are shown below:

$$\Phi_d = \frac{1}{N_{obs}|\sigma_d^2|^{1/2}}||log((\mathbf{d}^o) - log(\mathbf{d}^p))||_1, \tag{6}$$

$$S_\rho = \frac{1}{[(ns-1)nl]|\sigma_\rho^2|^{1/2}}||\mathbf{D}_\rho[log(\boldsymbol{\rho})]||_1, \tag{7}$$

$$S_h = \frac{1}{[(ns-1)(nl-1)]|\sigma_h^2|^{1/2}}||\mathbf{D}_h\mathbf{h}||_1, \tag{8}$$

where $||\cdot||_1$ expresses the L_1 norm, with the other parameters being the same as in the L_2 norm.

4 Considerations about Global Optimization Methods

Meta-heuristic global optimization techniques such as PSO, GWO, and WOA have become well-known among scientific communities. All these techniques emulate the social conduct of a group of individuals in nature. In particular, WOA, PSO, and GWO algorithms find promissory regions to locate the global minimum, a common objective among them. To find this minimum, each algorithm uses certain mechanisms that emulate how animals locate the prey and obtain food in the search space. Therefore, in this section, we described each algorithm's principal characteristics and mathematical equations and addressed the general stopping condition applied in all simulations.

4.1 Particle Swarm Optimization

This method, implemented by Kennedy and Eberhart (1995), is a robust evolucionary strategy inspired by social behavior of animal species living in colonies like birds, ants or fish. Their bevy behavior usually works to achieve a shared objective (e.g., to search meal). In this pursuit, each entity uses its own experience and the bevy experience to attain the common goal (Kennedy, 2006).

For the case of PSO algorithm, the search for food is controlled by each particle i, having its position ($\mathbf{m}_i$), and its velocity ($\mathbf{v}_i$) within a search space Π. The performance of each particle is evaluated using equation 2, which expresses the fitness of the particle's positions. The best position reached by the i-th particle (noted as personal best) is described by ($\mathbf{p}_i$), while the best location reached by nearby particles in the bevy (noted as global best) is depicted by ($\mathbf{g}_i$). Along each iteration k, each particle changes its position corresponding to its recent velocity as in

$$\mathbf{v_i}^{k+1} = \chi[\mathbf{v_i}^k + c_1 \otimes \mathbf{r}_1(\mathbf{p}_i - \mathbf{m}_i{}^k) + c_2 \otimes \mathbf{r}_2(\mathbf{g}_i - \mathbf{m}_i{}^k)]. \tag{9}$$

$$\mathbf{m_i}^{k+1} = \mathbf{m_i}^k + \mathbf{v_i}^{k+1} \tag{10}$$

where $i = 1, 2, \ldots, n_{part}$, the parameters χ, c_1 and c_2, $\mathbf{r}_1$ and $\mathbf{r}_2$ are the constriction factor introduced by Clerc (1999), two constants and two random vectors enclosed by [0,1], respectively. The symbol $\otimes$ implies point-to-point vector product. After several inversion tests, we configured $\chi = 0.73$, $c_1 = 2.9$ and $c_2 = 1.2$ for all simulations. Similarly, we defined a number of particles $n_{part} = 12 \times n_p$, being n_p the total number of model parameters. Finally, we chose a value of 5000 iterations in all simulations.

4.2 Grey Wolf Optimizer

For almost ten years, the Grey Wolf Optimizer (GWO) has been used widely by many scientists in different fields of knowledge, including geophysics (e.g., Chandra et al 2017; Agarwal et al 2018; Li et al 2018). This probabilistic technique introduced by Mirjalili et al (2014) emulates the hierarchical social behavior and the hunting approach of grey wolves (GW) in nature. The social classification structure is comprised of the alpha, beta, delta, and omega categories of wolves. The best three wolves in the group (alpha, beta, and delta) control omega wolves' guidance towards a shared goal (e.g., to find the prey). To achieve this goal Muro et al (2011) defined three keys to explain the hunting approach used by the wolves: (i) searching, enchasing, and focusing on the prey; (ii) tracking, involving, and suffocating the victim up to its quits moving, and (iii) attack against the prey. The first principle is mathematically modeled according to (Mirjalili et al, 2014) as

$$\begin{aligned}\mathbf{D} &= |\mathbf{C}.\mathbf{m}_p(k) - \mathbf{m}(k)|,\\ \mathbf{m}(k+1) &= \mathbf{m}_p(k) - \mathbf{A}.\mathbf{D},\end{aligned} \tag{11}$$

where $\mathbf{m}_p$ is the prey's position vector at each iteration (k) and $\mathbf{D}$ is the vector between each wolf and the prey. The other constant coefficients $\mathbf{A}$ and $\mathbf{C}$ in equation (11) are computed as follow:

$$\begin{aligned}\mathbf{A} &= 2\mathbf{a}.r_1 - \mathbf{a},\\ \mathbf{C} &= 2\mathbf{r}_2,\end{aligned} \tag{12}$$

where r_1 and r_2 are two random vector within [0,1] and $\mathbf{a}$ is a linear vector that decreases from 2 to 0 along the iterations.

In the second principle, the hunting step is modeled by the alpha, beta, and delta wolves, with the alpha wolf being considered the best solution among them. These three first solutions that have been attained until a given iteration are saved and used to force the resting wolves in the pack (e.g., omega wolves) to update their locations in the search space according to the optimal position. The approach used in updating the wolves' positions is described in

$$\mathbf{m}(k+1) = \frac{\mathbf{m}_1}{3} + \frac{\mathbf{m}_2}{3} + \frac{\mathbf{m}_3}{3}, \tag{13}$$

where $\mathbf{m}_1$, $\mathbf{m}_2$, $\mathbf{m}_3$ are estimated and computed using

$$\begin{aligned} \mathbf{m}_1 &= \mathbf{m}_\alpha - \mathbf{A}_1.\mathbf{D}_\alpha, \\ \mathbf{m}_2 &= \mathbf{m}_\beta - \mathbf{A}_2.\mathbf{D}_\beta, \\ \mathbf{m}_3 &= \mathbf{m}_\delta - \mathbf{A}_3.\mathbf{D}_\delta, \end{aligned} \tag{14}$$

where $\mathbf{m}_\alpha$, $\mathbf{m}_\beta$, and $\mathbf{m}_\delta$ are the three fittest wolves (solutions) in the pack at a certain iteration k. The vectors $\mathbf{A}_1$, $\mathbf{A}_2$ and $\mathbf{A}_3$ are calculated according equation (12), whilst $\mathbf{D}_\alpha$, $\mathbf{D}_\beta$ and $\mathbf{D}_\delta$ are computed using

$$\begin{aligned} \mathbf{D}_\alpha &= \mathbf{C}_1.\mathbf{m}_\alpha - \mathbf{m}, \\ \mathbf{D}_\beta &= \mathbf{C}_2.\mathbf{m}_\beta - \mathbf{m}, \\ \mathbf{D}_\delta &= \mathbf{C}_3.\mathbf{m}_\delta - \mathbf{m}, \end{aligned} \tag{15}$$

where $\mathbf{C}_1$, $\mathbf{C}_2$ and $\mathbf{C}_3$ are vectors computed according to equation (12).

Finally, in the third principle, (i.e., the hunting mechanism) used by the pack of wolves is mathematically modeled based on the vector **A** that is used in equation (12). **A** is between the range [-2a,2a], and the value of **a** is updated along the iterations resorting to

$$\mathbf{a} = 2 - \frac{2 \times k}{maxit}, \tag{16}$$

where $maxit$ is the maximum number of iterations. The approach of exploration in the pack is controlled when $|\mathbf{A}| < 1$ and the exploitation is controlled when $|\mathbf{A}| > 1$. Similarly to the PSO algorithm, we used a certain number of particles $n_{part} = 12 \times n_p$, being n_p the total number of model parameters, and addressed a value of 5000 iterations in all simulations.

4.3 Whale Optimization Algorithm

The Whale Optimization Algorithm (WOA) forms part of the swarm intelligence (SI) algorithms, whose essence is based on the nature-inspired behavior of organisms in nature. Introduced and developed by Mirjalili and Lewis (2016), this technique simulates the natural hunting conduct of humpback whales for obtaining food. Its adoption has risen in the last 5 years among different researches comprising several scientific fields (e.g., Chen et al 2019; Lee and Zhuo 2021). Particularly, the use of WOA in geophysical applications is very restricted and reduced to a few studies (e.g., Abdelazeem et al 2019; Liang et al 2022) in comparison to the well known PSO and GWO algorithms discussed above.

Similarly, with the GWO algorithm, WOA is based on three similar phases to explain its functioning: (i) involving the prey, (ii) exploitation phase (i.e., air-ball net assaulting against the prey), and (iii) exploration phase (i.e., the pursuit for the prey). In the first phase, the whale identifies the prey's location, the target, or the best position. The other individuals move to this optimal solution. To compute this phase, we used the equations (11) and (12) above mentioned. The second phase involves two systems of depredation of whales (i.e., shrinking and enclosing system and spiral update location).

The first system is controlled reducing the a value in the equation (11), (12) and (16). The second system first calculates the distance between the individual whale and the best current position (i.e., the target prey).Then, it emulates the process of capturing food in the form of a spiral of each whale. This system can be computed as

$$\mathbf{m}(k+1) = \mathbf{D'}.e^{bl}.cos(2\pi l) + \mathbf{m}^*(k), \tag{17}$$

where $\mathbf{D'} = |\mathbf{m}^*(k) - \mathbf{m}(k)|$, that is the distance between the i-th whale and the current best location, $\mathbf{m}(k)^*$ is the prey's location vector (i.e., current best solution) at the k-th iteration, b is a constant value defined to evaluate the logarithmic spiral form, and l is a random value within [-1,1]. In this depredation system, whales swim simultaneously following a shrinking circle model and spiral-shaped path. For this simultaneous system, we suppose that there is a probability of 50% to choose either the shrinking circle's strategy or the spiral-shaped's strategy. This is computed as

$$\mathbf{m}(k+1) = \begin{cases} \mathbf{m}^* - \mathbf{A}.\mathbf{D}, \text{p} < 0.5, \\ \\ \mathbf{D'}.e^{bl}.cos(2\pi l) + \mathbf{m}^*(k), p \geq 0.5 \end{cases} \tag{18}$$

where p is a random number within [0.1].

Finally, in the exploitation phase, the exploration mechanism of the whales is controlled by the $\mathbf{A}$ vector to force whales to move away from a reference whale. This task contributes to update the location vector of each whale based on a random whale's position and perform a global search. This is mathematically computed as

$$\begin{aligned}\mathbf{m}(k+1) &= \mathbf{m}^{rand} - \mathbf{A}.\mathbf{D}, \\ \mathbf{D} &= |\mathbf{C}.\mathbf{m}^{rand} - \mathbf{m}(k)|,\end{aligned} \tag{19}$$

where $\mathbf{m}^{rand}$ is a random location vector. Similarly with PSO and GWO, we configured 5000 iterations and a number of particles of $n_{part} = 12 \times n_p$, being n_p the total number of parameters, for all simulations.

4.4 General stopping condition

There are several rules to determine the general stopping condition for GOMs approaches: (i) a predefined maximum number of iterations; (ii) a maximum number of cost function evaluations; (iii) a break condition ends of iterating the algorithm once has reached a specific threshold, defined by the user. In this research, we used the latter condition proposed by Barboza et al (2018), in which a stopping condition is accepted when the difference of the parametric functional $P(\mathbf{m})$ between one previous iteration $(k-1)$ and the current (k), continues approximately equal below a determined tolerance δ_P that is,

$$P(\mathbf{m})_k = |P(\mathbf{m}_k) - P(\mathbf{m}_{k-1})| \leq \delta_P, for\ N_i\ successive\ times, \tag{20}$$

where || denotes the absolute value, and P_k and P_{k-1} are the best solutions attained in the (k) and $(k-1)$ iterations, respectively. We configured for all simulations a $N_i = 500$, that is the number of iterations in which the equation (20) is fulfilled, and the threshold value was configured as $\delta_P = 10^{-6}$.

5 Parallel implementation of the GOMs

Given that GOMs need a high number of cost function evaluations, it is simple to understand that the use of serial versions would demand a high computational cost. To decrease the processing time, we coded a parallelized classical version of each algorithm (WOA, PSO, and GWO). In the parallel code implementation, we used the MPI library to distribute the computing across several processes, following the scheme applied by Metcalfe and Charbonneau (2003), where a master process assigns tasks to other MPI processes (workers) on the requirement. To clarify the parallelized code's general functioning, we established a common terminology of "agent" to refer to a whale in WOA, a particle in PSO, and a wolf in GWO. Additionally, we used the common nomenclature of "specific subroutine" to refer to the principal operations performed in WOA, PSO, and GWO, respectively. These operations are well described in section (4).

5.1 General functioning of the parallelized code

In general, the functioning of the parallel code begins in the main algorithm, which is invoked by the master process a "specific subroutine". Once the population is created, the agents are distributed among the available worker processes resorting to a subroutine that computes the cost function in parallel (mpi-cost subroutine), sending an agent (one array that contains the model parameters) to each worker process. The subroutine waits for the responses of other workers and sends new agents to each worker since the earlier computation has finished (cost function assessment). A determinate worker process will perform 1.5D forward modeling linked to the agent received and will also evaluate the cost function according to the equation (2). The cost function value returns to a master process, and again a worker process is available to receive an agent from him. This programming paradigm allows the master process to maintain all worker processes busy if there are no sufficient agents available to release. Once the cost function value is allocated to the agents in each iteration, the mpi-cost subroutine returns to the "specific subroutine" that executes its own operations.

6 Synthetic examples

To evaluate the performance of the parallelized versions of WOA, PSO, and GWO, we elaborated two synthetic models (Figure 1a and 1b) referred as models A and B.

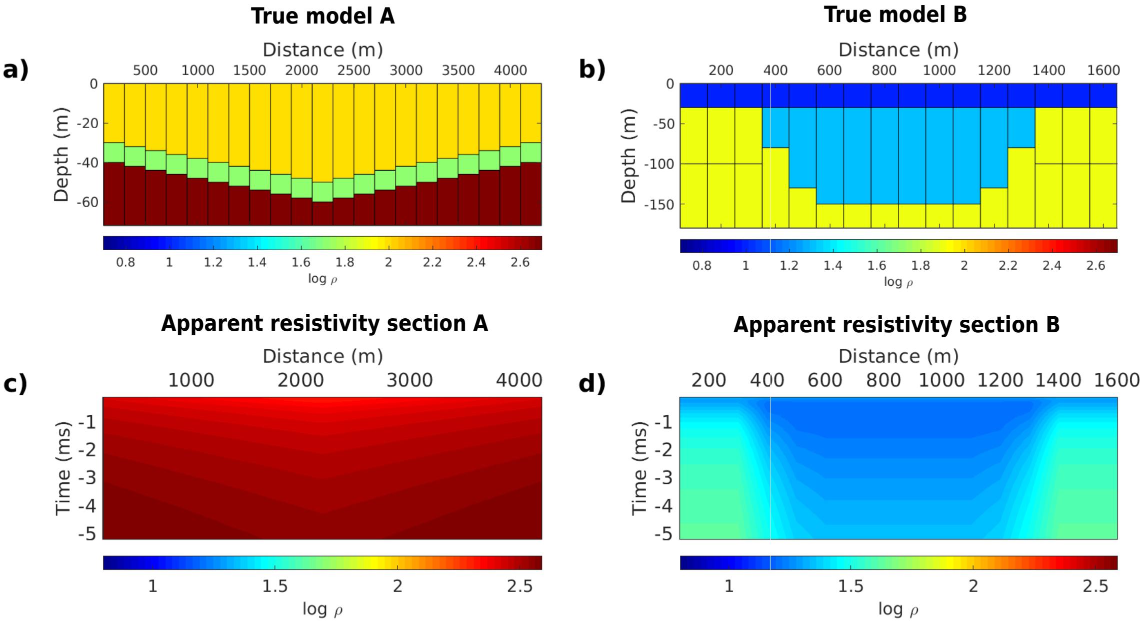

Figure 1: Synthetic data models: (a) A and (b) B True resistivity model. (c) Apparent resistivity section modeled with 21 transient electromagnetic soundings equally spaced 250 m apart), (d) Apparent resistivity section modeled with 16 transient electromagnetic soundings equally spaced (100 m apart).

Both models have layers cake-shaped and represent sedimentary rocks, considered hydrogeological targets. In the model A, we elaborated a sedimentary basin having three layers of $100\Omega.m$ (yellow color), $50\Omega.m$ (green color) and

$500\Omega.m$ (dark brown color) resistivity values. The basin has a very slight slope ($< 5\%$), where the lateral distance between neighboring soundings is larger than the vertical depth. In model B, we created a glacial model, emulating a paleochannel model (Auken et al, 2008). The valley is made up by sand with medium to the high resistivity of 80 $\Omega.m$ (blue color), filled with glacio-lacustrine clay of 20 $\Omega.m$ (yellow color), and covered all by clay till of 10 $\Omega.m$ (dark blue color). TDEM soundings in 21 sites separated by 200 m were generated using a 2D forward modeling code (Cockett et al, 2015) for model A and 16 sites separated by 100 m for model B. Also, we corrupted the synthetic data with Gaussian random noise with zero mean and standard deviation of 2.5% at the data.

Figures 1c and 1d show pseudo-sections of apparent resistivity, for the model A and B, respectively. For both models, the apparent resistivity data were carried out using a central-loop configuration with a loop-size of 100 m, receiver coil area of 14.14 m^2, and with a time range of 0.146-5.21 ms. All inversions results were interpreted using 105 and 80 parameters for models A and B, respectively.

An important issue related to the inversion results of the 1.5D TDEM data is to set the search space for the model parameters, that is, to choose properly the upper and lower bounds in which GOMs will perform the search. To overcome this problem, we have performed 1D independent inversions of TEM soundings, to provide the best range of search space. The Lagrange multipliers λ_ρ and λ_h in equation (2) are also an important issue in the inversion. Most common approaches to determinate these values are evaluated through the Pareto efficiency (i.e., to choose a pair of values around the corner closer to the origin), following line searches (e.g., Rawlinson et al 2006; Barboza et al 2018).

Nonetheless, this technique is quite influenced by the used model and the normalization values introduced in equations (3-5). For this reason, we opted the trial and error method. Thus, after executing several individual simulations was established for both synthetic models, a value of 0.01 for both multipliers (λ_r and λ_h) and $\sigma_\rho = 400$, $\sigma_d = 0.5$, $\sigma_h = 0.5$ for the normalization values mentioned above.

Since the parallel versions of WOA, PSO, and GWO allow us to run the inversions efficiently, we carried out ten simulations. Similarly with Barboza et al (2019), it was chosen the mean value of those simulation outcomes as the final solution. To select the number of agents that make up the population in WOA, PSO or GWO, we first have used the strategies proposed by Engelbrecht (2007) and Juan et al (2010). Using these authors' theories, we tested using factors of 6, 8, 10, and 12 times, the number of model parameters to configure the number of agents in the population. We chose to use 1000 agents for the experiments with both data sets. In the following subsections, are the outcomes for the two synthetic models elaborated for testing the performance of the parallel WOA, PSO and GWO algorithms. In both cases, to assure that the termination criterion is attained ($N_i = 500$), we kept 5000 as the maximum range of iterations allowed.

6.0.1 Model A: Sedimentary Basin

Figure 2 is shown the inverted resistivity models using the SL_1 and SL_2 stabilizers for the parallelized versions of WOA, PSO, and GWO, respectively.

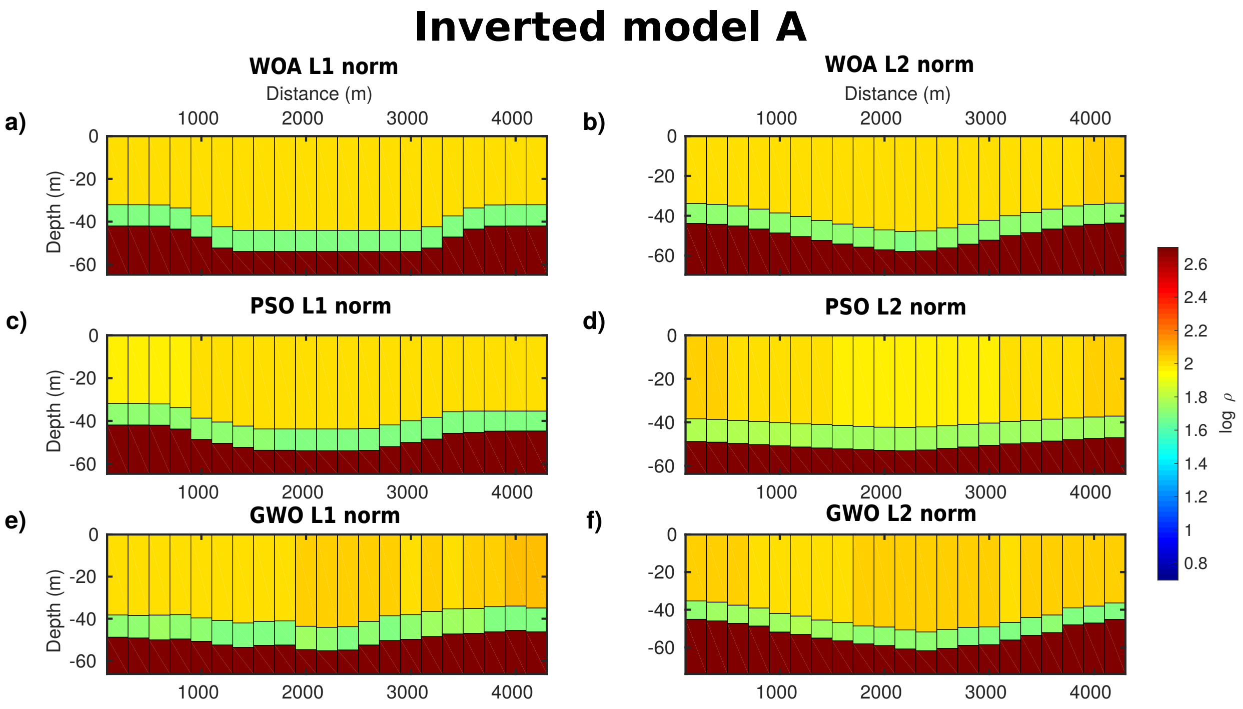

Figure 2: Model A. Parallelized inversion outcomes estimated using WOA in (a and b), PSO in (c and d), and GWO in (e and f). The left and right columns show inversion outcomes using L_1 and L_2 norm stabilizers respectively

Comparing the outcomes obtained with the true model depicted in Figure 1a, we concluded that all stabilizers reproduce quite well in the true model.

To check the quality of both the calculated data and the best model solution, we performed an analysis of cumulative distribution frequency on the relative errors (RE) (%) of the apparent resistivity data and inverted model parameters.

The study shows that almost the 85% of the inverted parameters model are reproduced with a value of RE=15% (Figure 3a) for the SL_1 stabilizer, whilst almost 85% of these inverted parameters are reproduced with a value of RE=10% (Figure 3b) for the SL_2 stabilizer.

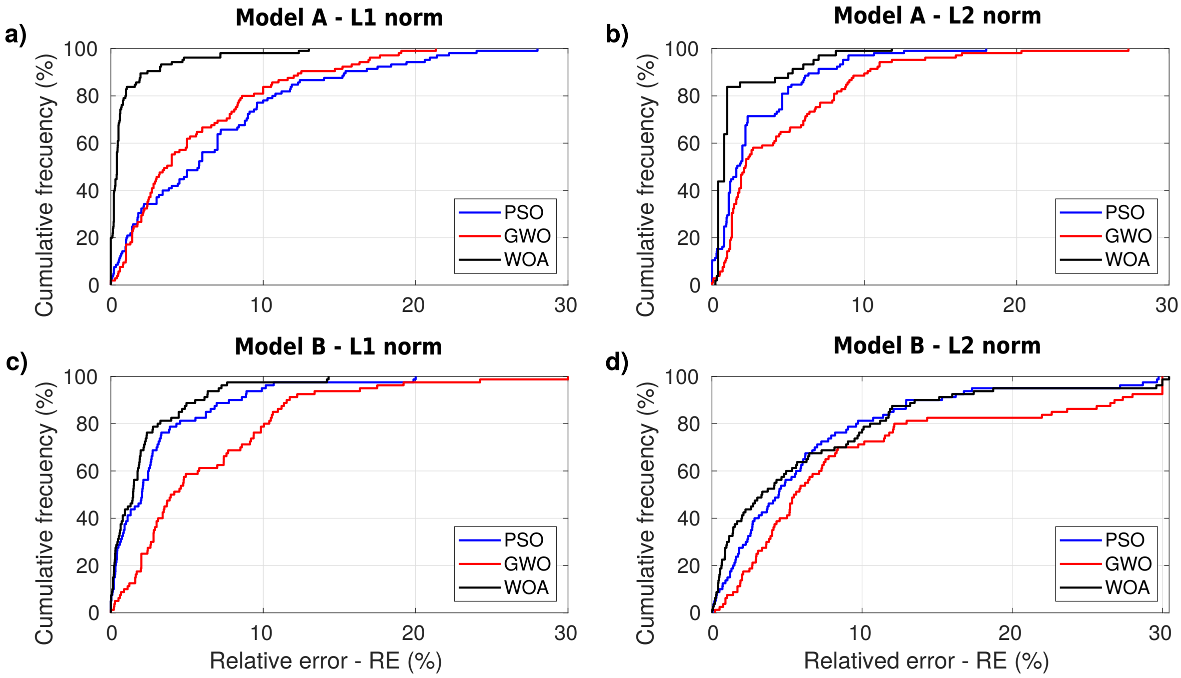

Figure 3: Cumulative distribution for the percentage discrepancy between true model A and the estimated model parameter values in (a and b) and for the model B in (c and d), obtained with WOA, PSO and GWO.The left and right panels are associated with the inversion results using the L_1 and L_2 norm, respectively. These outcomes were calculated using the best estimates shown in Figure 2 and Figure 6 for models A and B, respectively.

It is worth saying that in almost all cases, the reproduced data with both stabilizers present the worst performance with GWO than WOA and PSO algorithms. It is also possible to infer that almost 90% of apparent resistivity data achieved RE values lower to 10% for the SL_1 stabilizer (Figure 4a) and RE values less than 5% for the SL_2 stabilizer (Figure 4b). Notably, the SL_2 stabilizer achieves better results due to the smooth nature of the model.

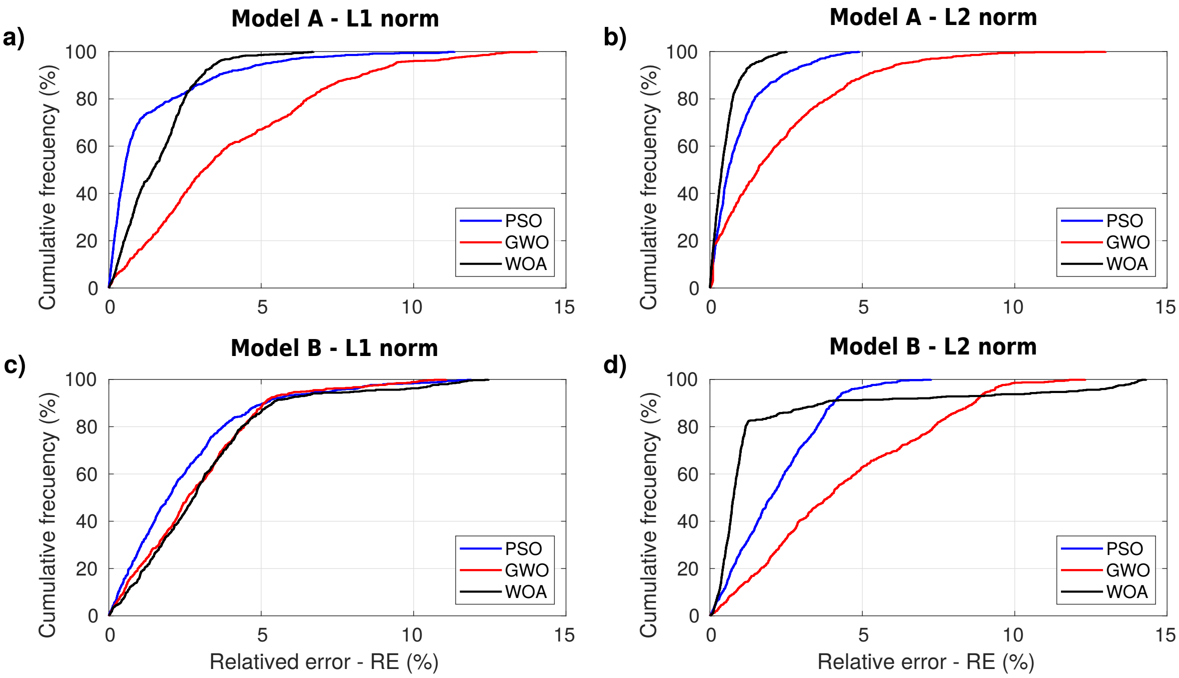

Figure 4: Cumulative distribution for the percentage discrepancy between observed and modeled apparent resistivity data for model A (a and b), and for model B (c and d), obtained with WOA, PSO and GWO. The left and right panels are associated with the inversion results using the L_1 and L_2 norm, respectively. These outcomes were calculated using the best estimates shown in Figure 2 and Figure 6 for models A and B, respectively.

Additionally, Figure 5 shows the evolution curves of the cost function for the parallelized versions of WOA, PSO, and GWO, using both stabilizers (for one of the ten inversions). Looking 5, we can infer that the GWO required more iterations to converge than the WOA and PSO. Also, we can conclude that the SL_2 stabilizer needed fewer iterations to satisfy the termination criterion defined in equation (20).

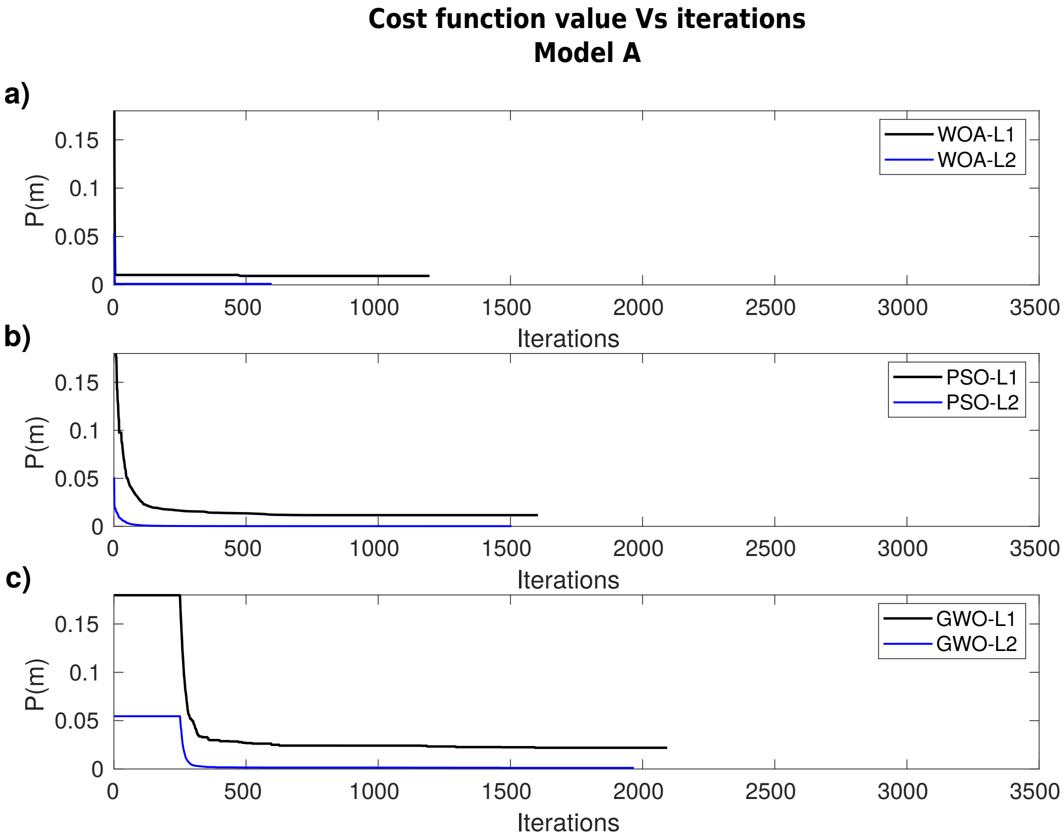

Figure 5: Evolutions of the cost function value along the iterations for the model A, using L_1 and L_2 norm stabilizers for the (a) WOA, (b) PSO and (c) GWO methods associated with the respective inversion estimates shown in Figure 2.

6.0.2 Model B: Buried Valley

Similarly to model A, Figure 6 depicts the inverted resistivity models using the $SL1$ and $SL2$ stabilizers for the parallelized versions of WOA, PSO and GWO. Both stabilizers are good approximations of the true model shown in Figure 1b.

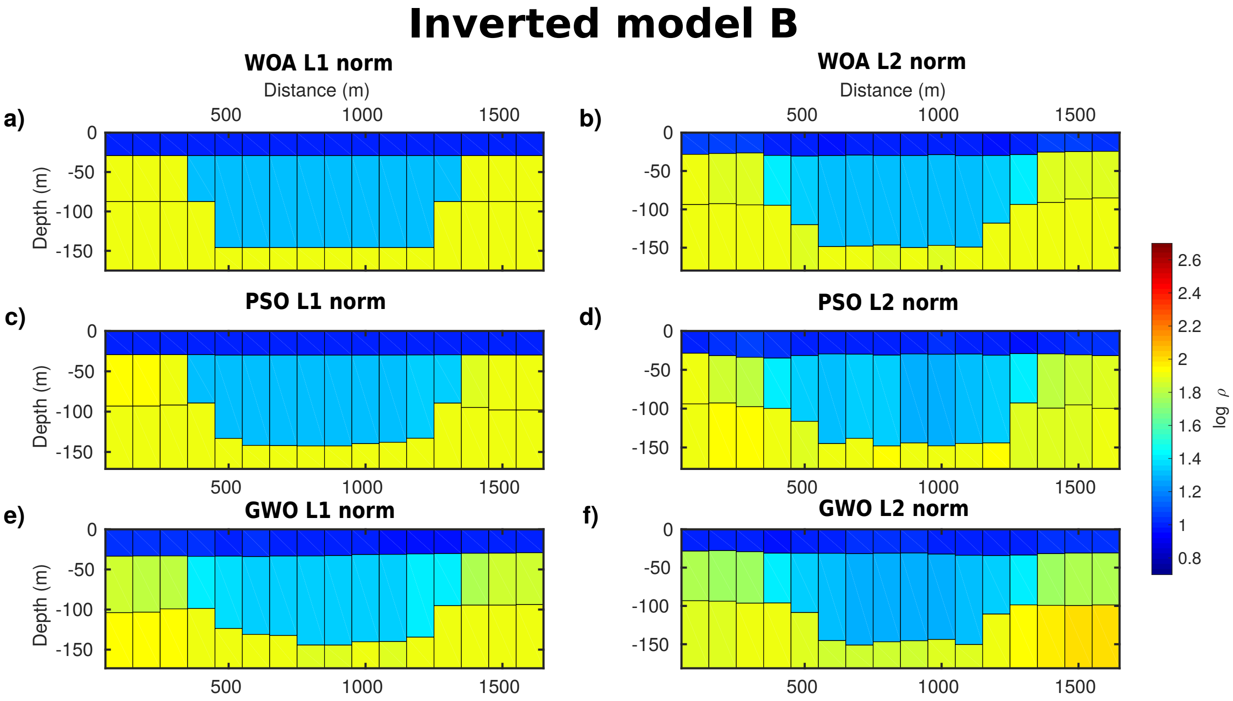

Figure 6: Model B. Parallelized inversion outcomes estimated using WOA in (a and b), PSO in (c and d), and GWO in (e and f). The left and right columns show inversion outcomes using L_1 and L_2 norm stabilizers, respectively.

As in Model A, we calculated the RE value for apparent data and inverted model parameters to validate these outcomes. The outcomes reflect that almost 85% of inverted model parameters are recovered with a value of RE=10% for the SL_1 stabilizer (Figure 3c), while 85% of data are reproduced with lower values of RE=15% for the SL_2 stabilizer (Figure 3d). It is also noted that almost 85% of the apparent resistivity data are reproduced with RE (%) less than 5% for the SL_1 stabilizer (Figure 4c), while 85% of these parameters are reproduced with RE values lower to 10% for the SL_2 stabilizer (Figure 4d). These results conclude that the SL_1 stabilizer favors solutions in which the physical property of interest may vary hugely.

Figure 7 depicts the evolution curves of cost function values. It is also inferred that the best performances are achieved with WOA, and PSO, while the GWO needs a higher number of iterations to achieve the stopping criterion defined in equation (20).

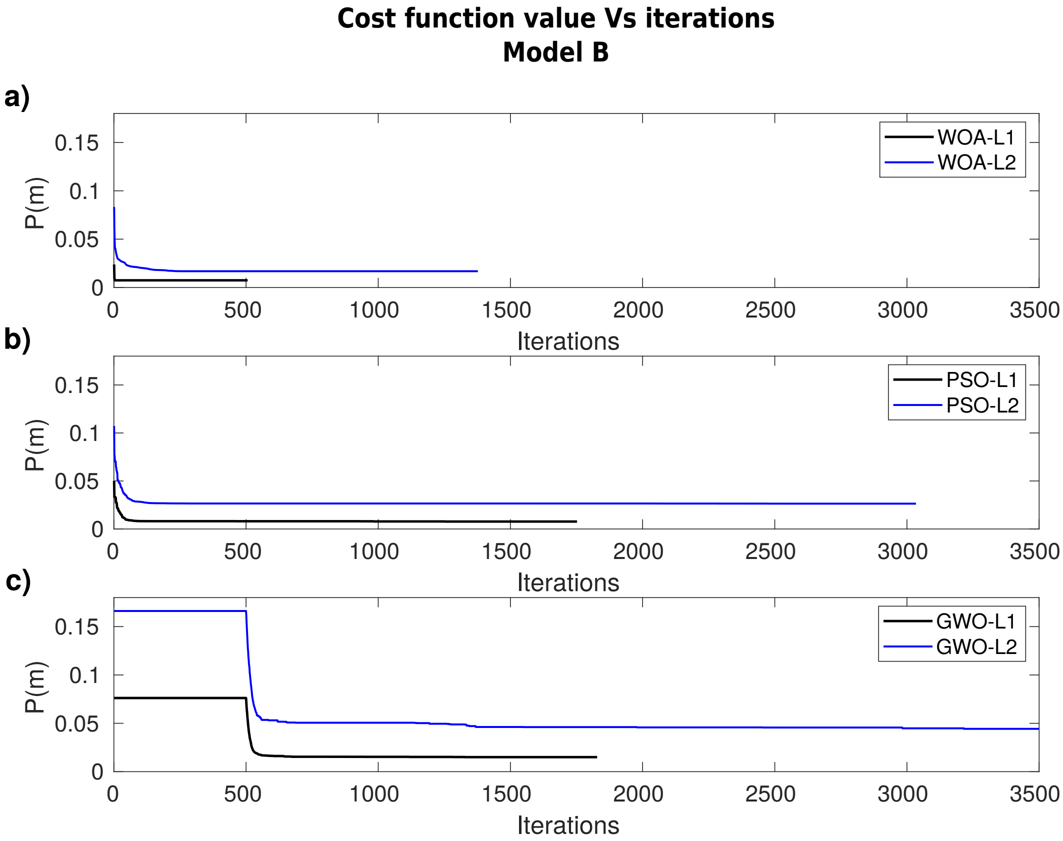

Figure 7: Evolutions of the cost function value along the iterations using L_1 and L_2 norm stabilizers for the (a) WOA, (b) PSO and (c) GWO methods associated with the respective inversion estimates shown in Figure 6.

7 Field data

We used experimental data of TDEM soundings, from a survey carried out in 2010 in the city of Bebedouro, São Paulo, State, in Brazil (Bortolozo et al, 2014). From this survey, we chose eight TDEM soundings that are approximately aligned forming a transverse (Figure 8a).

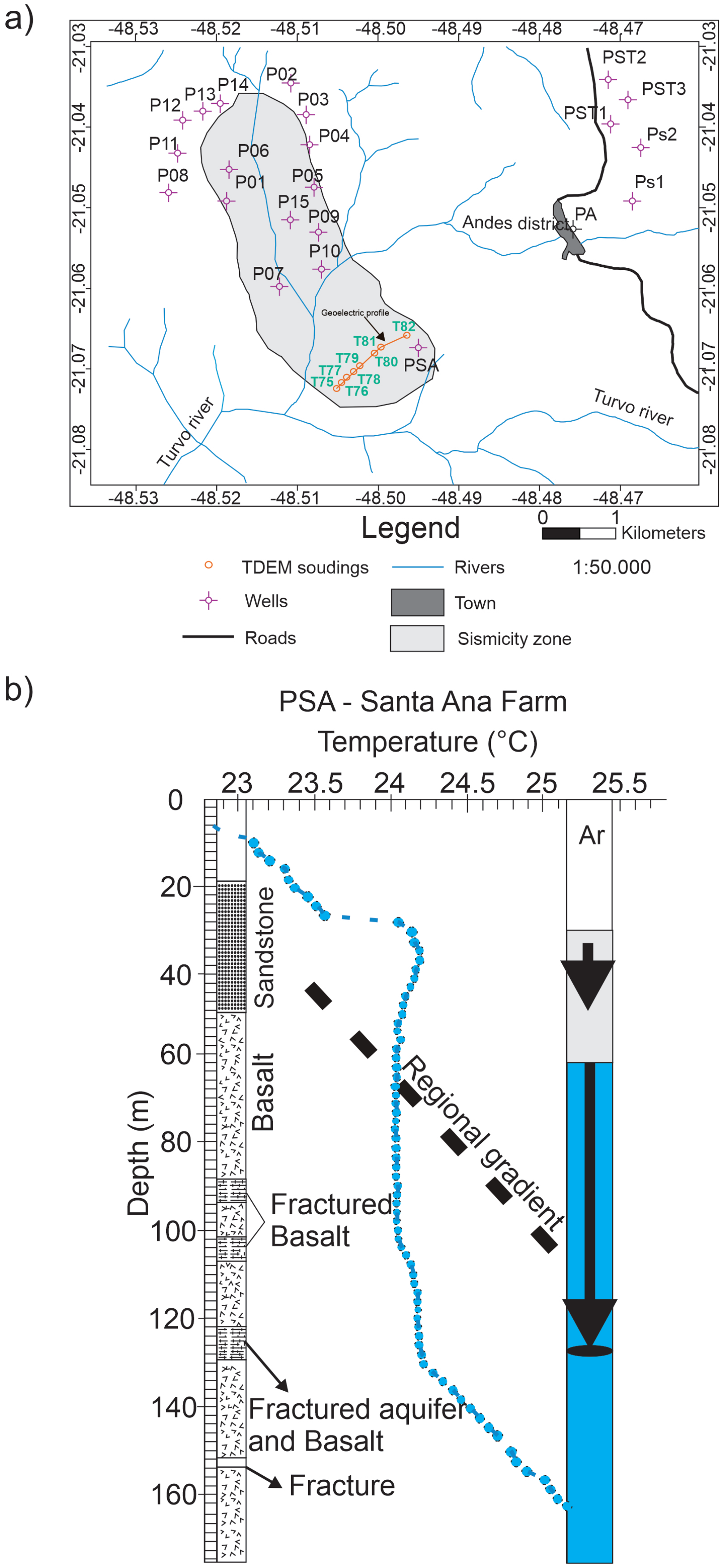

Figure 8: (a) TDEM soundings surveyed in the Bebedouro city. In orange is the geolectric profile containing the 8 soundings (T75-T82) used in this work. Adapted from Bortolozo et al (2014). (b) Thermal well PSA located in the Santa Ana Farm, near to line of TDEM soundings. The depth of the fractured basalt is shown from 50 m approximately. Adapted from Assumpção et al (2010).

These soundings were acquired using a square central loop array with 100 × 100 m, and within a time range of 0.088 to 33.73 ms. According to Porsani et al (2012), this survey was done for better understanding the hydrogeologic settings of fractured basaltic and sedimentary aquifers in the region. It is worth to mention that there are three principal geological formations in this area: (i) an upper layer that is Adamantina Formation, with approximate thickness that varies from 50 to 100 m, (ii) an intermediate layer made up by the Serra Geral Formation, composed by fractured basalt, with approximate thicknesses that varies from 400 to 600 m (Giampá and de Souza, 1982), and (iii) the bottom layer that corresponds to the Botucatu Formation, which involves the Guarani Aquifer.

In Figure 8b is shown the PSA borehole, located near to the transverse of TDEM soundings. This well exhibits a sedimentary layer that comprises the Adamantina Formation, with an approximate thickness of 50 m. Under the sedimentary layer we have the basalts from the Serra Geral Formation, that presents fractures in several intervals as depicted in Figure 8b. These records, allow us to support the geological settings above mentioned.

The interpretations results of Bortolozo et al (2014) to the transverse TDEM soundings are depicted in Figures 9a and 10a. We use an interpretation model of 4 layers according to the results obtained by Bortolozo et al (2014), and thus, the inversion outcomes were interpreted using 56 parameters. Additionally, the same analysis used to configure the Lagrange multipliers and the normalization factors in synthetic data, were used to configure the experimental data. We chose a value of 0.01 for both Lagrange multipliers (λ_r and λ_h), and $\sigma_\rho = 100$, $\sigma_d = 0.8$, and $\sigma_h = 0.5$, as the configuration parameters obtained after executing 10 previous inversions.

Figures 9 and 10 show the inversion results obtained using SL_1 and SL_2 stabilizers with WOA, PSO and GWO, respectively.

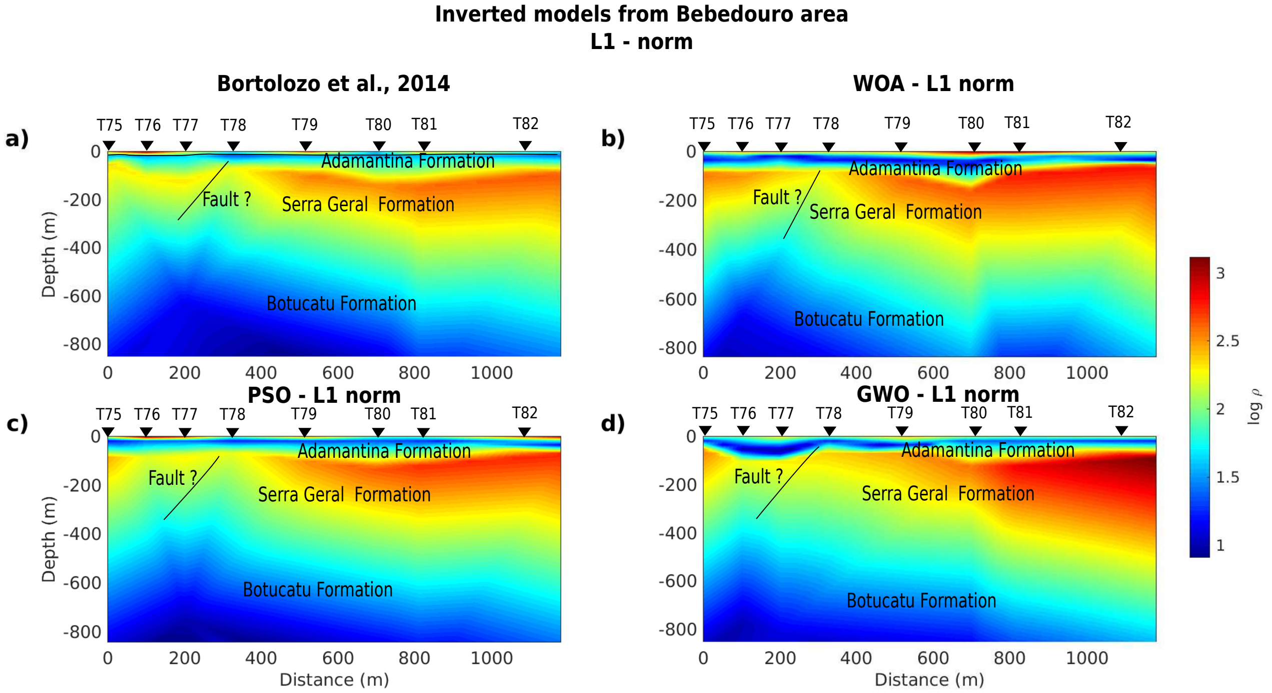

Figure 9: Field data. (a) Solution of Bortolozo et al 2014. (b-d) The inversion obtained with WOA, PSO and GWO, respectively, using L_1 norm stabilizer. The color bar that represents the resistivity value is the same in all figures.

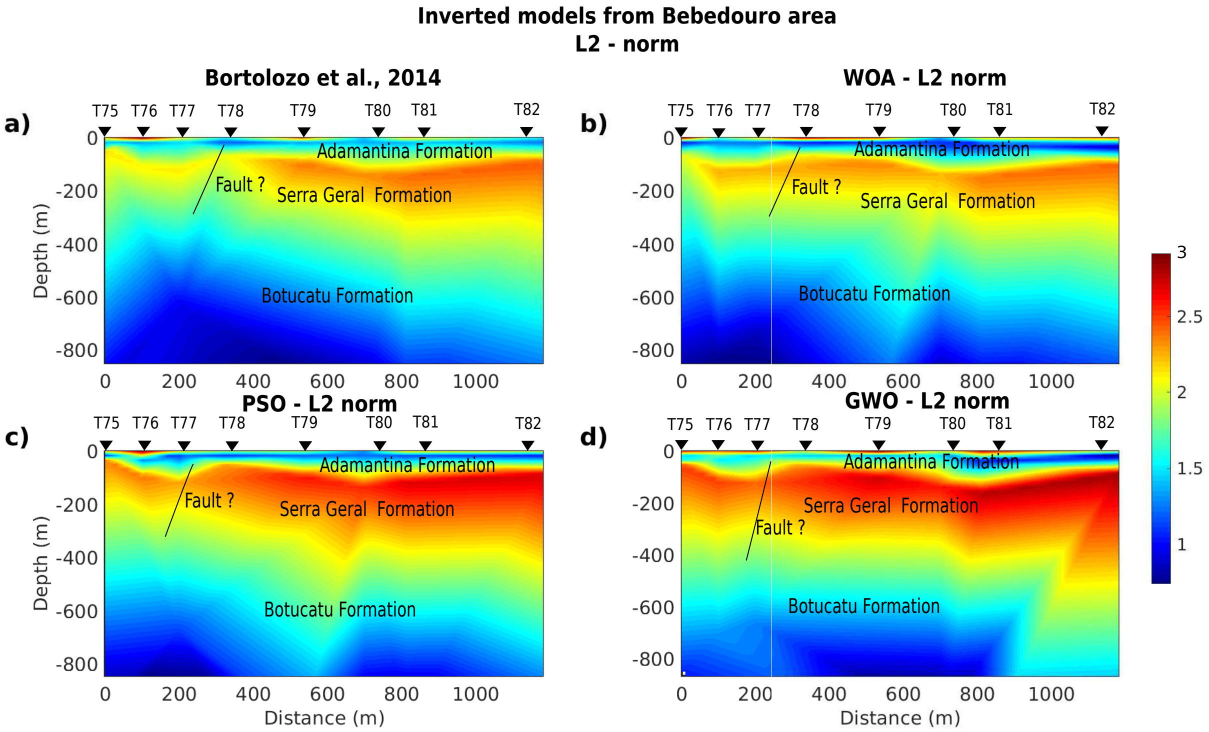

Figure 10: Field data. (a) Solution of Bortolozo et al 2014. (b-d) The inversion of obtained with WOA, PSO and GWO, respectively, using L_2 norm stabilizer. The color bar that represents the resistivity value is the same in all figures.

Both stabilizers reproduce quite well the solution of Bortolozo et al (2014) with small differences. The principal discrepancy occurs between T77 and T78 sounding. It seems that a fault is affecting the basalt layer of the Serra Geral Formation (yellow to orange color) at a depth of 100 m. These vertical displacements in the upper basalt layer, were also described in detail by (Almeida et al, 2017). It is also noted in all inversion results, that the SL_1 stabilizer identified this abrupt discontinuity better than the SL_2 stabilizer.

To evaluate the quality of the inversion estimates obtained with both stabilizers, we performed statistical analysis of cumulative frecuency distribution of the relative errors of all inversion results. These outcomes showed that almost 95% of the apparent resistivity data are reproduced with values less to

RE=20% for SL_1 stabilizer (Figure 11a), whilst 95% of the data are reproduced with RE lower than 30% for SL_2 stabilizer (Figure 11b). It is also noted that the WOA and PSO exhibit the better performances than the GWO.

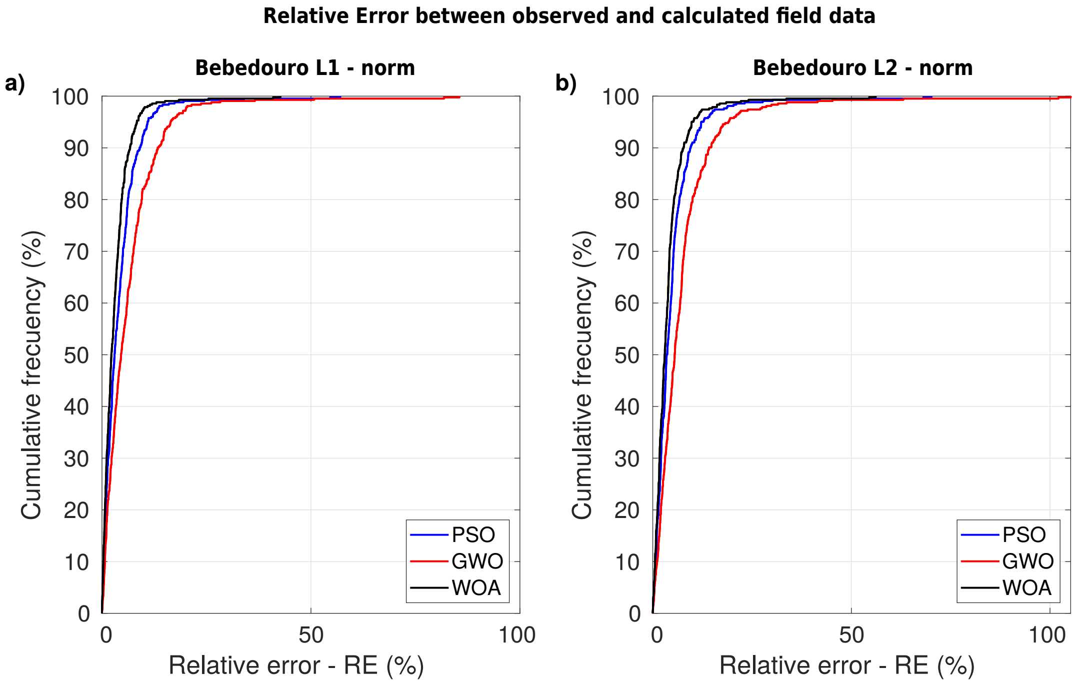

Figure 11: Cumulative distribution for the percentage discrepancy between observed and modeled apparent resistivity data for Bebedouro area with the three methods, using L_1 norm stabilizer (a), and (b) using L_2 norm stabilizer. These outcomes were calculated using the best inverted models obtained for PSO, GWO and WOA, respectively.

This last information is also validated in Figure 12, where the WOA needed fewer iterations to achieve the termination criterion than the PSO and GWO algorithms. It is also noted that the SL_1 stabilizer in all cases exhibited better results than the SL_2 stabilizer.

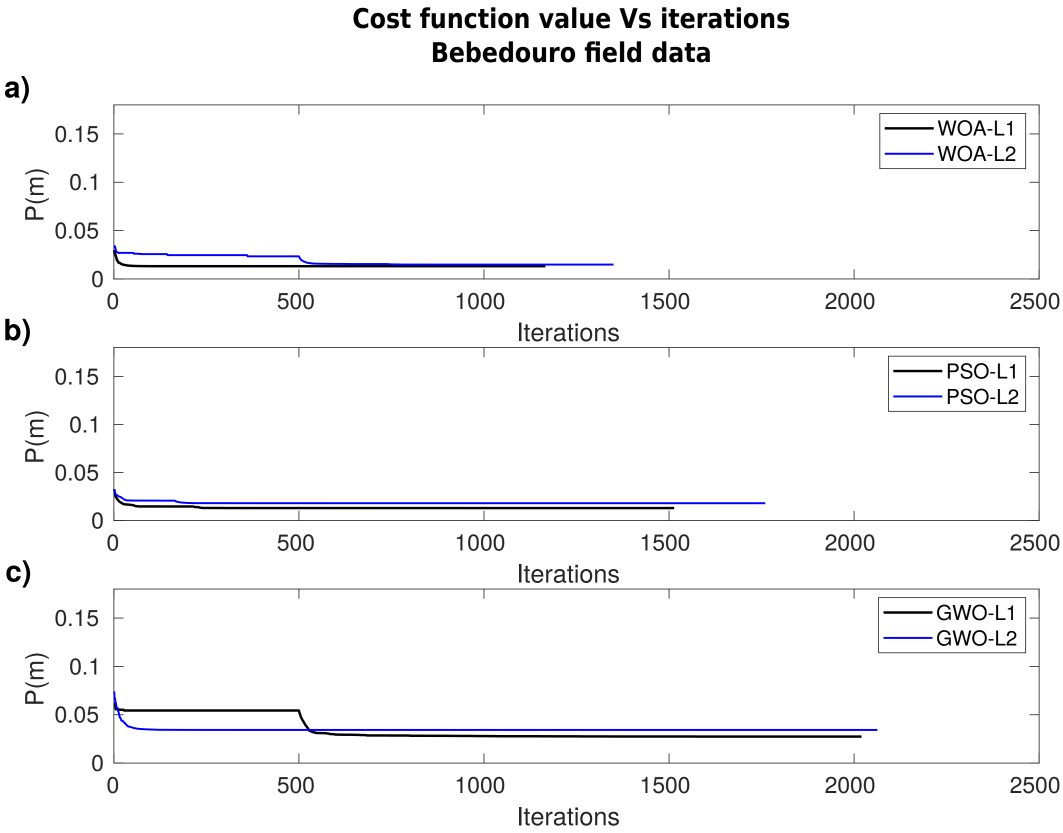

Figure 12: Field data. Evolution of the cost function value along the iterations using L_1 and L_2 norm in the lateral continuity constraints for the (a) WOA, (b) PSO and (c) GWO methods associated with the respective inversion estimates shown in Figures 9 and 10.

8 Discussions

GOMs offer high flexibility to incorporate several kinds of stabilizers once they are implemented computationally. This vantage offers an easy way for the interpreter to establish possible characteristics of the electrical resistivity distributions on the subsurface. In this work, we compare two lateral continuity constraints such as L_1 and L_2 norm on the model parameters to determine localized discontinuities such as sharp structures and faults. These constraints were incorporated easily in classical versions of WOA, PSO, and GWO. However, these techniques require a high computational cost associated with the vast quantity of cost function evaluations (Sen et al, 1993). As a manner of overcoming this disadvantage, several approaches adopting parallel architectures and programming languages have been used in the last years (e.g., Lalwani et al (2019)).

Such parallel techniques allowed us to reduce the inversion time enormously. Table 1 presents a quantitative distribution of such reduction in time, taking as a pattern the serial processing time.

Model	Algorithm	Parallel processing (MPI)		Sequential processing	
		L1 norm	L2 norm	L1 norm	L2 norm
Model A	WOA	2.14 hr	1.06 hr	36.8 hr	18.44 hr
	PSO	3.8 hr	3.03 hr	68.9 hr	52.8 hr
	GWO	6.5 hr	3.64 hr	115.3 hr	64.8 hr
Model B	WOA	0.57 hr	1.58 hr	9.92 hr	27 hr
	PSO	2.06 hr	3.6 hr	35.13	62.53 hr
	GWO	2.08 hr	4.2 hr	36.1 hr	71.42 hr
Bebedouro	WOA	0.32 hr	0.48 hr	16.19 hr	24.4 hr
	PSO	0.42 hr	0.57 hr	21.02 hr	28.65 hr
	GWO	0.5 hr	0.63 hr	25.06 hr	31.93 hr

Table 1: Total execution time comparison between parallel (MPI) and sequential inversions.

We run the inversions with synthetic models and real data for the performance tests. We tested all parallelized versions of WOA, PSO, and GWO on a high-performance cluster for scientific computing. We use 32 MPI processes per node, which provides the best performance, reducing the computational time significantly. The pure MPI version achieved a 17x speedup (quotient between the sequential time execution and parallel execution time) for both synthetic models and a 50x speedup for the real data.

In summary, synthetic model B and real data had better performances with the SL_1 stabilizer, which is very useful for determining sharp structures and faults. Figure 9 is a reliable example of that, where the inverted model parameters reproduced a well-fractured basalt layer below (T77-T78) soundings. In the same way, Figure 6 proved to be efficient in localizing sharp contacts and alternation of materials resulting from buried valleys. These evidences support the fact that SL_1 stabilizer is more sensitive to the model, principally to high-low variations of resistivity (Menke, 2018).

9 Conclusions

Our research compared the performances of three GOMs to solve 1.5D TEM inverse problem using synthetic and real data. These techniques offer flexibility in incorporating different stabilizers in the cost function but present some drawbacks, such as rapid entrapment in global minimum and high computational cost. To mitigate these disadvantages, we made a parallelization for the classical versions of WOA, PSO, and GWO. This implementation is based on the master/worker paradigm of parallel processing, in which the master process performs the WOA, PSO, and GWO related operations, and the worker processes complete the evaluation of the agents. The pure MPI version demonstrated to be highly effective in reducing computational time for all simulations. We validated the parallelized versions of WOA, PSO, and GWO in both synthetic models and the experimental data, using statistical analysis of cumulative distribution frequency of relative errors for apparent resistivity data and inverted model parameters. Both solutions provided reliable results and appeared to be an excellent alternative to the classical inversion techniques of time electromagnetic data.

10 CONFLICT OF INTEREST

The authors declares that they have no conflict of interest by a third party.

11 ACKNOWLEDGMENTS

Jorge Abril acknowledges to the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001 for his PhD. scholarship. We also thank to the Federal University's High Performance Processing Nucleus (NPAD) Rio Grande do Norte (UFRN) for the provision of computational resources for this research.

References

Abdelazeem M, Gobashy M, Khalil MH, et al (2019) A complete model parameter optimization from self-potential data using whale algorithm. Journal of Applied Geophysics 170:103,825

Agarwal A, Chandra A, Shalivahan S, et al (2018) Grey wolf optimizer: a new strategy to invert geophysical data sets. Geophysical Prospecting 66(6):1215–1226

Almeida ER, Porsani JL, dos Santos FAM, et al (2017) 2D TEM modeling for a hydrogeological study in the Paraná Sedimentary Basin, Brazil. International Journal of Geosciences 8(5):693–710

Anderson WL (1982) Calculation of transient soundings for a coincident loop system (program tcoloop). Tech. rep., US Geological Survey,

Assumpção M, Yamabe TH, Barbosa JR, et al (2010) Seismic activity triggered by water wells in the paraná basin, brazil. Water Resources Research 46(7)

Auken E, Christiansen AV (2004) Layered and laterally constrained 2D inversion of resistivity data. Geophysics 69(3):752–761

Auken E, Christiansen AV, Jacobsen BH, et al (2005) Piecewise 1D laterally constrained inversion of resistivity data. Geophysical Prospecting 53(4):497–506

Auken E, Christiansen AV, Jacobsen LH, et al (2008) A resolution study of buried valleys using laterally constrained inversion of tem data. Journal of Applied Geophysics 65(1):10–20

Barboza FM, Medeiros WE, Santana JM (2018) Customizing constraint incorporation in direct current resistivity inverse problems: A comparison among three global optimization methods. Geophysics 83(6):E409–E422

Barboza FM, Medeiros WE, Santana JM (2019) A user-driven feedback approach for 2d direct current resistivity inversion based on particle swarm optimization. Geophysics 84(2):E105–E124

Bertete-Aguirre H, Cherkaev E, Oristaglio M (2002) Non-smooth gravity problem with total variation penalization functional. Geophysical Journal International 149(2):499–507

Bortolozo CA, Couto Jr MA, Porsani JL, et al (2014) Geoelectrical characterization using joint inversion of VES/TEM data: A case study in paraná sedimentary basin, São Paulo State, Brazil. Journal of Applied Geophysics 111:33–46

Chandra A, Agarwal A, Singh RK (2017) Grey wolf optimisation for inversion of layered earth geophysical datasets. Near Surface Geophysics 15(5):499–513

Chen W, Hong H, Panahi M, et al (2019) Spatial prediction of landslide susceptibility using gis-based data mining techniques of anfis with whale optimization algorithm (WOA) and grey wolf optimizer (GWO). Applied Sciences 9(18):3755

Cheng J, Li F, Peng S, et al (2015) Joint inversion of TEM and DC in roadway advanced detection based on particle swarm optimization. Journal of Applied geophysics 123:30–35

Clerc M (1999) The swarm and the queen: towards a deterministic and adaptive particle swarm optimization. In: Proceedings of the 1999 congress on evolutionary computation-CEC99 (Cat. No. 99TH8406), IEEE, pp 1951–1957

Cockett R, Kang S, Heagy LJ, et al (2015) Simpeg: An open source framework for simulation and gradient based parameter estimation in geophysical applications. Computers & Geosciences 85:142–154

Constable SC, Parker RL, Constable CG (1987) Occam's inversion: A practical algorithm for generating smooth models from electromagnetic sounding data. Geophysics 52(3):289–300

Engelbrecht AP (2007) Computational intelligence: an introduction. John Wiley & Sons

Giampá CEQ, de Souza JC (1982) Potencial aquífero dos basaltos da formação Serra Geral no estado de São Paulo. Águas Subterrâneas

Gill PE, Murray W, Wright MH (2019) Practical optimization. SIAM

Godio A, Santilano A (2018) On the optimization of electromagnetic geophysical data: Application of the PSO algorithm. Journal of Applied Geophysics 148:163–174

Goldberg D (1989) Genetic algorithms in search, optimization, and machine learning. boston, ma: Addison-wesley

Hansen P (1992) Analysis of discrete ill-posed problems by means of the l-curve: Siam

DeGroot Hedlin C, Constable S (1990) Occam's inversion to generate smooth, two-dimensional models from magnetotelluric data. Geophysics 55(12):1613–1624

Juan L, Esperanza G, José P, et al (2010) PSO: a powerful algorithm to solve geophysical inverse problems: application to a 1D-DC resistivity case. J Appl Geophys 71:13–25

Kennedy J (2006) Swarm intelligence. In: Handbook of nature-inspired and innovative computing. Springer, p 187–219

Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of ICNN'95-international conference on neural networks, IEEE, pp 1942–1948

Kirkpatrick S, Gelatt CD, Vecchi MP (1983) Optimization by simulated annealing. Science 220(4598):671–680

Lalwani S, Sharma H, Satapathy SC, et al (2019) A survey on parallel particle swarm optimization algorithms. Arabian Journal for Science and Engineering 44(4):2899–2923

Lee CY, Zhuo GL (2021) A hybrid whale optimization algorithm for global optimization. Mathematics 9(13):1477

Li SY, Wang SM, Wang PF, et al (2018) An improved grey wolf optimizer algorithm for the inversion of geoelectrical data. Acta Geophysica 66(4):607–621

Liang X, Xu S, Liu Y, et al (2022) A modified whale optimization algorithm and its application in seismic inversion problem. Mobile Information Systems 2022

Loke MH, Acworth I, Dahlin T (2003) A comparison of smooth and blocky inversion methods in 2D electrical imaging surveys. Exploration geophysics 34(3):182–187

Menke W (2018) Geophysical data analysis: Discrete inverse theory. Academic press

Metcalfe TS, Charbonneau P (2003) Stellar structure modeling using a parallel genetic algorithm for objective global optimization. Journal of Computational Physics 185(1):176–193

Mirjalili S, Lewis A (2016) The whale optimization algorithm. Advances in engineering software 95:51–67

Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. Advances in engineering software 69:46–61

Monteiro FAS, El-Kaliouby HM (2010) Comparative study of local versus global methods for 1D joint inversion of direct current resistivity and time-domain electromagnetic data. Near Surface Geophysics 8(2):135–143

Muro C, Escobedo R, Spector L, et al (2011) Wolf-pack (canis lupus) hunting strategies emerge from simple rules in computational simulations. Behavioural processes 88(3):192–197

Porsani JL, Almeida ER, Bortolozo CA, et al (2012) TDEM survey in an area of seismicity induced by water wells in Paraná sedimentary basin, northern São Paulo State, Brazil. Journal of Applied Geophysics 82:75–83

Rawlinson N, Reading AM, Kennett BL (2006) Lithospheric structure of tasmania from a novel form of teleseismic tomography. Journal of Geophysical Research: Solid Earth 111(B2)

Rudin LI, Osher S, Fatemi E (1992) Nonlinear total variation based noise removal algorithms. Physica D: nonlinear phenomena 60(1-4):259–268

Sandberg SK (1988) Microcomputer software for the processing and forward modeling of transient electromagnetic data taken in the central loop sounding configuration. New Jersey Department of Environmental Protection, Division of Water

Santos FAM (2004) 1D laterally constrained inversion of em34 profiling data. Journal of Applied Geophysics 56(2):123–134

Sen MK, Stoffa PL (1995) Global optimization methods in geophysical inversion. Cambridge University Press

Sen MK, Stoffa PL (2013) Global optimization methods in geophysical inversion. Cambridge University Press

Sen MK, Bhattacharya BB, Stoffa PL (1993) Nonlinear inversion of resistivity sounding data. Geophysics 58(4):496–507

Silva JB, Medeiros WE, Barbosa VC (2001) Potential-field inversion: Choosing the appropriate technique to solve a geologic problem. Geophysics 66(2):511–520

4

Conclusões

Os métodos de otimização global se mostraram ser poderosas ferramentas na inversão bidimensional de dados elétricos e 1,5D de dados eletromagnéticos. No entanto, os métodos de otimização global exigem o uso de técnicas de paralelismo para acelerar o tempo de computação, especialmente quando se trabalha com muitos parâmetros, ou se faz uso de modelagens computacionalmente custosas. Além disso, o uso de estabilizadores no processo da inversão permite obter melhores resultados da distribuição da resistividade em subsuperfície, quando se tem estruturas que apresentam fortes heterogeneidades laterais ou falhas.

Para o caso do problema inverso de resistividade 2D, a técnica de otimização de enxame de partículas com elitismo e mutação (EMPSO) permitiu obter bons resultados tanto nos modelos sintéticos quanto nos dados reais, usando estabilizadores de Variação Total (VT) e Suavidade Global (SG). Além disso, a técnica de paralelismo MPI puro tornou-se mais eficiente que a técnica híbrida (MPI+ OpenMP), permitindo acelerar o tempo de processamento 17x em relação ao tempo de processamento sequêncial. Ambos os estabilizadores (VT e SG) apresentaram bom desempenho, sendo o vínculo de Variação Total o que apresentou melhores taxas de convergência da função custo. Estes resultados também são refletidos nas curvas da frequência acumulada do erro relativo para os modelos invertidos e os dados observados.

Por outro lado, no caso do problema inverso electromagnético 1,5D, o uso das versões paralelizadas canônicas do WOA, PSO e GWO aplicando o MPI puro, permitiram acelerar o tempo de inversão 17 vezes nos modelos sintéticos e 50 vezes nos dados reais, em relação o aos tempos de processamento sequencial. Além disso, o estabilizador norma L_1 foi mais eficaz para mapear altos contrastes de resistividade que o estabilizador norma L_2. Estes resultados são refletidos nas curvas do erro relativo para os modelos invertidos e os dados obervados. O desempenho dos algoritmos WOA, PSO e GWO apresentados nas curvas de evolução da função objetivo ao longo das iteracções permite reforçar estas informações.

Agradecimentos

Agradeço a Deus pela vida e pela força que me dá todos os dias para cumprir meus objetivos.

Agradeção aos meus pais e meu irmão por seu incondicional amor e apoio ao longo deste trabalho e da minha vida.

Agradeço a CAPES pela bolsa de doutorado fornecida para o desenvolvimento desta pesquisa e ao Brasil maravilhoso que me acolheu durante meus estudos.

Agradeço ao Nucleo de Processamento de Alto Desempenho (NPAD) da UFRN pelo uso supercomputador, peça fundamental para o desenvolvimento desta pesquisa.

Agradeço ao engenheiros Danilo Chaves e Haroldo Watson do NPAD pela suporte com o uso do supercomputador.

Agradeço ao meu orientador Marcos Vasconcelos.

Agradeço ao meu co-orientador Márcio Barboza pelas úteis conversações para melhorar a pesquisa.

Agradeço ao Dr Mojica pela seu apoio na escrita do primeiro artigo e seu apoio no processamento paralelo.

Agradeço ao Dr Bortolozo pelo apoio na pesquisa do segundo artigo.

Agradeço aos meus professores da Pos-graduação por seus ensinamentos acadêmicos e profissionais (Prof. Dr. Milton Porsani, Amin Bassrei, Olivar Lima).

Finalmente agradeço a todos os direta e indiretamente envolvidos com este trabalho.

Apêndice A

Pseudocódigos do WOA, GWO e PSO - caso 1.5D

A seguer são definidos os três pseudocódigos usados para a resolução do problema inverso 1,5D de dados eletromagnéticos no domínio do tempo. O código WOA foi adaptado de Mirjalili e Lewis (2016), o GWO adaptado de Mirjalili et al. (2014) e o PSO adaptado de Abril et al. (2022).

A.1 Pseudo-code Whale Optimization Algorithm (WOA)

```
pos(:,:)←initialize(n,nwhales,lb,ub)                    ▷ Initialize all whale's position
iter=1
while termination criterion not satisfied do                  ▷ Main optimization loop
    whalefit(:)←call_mpi_fitness(n,n_whales,pos)     ▷ Parallel cost evaluation with MPI
    leaderpos(:)←updateleader(whalefit,pos)                        ▷ Update the leader
    for j = 1 to n_whales do
        update a, A, C, l and p
        for i = 1 to n do
            if $p \leq 0.5$ then
                if $A \leq 1$ then
                    d_leader←C* leaderpos(i) - pos(i,j)     ▷ Update distance to leader
                    pos(:,j)←leaderpos(i)- A* d_leader                ▷ Update position
                else if $A \geq 1$ then
                    d_random←C*random_pos(i)-pos(i,j)        ▷ Update random distance
                    pos(i,j)←(leaderpos(i)-A*d_random         ▷ Update random position
                end if
            else if $p \geq 0.5$ then
                d_leader←leaderpos(i)- pos(i,j)             ▷ Update distance to leader
                pos(i,j)←d_leader* $e^{bl} cos(2\pi l)$ + leaderpos(i)  ▷ Update position
            end if
        end for
    end for
    iter= iter +1                                                ▷ Advance in iteration
end while
[1]
```

Figura A.1: A parallel pure MPI WOA pseudo-code. Adapted from Mirjalili and Lewis (2016).

A.2 Pseudo-code Particle Swarm Optimization (PSO)

```
p(:,:),v(:,:)←initialize(n,npart,lb,ub)  ▷ Initialize all particle's position and velocity
for j = 1 to npart do
    v(:,j)←call_velocity_boundary_treatment(n,vmax)  ▷ Constrains the components of velocity
end for
pBest(:,:)←particle(:,:)                            ▷ Initialize particle's best position
iter=1
while termination criterion not satisfied do                  ▷ Main optimization loop
    particlefit(:)←call_mpi_fitness(n,npart,particle)  ▷ Parallel cost evaluation with MPI
        ▷ Updates the values of pbest and pbestfit for all the swarm using syncronous strategy
    pbest(:,:), pbestfit(:),gbestfit,gbestpos←sync_pgbest(npart,pbestfit,pneigbours,minmax)
        ▷ Updates the best-known position for the neighbouring particles
    localbest_pos(:),localbest_fit(:)←updatelocalbest(npart,pbest_fit,p_neigbours,minmax)
    for j = 1 to npart do
            ▷ Calculate a new particle's velocity
        v(:,j)←compute_vel(n,particle(:,j),w,c1,c2,cf,pbest(:,j),pbest(:,localbestpos(j)),
                localbestpos(j))
            ▷ Constraints the components of the velocity
        v(:,j)←velocity_boundary_treatment(n,vmax)
            ▷ Updates the position of the particles and keeps them in the research space
        particle(:,j)←position_boundary_treatment(n,v(:,j),lb,ub,boundary_wall)
    end for
    iter= iter +1                                                ▷ Advance in iteration
end while
```

Figura A.2: A parallel pure MPI PSO pseudo-code. Adapted from Abril et al. (2022).

A.3 Pseudo-code Grey Wold Optimizer (GWO)

```
pos(:,:)←initialize(n,n_wolves,lb,ub)                  ▷ Initialize all wolves' position
Update a, A, C                                         ▷ Configure other parameters
iter=1
while termination criterion not satisfied do           ▷ Main optimization loop
    wolffit(:)←call_mpi_fitness(n,n_wolves,pos)        ▷ Parallel cost evaluation with MPI
    alphapos(:), betapos(:), deltapos(:)←update_alpha_beta_delta(wolffit,pos)    ▷ Update alpha, beta and delta positions
    for j = 1 to n_wolves do
        for i = 1 to n do
            d_alpha←C*alphapos(i)-pos(i,j)             ▷ Update alpha distance
            pos1←alphapos(i)-A*d_alpha                 ▷ Update position 1
            d_beta←C*betapos(i)-pos(i,j)               ▷ Update beta distance
            pos2←betapos(i)-A*d_beta                   ▷ Update position 2
            d_delta←C*deltapos(i)-pos(i,j)             ▷ Update delta distance
            pos3←deltapos(i)-A*d_delta                 ▷ Update position 3
            pos(i,j)←pos1 +pos2+ pos3 /3               ▷ Update position
        end for
    end for
    iter= iter +1                                      ▷ Advance in iteration
end while
```

[1]

Figura A.3: A parallel pure MPI GWO pseudo-code. Adapted from Mirjalili et al. (2014).

Referências Bibliográficas

Abril, J. L.; Vasconcelos, M. A.; Barboza, F. M. e Mojica, O. F. (2022) A parallel improved PSO algorithm with genetic operators for 2D inversion of resistivity data, Acta Geophysica, pp. 1–18.

Auken, E. e Christiansen, A. V. (2004) Layered and laterally constrained 2d inversion of resistivity data, Geophysics, **69**(3):752–761.

Auken, E.; Christiansen, A. V.; Jacobsen, B. H.; Foged, N. e Sørensen, K. I. (2005) Piecewise 1d laterally constrained inversion of resistivity data, Geophysical Prospecting, **53**(4):497–506.

Bortolozo, C. A. (2016) Inversão conjunta 1D e 2D de dados de Eletrorresistividade e TDEM aplicados em estudos de hidrogeologia na bacia do Paraná, Tese de Doutorado, Tese de Doutorado (Geofísica), IAG-USP, São Paulo.

Constable, S. C.; Parker, R. L. e Constable, C. G. (1987) Occam's inversion: A practical algorithm for generating smooth models from electromagnetic sounding data, Geophysics, **52**(3):289–300.

Dey, A. e Morrison, H. (1979) Resistivity modelling for arbitrarily shaped two-dimensional structures, Geophysical prospecting, **27**(1):106–136.

Dieter, K.; Paterson, N. e Grant, F. (1969) Ip and resistivity type curves for three-dimensional bodies, Geophysics, **34**(4):615–632.

Ellis, R. e Oldenburg, D. (1994) The pole-pole 3-d dc-resistivity inverse problem: a conjugategradient approach, Geophysical Journal International, **119**(1):187–194.

Harrington, R. F. (1961) Time-harmonic electromagnetic fields, McGraw-Hill College.

Hohmann, G. W. (1975) Three-dimensional induced polarization and electromagnetic modeling, Geophysics, **40**(2):309–324.

Kennedy, J. (2006) Swarm intelligence, In: Handbook of nature-inspired and innovative computing, pp. 187–219, Springer.

Kennedy, J. e Eberhart, R. (1995) Particle swarm optimization, In: Proceedings of ICNN'95-international conference on neural networks, vol. 4, pp. 1942–1948, IEEE.

Lalwani, S.; Sharma, H.; Satapathy, S. C.; Deep, K. e Bansal, J. C. (2019) A survey on parallel particle swarm optimization algorithms, Arabian Journal for Science and Engineering, **44**(4):2899–2923.

Lowry, T.; Allen, M. e Shive, P. N. (1989) Singularity removal: A refinement of resistivity modeling techniques, Geophysics, **54**(6):766–774.

Marescot, L.; Lopes, S. P.; Rigobert, S. e Green, A. G. (2008) Nonlinear inversion of geoelectric data acquired across 3d objects using a finite-element approach, Geophysics, **73**(3):F121–F133.

Metcalfe, T. S. e Charbonneau, P. (2003) Stellar structure modeling using a parallel genetic algorithm for objective global optimization, Journal of Computational Physics, **185**(1):176–193.

Mirjalili, S. e Lewis, A. (2016) The whale optimization algorithm, Advances in engineering software, **95**:51–67.

Mirjalili, S.; Mirjalili, S. M. e Lewis, A. (2014) Grey wolf optimizer, Advances in engineering software, **69**:46–61.

Mufti, I. R. (1976) Finite-difference resistivity modeling for arbitrarily shaped two-dimensional structures, Geophysics, **41**(1):62–78.

Muro, C.; Escobedo, R.; Spector, L. e Coppinger, R. (2011) Wolf-pack (canis lupus) hunting strategies emerge from simple rules in computational simulations, Behavioural processes, **88**(3):192–197.

Nagesh Kumar, D. e Janga Reddy, M. (2007) Multipurpose reservoir operation using particle swarm optimization, Journal of Water Resources Planning and Management, **133**(3):192–201.

Papadopoulos, N.; Tsourlos, P.; Tsokas, G. e Sarris, A. (2007) Efficient ert measuring and inversion strategies for 3d imaging of buried antiquities, Near Surface Geophysics, **5**(6):349–361.

Park, S. K. e Van, G. P. (1991) Inversion of pole-pole data for 3-d resistivity structure beneath arrays of electrodes, Geophysics, **56**(7):951–960.

Rudin, L. I.; Osher, S. e Fatemi, E. (1992) Nonlinear total variation based noise removal algorithms, Physica D: nonlinear phenomena, **60**(1-4):259–268.

Santos, F. A. M. (2004) 1-d laterally constrained inversion of em34 profiling data, Journal of Applied Geophysics, **56**(2):123–134.

Sasaki, Y. (1994) 3-d resistivity inversion using the finite-element method, Geophysics, **59**(12):1839–1848.

Spitzer, K. (1995) A 3-d finite-difference algorithm for dc resistivity modelling using conjugate gradient methods, Geophysical Journal International, **123**(3):903–914.

Vachiratienchai, C. (2007) Two-dimensional direct current resistivity inversion: finite difference method, Tese de Doutorado, Mahidol University.

Zhang, J.; Mackie, R. L. e Madden, T. R. (1995) 3-d resistivity forward modeling and inversion using conjugate gradients, Geophysics, **60**(5):1313–1325.