



Inversão da forma de onda completa usando diferenciação automática: uma alternativa ao método dos estados adjuntos

Felipe Louzeiro Azevedo* ^{1 2}, Jessé Carvalho Costa ^{1 2}, Bruno dos Santos Silva ^{1 2}, Natiê Almeida Albano ^{1 2},

¹ Universidade Federal do Pará (UFPA), ² Instituto Nacional de Ciência e Tecnologia de Geofísica do Petróleo (INCT-GP)

Copyright 2023, SBGf - Sociedade Brasileira de Geofísica

This paper was prepared for presentation during the 18th International Congress of the Geophysical Society held in Rio de Janeiro, Brazil, 16 to 19 October 2023.

Contents of this paper were reviewed by the Technical Committee of the 18th International Congress of the Brazilian Geophysical Society. Ideas and concepts of the text are authors' responsibility and do not necessarily represent any position of the SBGf, its officers or members. Electronic reproduction or storage of any part of this paper for commercial purposes without the written consent of the Brazilian Geophysical Society is prohibited.

Abstract

The derivation of the gradient of the objective function used in full waveform inversion traditionally employs the adjoint-state method. This approach needs to be manually implemented and verified for each specific problem. This takes human effort and can be prone to error when the forward problem involves higher computational complexity. To reduce human intervention of the gradient computation, automatic differentiation can be an attractive technique. Even if only for the validation of the human-derived gradient implementation. In this paper, we derive the gradient of the least-squares objective function for the inversion of scalar acoustic wavefield recordings using the adjoint-state method and automatic differentiation. These approaches are validated on the Marmousi dataset. The results show that automatic differentiation produces a gradient similar to the gradient obtained by the adjoint-state method, but with differences in computational cost and, mainly, storage demand.

Introdução

A inversão da forma de onda completa (FWI) visa recuperar os parâmetros físicos da subsuperfície da Terra ajustando o campo de onda sísmico medido na superfície (Tarantola, 1984). A solução desse problema envolve a minimização da função objetivo que normalmente é definida como a soma quadrática do erro entre o dado sísmico observado e o dado modelado. O dado modelado é obtido resolvendo a equação da onda completa tomando como parâmetro um modelo inicial das propriedades físicas da subsuperfície. Tradicionalmente, a derivação do gradiente da função objetivo usa o método dos estados adjuntos (ASM) (Plessix, 2006). Nesta metodologia, o gradiente precisa ser derivado manualmente para cada problema de modelagem, bem como para funções objetivos diferentes. Assim, a implementação da derivação do gradiente da função objetivo pode demandar tempo e está sujeito a erros, mesmo para um problema inverso simples, como a inversão do campo de onda acústico escalar. Nesse contexto, empregar uma abordagem de diferenciação automática (AD) (Elliott, 2018) para calcular o gradiente pode agilizar o tempo de desenvolvimento do projeto e fornecer uma avaliação inicial do gradiente da

função objetivo usada na inversão.

A diferenciação automática é um conjunto de técnicas que permite calcular derivadas de funções matemáticas de forma automatizada, fornecendo uma alternativa aos métodos numéricos tradicionais, como o método das diferenças finitas (FDM) (Klein et al., 2002). O pressuposto que a AD é um método de derivação exato (ou seja, dentro da precisão de máquina) só é válido se a função avaliada tem uma solução analítica. Todavia, se a função avaliada é resultante de um procedimento numérico, como a solução numérica de uma equação de diferenciação parcial, então, a derivada calculada pela AD pode não ser tão precisa (Nocedal and Wright, 2006). Por exemplo, a função objetivo utilizada na inversão do campo de onda sísmico requer a solução da equação da onda, que comumente é resolvida usando o método numérico FDM. Tal procedimento introduz um erro de aproximação que normalmente é pequeno. Esse é também propagado para a função objetivo. No entanto, mesmo que o erro de aproximação inserido na função objetivo seja pequeno, sua derivada pode não ser. Portanto, o gradiente gerado pela AD de uma função objetivo resultante de um procedimento numérico pode não ser mais exato.

A diferenciação automática é uma abordagem ainda muito recente em aplicações na FWI. Sambridge et al. (2007) mostrou que a AD tem potencial para calcular a sensibilidade de funcionais em uma variedade de problemas específicos de geociências, entretanto, é necessário avaliar seu desempenho em problemas de grande escala. Essa questão foi investigada por Zhu et al. (2021), que aplicou a AD para resolver a FWI para o dado do modelo Marmousi. O autor afirma que a AD produz resultados semelhantes ao ASM.

As ferramentas de AD atuais se tornaram muito eficientes para avaliar derivadas automaticamente (Griewank and Walther, 2000). No entanto, ainda é notório que para problemas de grande escala tais ferramentas devem contornar o alto custo computacional e principalmente a alta demanda de armazenamento exigidos na computação do gradiente. Neste trabalho usamos a ferramenta de AD Enzyme.jl (uma interface para o Enzyme) (Moses et al., 2021).

Existem outros desafios em usar AD para computação do gradiente, aqui ressaltamos a necessidade de estruturação do código de modelagem a depender das limitações de cada ferramenta de AD. Não obstante, a AD tem algumas vantagens, como: i) reduz consideravelmente o tempo de desenvolvimento em comparação ao exigido pelo ASM, que precisa ser implementado e verificado manualmente, ii) pode facilitar análise de outras funções

objetivos com pouco custo de implementação adicional e iii) pode ser usado para validar gradientes de funções objetivos de problemas inversos com alta complexidade de derivação matemática.

Neste trabalho, mostramos que a AD pode ser um método alternativo para validar o cálculo do gradiente da função objetivo usada na FWI. Para isso, apresentamos a formulação do problema inverso para a equação da onda acústica escalar. Adicionalmente, mostramos o cálculo do gradiente da função objetivo de quadrados mínimos usando o ASM e AD. Experimentos com o modelo de velocidade Marmousi mostram que tanto a AD quanto o ASM geram gradientes semelhantes, mas apresentam diferenças no custo computacional e principalmente na demanda de armazenamento. Os resultados das inversões com o método de otimização de gradiente conjugado não linear (NLCG) usando os dois gradientes atingiram precisão semelhantes.

Derivação do gradiente da função objetivo usando o Método dos Estados Adjuntos Discreto

Discretização da equação da onda acústica usando o FDM

A equação da onda acústica escalar com densidade constante para um meio bidimensional é dada por:

$$\partial_{tt} p(x, z, t) = c^2 \left[\nabla^2 p(x, z, t) + w(t) \delta(x - x_s) \delta(z - z_s) \right], \quad (1)$$

com as condições iniciais

$$p(x, z, t)|_{t \leq 0} = 0 \quad \text{e} \quad \partial_t p(x, z, t)|_{t \leq 0} = 0 \quad (2)$$

e condições de fronteiras

$$p(x, z, t)|_{\partial\Omega} = 0 \quad \text{ou} \quad \nabla p(x, z, t) \cdot \mathbf{n}|_{\partial\Omega} = 0, \quad (3)$$

onde $\partial\Omega$ é a fronteira do domínio, representado por Ω , e \mathbf{n} é o vetor normal unitário à superfície $\partial\Omega$.

Na equação (1), $p(x, z, t)$ representa o campo de onda de pressão, $c = c(x, z)$ é a velocidade de propagação da onda compressional, $w(t)$ é a função do sinal da fonte sísmica, $\delta(x - x_s)$ e $\delta(z - z_s)$ são deltas de Dirichlet e $\nabla^2 p(x, z, t)$ é o operador Laplaciano aplicado ao campo de pressão e é dado por:

$$\nabla^2 p(x, z, t) = \partial_{xx} p(x, z, t) + \partial_{zz} p(x, z, t), \quad (4)$$

que descreve as variações espaciais de segunda ordem do campo de pressão ao longo dos eixos espaciais x e z , respectivamente. A variável t representa o eixo de tempo e x_s e z_s formam a coordenada de aplicação fonte.

Para discretizar equação (1) usando o FDM, considere que uma coordenada no espaço contínuo (x, z) é representada por uma coordenada (x_i, z_j) no espaço discreto. Considere também que o tempo contínuo t seja representado no tempo discreto por t_n . Assim, o campo de pressão contínuo $p(x, z, t)$ para o espaço discreto $(x_i, z_j) = (i\Delta x, j\Delta z)$ e para um instante de tempo discreto $t^n = n\Delta t$ é representado por:

$$p_{i,j}^n = p(x_i, z_j, t_n) = p(i\Delta x, j\Delta z, n\Delta t), \quad (5)$$

onde $p_{i,j}^n$ denota o campo de pressão na malha, Δx e Δz são os espaçamentos ao longo dos eixos x_i e z_j , respectivamente, e Δt é o tamanho do passo no tempo.

Agora usando uma aproximação de diferenças finitas centrada de segunda ordem para derivada temporal na equação (1),

$$\left(\frac{\partial^2 p}{\partial t^2} \right)_{i,j}^{n-1} = \frac{p_{i,j}^n - 2p_{i,j}^{n-1} + p_{i,j}^{n-2}}{(\Delta t)^2} + \mathcal{O}^2, \quad (6)$$

onde $\mathcal{O}^2 = \mathcal{O}(\Delta t^2)$ é o erro de truncamento da aproximação da derivada de tempo do campo de pressão. Enquanto que, usando uma aproximação de diferenças finitas centrada de ordem $2M$ para o operador Laplaciano

$$\left(\nabla^2 p \right)_{i,j}^{n-1} = a_0 \left[\frac{1}{(\Delta x)^2} + \frac{1}{(\Delta z)^2} \right] p_{i,j}^{n-1} + \sum_{l=1}^{M-1} a_l \left\{ \left[\frac{p_{i-l,j}^{n-1} + p_{i+l,j}^{n-1}}{(\Delta x)^2} \right] + \left[\frac{p_{i,j-l}^{n-1} + p_{i,j+l}^{n-1}}{(\Delta z)^2} \right] \right\} + \mathcal{O}_{2M}, \quad (7)$$

onde a_l são os coeficientes do operador de diferenças finitas, M é a metade do comprimento do operador de diferenças finitas e $\mathcal{O}_{2M} = \mathcal{O}(\Delta x^{2M}, \Delta z^{2M})$ é o erro de truncamento das aproximações das derivadas espaciais.

Finalmente, pode-se resolver a equação (1) usando um esquema explícito de diferenças finitas, que atualiza o campo de onda de pressão através da seguinte relação recursiva:

$$p_{i,j}^n = 2p_{i,j}^{n-1} - p_{i,j}^{n-2} + c_{i,j}^2 (\Delta t)^2 \left[\left(\nabla^2 p \right)_{i,j}^{n-1} + W_{i,j}^{n-1,s} \right] + \mathcal{O}_{2M}^2, \quad (8)$$

onde $p_{i,j}^n$ é campo de onda na malha de diferenças finitas, $(\nabla^2 p)_{i,j}^n$ é uma matriz quadrada simétrica do operador Laplaciano discreto, $W_{i,j}^{n,s}$ é uma matriz de zeros exceto nas coordenadas das fontes e $\mathcal{O}_{2M}^2 = \mathcal{O}(\Delta t^2, \Delta x^{2M}, \Delta z^{2M})$ é o erro de truncamento das aproximações de diferenças finitas. A equação (8) assume condições iniciais para os campos de pressão (ou seja, quando $n = 2$) da forma

$$p_{i,j}^2 = 0, \quad p_{i,j}^1 = 0 \quad \text{e} \quad p_{i,j}^0 = 0 \quad (9)$$

e condições de fronteiras para os campos de pressão,

$$p_{0,j}^n = 0, \quad p_{i,0}^n = 0, \quad p_{N_x,j}^n = 0 \quad \text{e} \quad p_{i,N_z}^n = 0, \quad (10)$$

onde os índices i e j variam até N_x e N_z , que denotam o número de pontos na malha ao longo dos eixos x_i e z_j , respectivamente, e $n = 2, \dots, N_t + 2$.

Gradiente da função objetivo de quadrados mínimos

A função objetivo de quadrados mínimos discretizada, que mede a diferença entre o campo de pressão modelado $p_{i,j}^n$ e campo de pressão observado d_r^n é (Tarantola, 1984):

$$J(c_{i,j}) = \frac{1}{2} \sum_{s=1}^{N_s} \sum_{n=0}^{N_t} \sum_{r=1}^{N_r} \left(S_r p_{i,j}^n - d_r^n \right)^2, \quad (11)$$

onde S_r denota uma matriz para registrar o campo de pressão nos receptores, s e n são os índices que representam as fontes e as etapas de tempo, respectivamente, e N_s , N_r e N_t são o número de fontes, o número de receptores e o tempo total de modelagem, respectivamente. A função objetivo é função do modelo de velocidade discreto através de $p_{i,j}^n = P_{i,j}^n(c_{i,j})$.

O problema inverso consiste em minimizar a equação (11) em relação ao modelo de velocidade e tendo a equação (8) com restrição. Em outras palavras, o problema inverso é tratado como um problema de minimização restrito, que pode ser reescrito como um problema de minimização sem restrição através do método da Lagrangeana (Plessix, 2006). Isto é,

$$\mathcal{L} = \frac{1}{2} \sum_{s=1}^{N_s} \sum_{n=0}^{N_t} \sum_{r=1}^{N_r} \sum_{i=0}^{N_x} \sum_{j=0}^{N_z} \left(S_r p_{i,j}^n - d_r^n \right)^2 - \sum_{s=1}^{N_s} \sum_{n=0}^{N_t} \sum_{i=0}^{N_x} \sum_{j=0}^{N_z} \times \lambda_{i,j}^n \left\{ p_{i,j}^n - 2p_{i,j}^{n-1} + p_{i,j}^{n-2} - c_{i,j}^2 (\Delta t)^2 \left[(\nabla^2 p)_{i,j}^{n-1} + W_{i,j}^{n-1,s} \right] \right\},$$

onde $\mathcal{L} = \mathcal{L}(c_{i,j}; p_{i,j}^n, \lambda_{i,j}^n)$ é a Lagrangiana que associa a função objetivo (11) e a equação da onda (8) através do multiplicador de Lagrange $\lambda_{i,j}^n$.

Para derivar o gradiente da função objetivo em relação a $c_{i,j}$ avaliamos a perturbação de primeira ordem da Lagrangeana, que se desenvolve tomando a perturbação da velocidade discreta; que deve produzir também pequenas perturbações ao campo de pressão discreto. Assim, depois algumas considerações, obtemos o gradiente da função objetivo (11) com relação ao modelo de velocidade a partir da variação da Lagrangiana associada, que segue

$$[\nabla J(c)]_{i,j}^n = \frac{2}{c_{i,j}} \sum_{s=1}^{N_s} \sum_{n=0}^{N_t} \lambda_{i,j}^n \left(\frac{\partial^2 p}{\partial t^2} \right)_{i,j}^n (\Delta t)^2, \quad (12)$$

onde $\lambda_{i,j}^n$ é agora o campo de onda adjunto e é obtido a partir da relação recursiva avaliada no tempo reverso

$$\lambda_{i,j}^n = 2\lambda_{i,j}^{n+1} - \lambda_{i,j}^{n+2} + c_{i,j}^2 (\Delta t)^2 \left[(\nabla^2 \lambda)_{i,j}^{n+1} + \frac{1}{c_{i,j}^2 (\Delta t)^2} S_r^\dagger (S_r p_{i,j}^n - d_r^n) \right]$$

e têm como condições finais para $\lambda_{i,j}^n$ (ou seja, para $n = N_t$)

$$\lambda_{i,j}^{N_t+2} = 0, \quad \lambda_{i,j}^{N_t+1} = 0 \quad \text{e} \quad \lambda_{i,j}^{N_t} = 0 \quad (13)$$

com $n = N_t, N_t - 1, \dots, 0$ e onde \dagger denota o transposto. As condições de fronteiras para campo adjunto são analoga as condições de fronteiras definidas para o campo de pressão (10).

A equação (12) define a correlação cruzada de atraso zero entre o campo de onda adjunto $\lambda_{i,j}^n$ e a variação temporal de segunda ordem do campo de onda da fonte $p_{i,j}^n$.

Derivação do gradiente da função objetivo usando Diferenciação Automática

Gradiente da função objetivo de quadrados mínimos

O gradiente da função objetivo $J = J(p_{i,j}^n(c_{i,j}))$ com relação ao modelo de velocidade pode ser obtido usando a regra da cadeia no modo reverso, que segue (Sun et al., 2019):

$$[\nabla J(c)]_{i,j}^n = \sum_{n=0}^{N_t} \left(\frac{\partial J}{\partial p_{i,j}^n} \right) \frac{\partial p_{i,j}^n}{\partial c_{i,j}}. \quad (14)$$

A expansão da derivada parcial $(\partial J / \partial p_{i,j}^n)$ é realizada de acordo com a relação recursiva para calcular o campo

$p_{i,j}^n$, equação (8), que para atualizar o campo de pressão no instante n depende dos campos avaliados em dois instantes de tempo passados. Isso implica que $(\partial J / \partial p_{i,j}^n)$ são determinados por três termos dependência, avaliados para $0 < n < N_t - 2$. Assim, o gradiente de $J(c_{i,j})$ é obtido avaliando as derivadas parciais que são formadas da expansão de $(\partial J / \partial p_{i,j}^n)$ e é dada por (Sun et al., 2019):

$$[\nabla J(c)]_{i,j}^n = \frac{2}{c_{i,j}} \sum_{s=1}^{N_s} \sum_{n=0}^{N_t} \lambda_{i,j}^n \left(\frac{\partial^2 p}{\partial t^2} \right)_{i,j}^n (\Delta t)^2 \quad (15)$$

com $\lambda_{i,j}^n$ sendo obtido pela retropropagação do resíduo, calculado através da relação recursiva abaixo:

$$\lambda_{i,j}^n = 2\lambda_{i,j}^{n+1} - \lambda_{i,j}^{n+2} + c_{i,j}^2 (\Delta t)^2 \left[(\nabla^2 \lambda)_{i,j}^{n+1} + \frac{1}{c_{i,j}^2 (\Delta t)^2} S_r^\dagger (S_r p_{i,j}^n - d_r^n) \right]$$

para $n = N_t - 2, N_t - 3, \dots, 0$, enquanto que, os estados iniciais da propagação reversa do campo $\lambda_{i,j}^n$ são, respectivamente,

$$\lambda_{i,j}^{N_t} = \sum_{r=1}^{N_r} S_r^\dagger (S_r p_{i,j}^{N_t} - d_r^{N_t}),$$

$$\lambda_{i,j}^{N_t-1} = \sum_{r=1}^{N_r} \left\{ S_r^\dagger (S_r p_{i,j}^{N_t} - d_r^{N_t}) \left[c_{i,j}^2 (\Delta t)^2 \nabla^2 + 2 \right] + S_r^\dagger (S_r p_{i,j}^{N_t-1} - d_r^{N_t-1}) \right\}.$$

Portanto, o gradiente da função objetivo de quadrados mínimos derivado através do ASM discreto, equação (12), é igual ao gradiente derivado pelo modo reverso de AD, equação (15).

Computação do gradiente usando o Enzyme

Para facilitar a descrição, considere a equação (8) reescrita da seguinte forma:

$$p_{i,j}^n - \left[2 + c_{i,j}^2 (\Delta t)^2 \nabla^2 \right] p_{i,j}^{n-1} + p_{i,j}^{n-2} = W_{i,j}^{n-1,s}. \quad (16)$$

Para escrever equação (16) na forma compacta, define-se

$$\mathbf{u}^n = [p_{1,1}^n, \dots, p_{N_x,1}^n, p_{1,2}^n, \dots, p_{N_x,2}^n, \dots, p_{1,N_z}^n, \dots, p_{N_x,N_z}^n]^\dagger,$$

$$\mathbf{v}^n = [W_{1,1}^n, \dots, W_{N_x,1}^n, W_{1,2}^n, \dots, W_{N_x,2}^n, \dots, W_{1,N_z}^n, \dots, W_{N_x,N_z}^n]^\dagger,$$

com dimensão $N = N_x \times N_z$, onde N é o número total de pontos da malha. Agora escrevendo o termo entre colchetes na equação (16) na forma: $\mathbf{G} = 2\mathbf{I} + \mathbf{AD}$, onde $\mathbf{A} = \text{diag}\{c_{i,j}^2 (\Delta t)^2\}$ é uma matriz diagonal de dimensão $N \times N$, \mathbf{D} é uma matriz simétrica do operador Laplaciano discretizado com dimensão $N \times N$ e \mathbf{I} é a matriz identidade também com dimensão $N \times N$. Assim, podemos reescrever a equação (16) na forma

$$\mathbf{u}^n - \mathbf{G}\mathbf{u}^{n-1} + \mathbf{u}^{n-2} = \mathbf{v}^{n-1}. \quad (17)$$

O algoritmo (1) descreve a implementação da função objetivo de quadrados mínimos fornecido a ferramenta de AD Enzyme.jl e é resumindo como segue: i) entre com o

modelo de velocidade $c_{i,j}$ e o dado observado $\tilde{\mathbf{d}}$, ii) define e inicialize o campo de onda \mathbf{u}^n e o dado calculado \mathbf{d} , iii) resolva a equação (17), iv) calcule \mathbf{d} para o r -ésimo receptor. Isto é, $\mathbf{d} = S_r \mathbf{u}^n$. O número de amostras de \mathbf{d} e de $\tilde{\mathbf{d}}$ são representas por N_a e v) calcule a equação (11).

Para calcular o gradiente da função objetivo de quadrados mínimos usando a ferramenta de AD Enzyme passamos o código da função objetivo com todas suas dependências, ou seja, com os dados de entradas e com o modelador da equação (17), algoritmo (1).

Algoritmo 1: Estrutura do código da função objetivo de quadrados mínimos passado ao pocote de AD Enzyme.jl.

- 1: **Entrada:** modelo de velocidade $c_{i,j}$, dado observado $\tilde{\mathbf{d}}$;
- 2: $\mathbf{u} = \mathbf{0}$ Inicialize o campo de onda;
- 3: $\mathbf{d} = \mathbf{0}$ Inicialize o dado calculado;
- 4: **para** cada etapa de tempo n
- 5: $\mathbf{u}^n = \mathbf{G}\mathbf{u}^{n-1} - \mathbf{u}^{n-2} + \mathbf{v}^{n-1}$
- 6: $\mathbf{d} = S_r \mathbf{u}^n$
- 7: **fim**
- 8: $J(c_{i,j}) = \frac{1}{2} \sum_{a=1}^{N_a} \sum_{r=1}^{N_r} (\mathbf{d} - \tilde{\mathbf{d}})^2$
- 9: **Resultado:** valor da função objetivo, $J(c_{i,j})$;

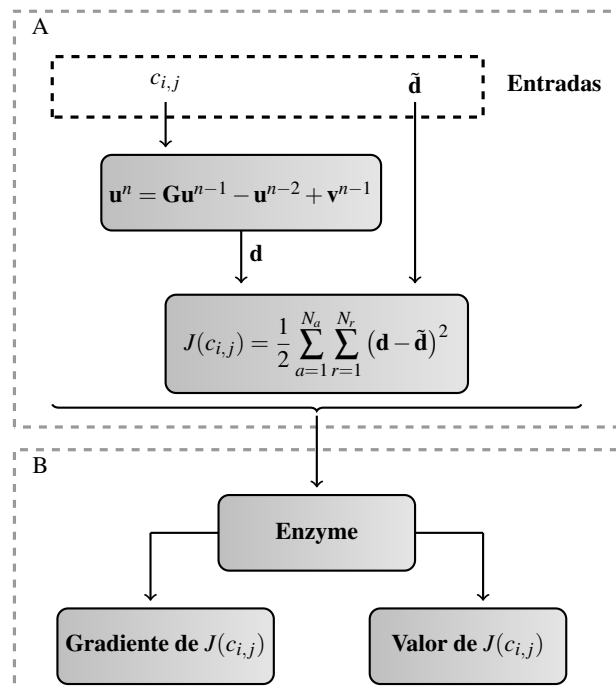


Figura 1: Fluxograma para calcular o gradiente da função objetivo de quadrados mínimos usando a ferramenta de AD Enzyme. O bloco A mostra como o valor da função objetivo é calculado, que exige resolver a equação da onda para obter o dado modelado. Enquanto que, o bloco B ilustra como o gradiente da função objetivo é obtido usando o Enzyme, que também retorna com o valor da função objetivo.

Na Figura 1 mostramos o fluxograma para calcular o gradiente de $J(c_{i,j})$ usando o Enzyme. O bloco A ilustra a implementação da função objetivo, algoritmo 1. Enquanto que, o bloco B mostra como calcular o gradiente da função objetivo usando Enzyme. A estrutura descrita pelo fluxograma é implementada dentro do laço dos tiros.

Inversão da Forma de Onda Completa

A FWI é formulada como um problema de minimização restrito que de modo prático é resolvida de forma iterativa usando a seguinte expressão (Virieux and Operto, 2009):

$$\mathbf{m}_{k+1} = \mathbf{m}_k + \alpha_k \mathbf{p}_k, \quad (18)$$

onde \mathbf{m}_k é o vetor dos parâmetro do modelo, que para o caso acústico é formado por todos os elementos de $c_{i,j}$ para $i = 1, \dots, N_x$ e $j = 1, \dots, N_z$, \mathbf{p}_k é chamado de vetor direção de busca, que define a direção provável da melhor estimativa de \mathbf{m}_{k+1} , α_k é o comprimento de passo (um escalar positivo), que define o quanto se deve avançar na direção de busca para atualizar o \mathbf{m}_k e k é o índice que descreve as iterações da inversão.

Uma escolha apropriada para o vetor direção de busca define uma variedade de métodos de otimização para resolver o problema da FWI (Nocedal and Wright, 2006).

Método de gradiente conjugado não linear

O NLCG é uma extensão do método de gradiente conjugado (CG) para problemas de otimização não linear. Esse é definido de forma similar ao método CG, ou seja, ele diminui o valor da função objetivo ao longo da direção conjugada (Nocedal and Wright, 2006):

$$\mathbf{p}_k = -\mathbf{g}_k, \quad k = 0, \quad (19)$$

$$\mathbf{p}_k = -\mathbf{g}_k + \beta_k \mathbf{p}_{k-1} \quad k > 0, \quad (20)$$

onde \mathbf{g}_k é o vetor gradiente, ou seja, definido através da equação (12) ou da equação (15). No método CG existe um expressão geral para calcular o parâmetro escalar β_k . Mas, para o método NLCG existem diferentes expressões para expressar o β_k (Hager and Zhang, 2006). No trabalho proposto por Dai and Yuan (2001) é sugerido um método híbrido, que segue:

$$\beta_k = \max\left(0, \min\left(\beta_k^{HS}, \beta_k^{DY}\right)\right), \quad (21)$$

onde β_k^{HS} e β_k^{DY} são, respectivamente,

$$\beta_k^{HS} = \frac{\mathbf{g}_k \cdot (\mathbf{g}_k - \mathbf{g}_{k-1})}{(\mathbf{g}_k - \mathbf{g}_{k-1}) \cdot \mathbf{p}_{k-1}} \quad \text{e} \quad \beta_k^{DY} = \frac{\mathbf{g}_k \cdot \mathbf{g}_k}{(\mathbf{g}_k - \mathbf{g}_{k-1}) \cdot \mathbf{p}_{k-1}},$$

No método NLCG a primeira iteração, $k = 0$, é a direção de busca de máxima descida, (19), enquanto que, para as iterações $k \geq 1$ é definida a direção conjugada, (20).

Resultados

Usamos o modelo de velocidade Marmousi, Figura 2a, para validação tanto da implementação do ASM quanto do Enzyme na computação do gradiente. A geometria de aquisição usada neste experimento mantém a linha dos receptores fixa na superfície do modelo, enquanto que, as coordenadas da fonte variam uniformemente a cada 100 m.

Neste experimento usamos 199 receptores espaçados em 20 m e na profundidade de 10 m. O traço registrado contém 2001 amostras com cada amostra gravada a cada 0,002 s para um tempo total de modelagem de 4 s. A frequência pico usada para a fonte do tipo Ricker foi 8 Hz.

O modelo inicial correspondente ao modelo de velocidade Marmousi verdadeiro é mostrado na Figura 2b, com velocidade variando verticalmente de 1500 m/s, na camada d'água, para 3400 m/s, na parte mais profunda do modelo.

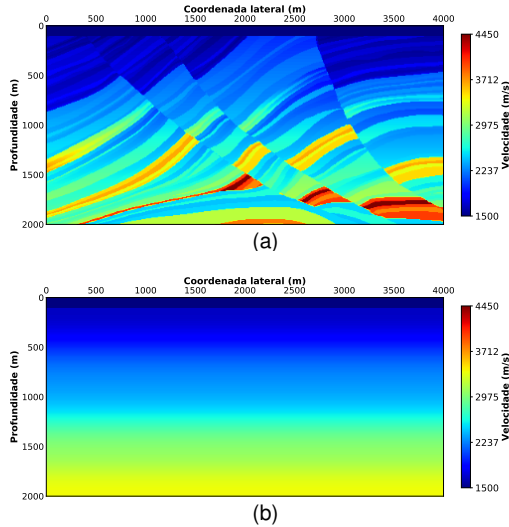


Figura 2: Modelo de velocidade Marmousi: (a) verdadeiro e (b) inicial.

Comparação dos gradientes ASM e AD

Os gradientes obtidos tanto pelo ASM (3a) quanto pela AD (3b) foram calculados para um experimento de múltiplos tiros, que totalizaram 40 tiros. Avaliamos também o custo computacional (4a) e demanda de alocação (4b) nesse experimento. Além disso, destacamos que tal análise foi realizada para execuções sem paralelismo.

Na Figura 3 mostramos os gradientes obtidos pelo ASM, Figura 3a, e pela AD, Figura 3b. Apresentamos também a diferença entre estes dois gradientes na Figura 3c.

Para uma análise quantitativa avaliamos a máxima diferença relativa entre eles, usando (Cao and Liao, 2015):

$$\max(|\mathbf{g}_{ASM} - \mathbf{g}_{AD}| / \|\mathbf{g}_{ASM}\|_2), \quad (22)$$

onde \mathbf{g}_{ASM} é o gradiente obtido pelo ASM e \mathbf{g}_{AD} é o gradiente obtido pela AD. Assim, a máxima diferença relativa entre os dois gradientes é aproximadamente 2%.

A Figura 4a mostra o custo para calcular os gradientes usando o ASM e a AD. Enquanto que, na Figura 4b mostramos a demanda de alocação para a computação dos dois gradientes.

Em termos de custo computacional a computação do gradiente usando a AD foi aproximadamente uma vez e um décimo mais caro em relação ao gradiente calculado com o ASM. A demanda de alocação para a computação do gradiente com AD apresentou duas ordens de magnitude a mais em relação a correspondente usando o ASM.

Comparação das inversões com ASM e AD

A inversão foi realizada para 100 iterações e número total de iterações foi assumido como critério de parada. Os resultados das inversões usando os gradientes obtidos pelo ASM e pela AD são mostrados nas Figuras 5a e 5b, respectivamente.

Para realizar análise quantitativa das estimativas dos modelos de velocidades usamos a medida de índice de

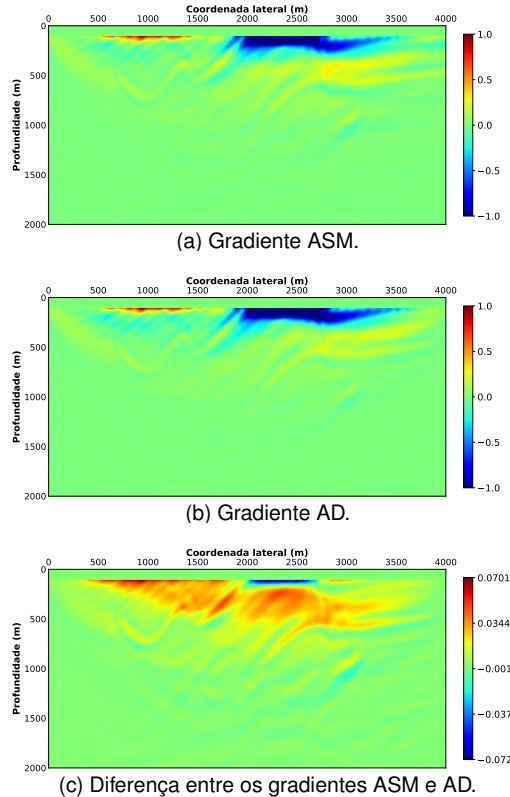


Figura 3: Gradiente da função objetivo usando: (a) o ASM e (b) a AD. Em (c) é mostrado a diferença entre os gradientes obtidos pelo ASM e pela AD.

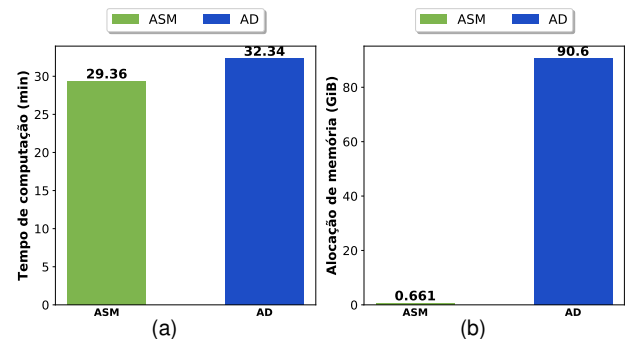


Figura 4: Gradiente calculado usando o ASM e a AD, mostrando: (a) custo computacional em minutos e (b) demanda de armazenamento em GiB.

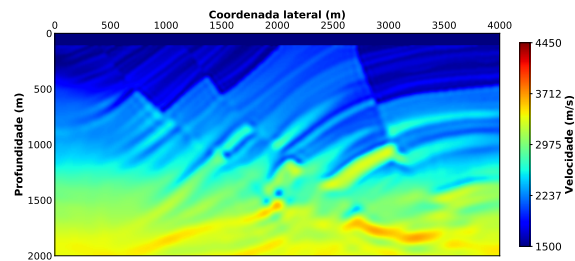
similaridade estrutural (SSIM) (Wang et al., 2004):

$$SSIM(\mathbf{m}, \mathbf{m}_e) = L(\mathbf{m}, \mathbf{m}_e)^{\gamma_1} C(\mathbf{m}, \mathbf{m}_e)^{\gamma_2} S(\mathbf{m}, \mathbf{m}_e)^{\gamma_3}, \quad (23)$$

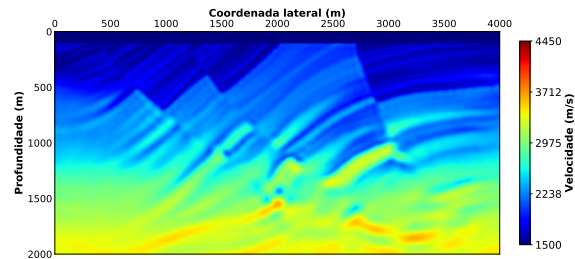
onde \mathbf{m} é o modelo de velocidade verdadeiro e \mathbf{m}_e é o modelo de velocidade estimado, L , C e S são três medidas estatísticas, chamadas de luminância, contraste e estrutura, respectivamente, e γ_1 , γ_2 e γ_3 são fatores de ponderação, que foram assumidos iguais a 1.

A SSIM mostrou que os modelos de velocidades estimados com o método NLCG usando os gradientes calculado pelo ASM e pela AD discordaram em 1.6%.

A inversão usando os dois gradientes apresentaram um



(a) Modelo estimado usando o gradiente obtido pelo ASM.



(b) Modelo estimado usando o gradiente obtido pela AD.

Figura 5: Resultados da FWI com o método NLCG usando os gradientes calculados: (a) pelo ASM e (b) pela AD.

comportamento muito similar na redução do valor da função objetivo (11), como mostrado na Figura 6.

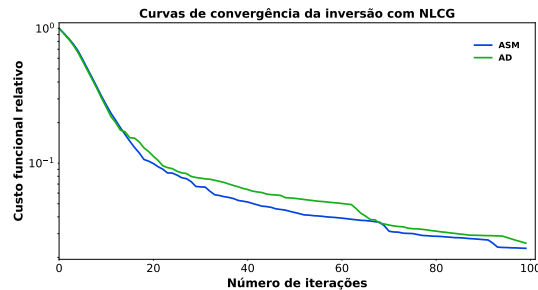


Figura 6: Comparação das curvas de convergência das inversões com o método NLCG usando os gradientes obtidos pelo ASM, curva azul, e pela AD, curva verde.

Discussão e Conclusão

Neste trabalho, mostra-se que a AD pode ser usada como uma abordagem alternativa para validar o cálculo do gradiente de funções objetivas usadas na FWI. Em outras palavras, a AD pode reduzir o tempo de desenvolvimento e possibilitar análise de outras funções objetivas e a inversão para problemas de maior complexidade, por exemplo, a inversão para meios viscoelásticos, com pouco custo de implementação adicional.

Ressaltamos que para calcular o gradiente da função objetivo usando AD é essencial que as condições de fronteira de absorção seja consideradas. Não considerar as bordas de absorção para calcular o gradiente por AD conduz à efeitos de alta amplitude ao longo das bordas do domínio do gradiente.

Experimentos no modelo de velocidade Marmousi, mostraram que o custo computacional para calcular os gradientes tanto com AD quanto com o ASM são

similares. No entanto, a demanda de armazenamento para calcular o gradiente usando AD apresentou duas ordens de magnitude maior com relação a demanda de armazenamento para calcular o gradiente usando o ASM.

Agradecimentos

A FUNCAMP e ao INCT-GP pelo suporte financeiro. Ao CPGf/UFGA pelo suporte à pesquisa.

Referências

- Cao, D., and W. Liao, 2015, A computational method for full waveform inversion of crosswell seismic data using automatic differentiation: *Computer Physics Communications*, **188**, 47–58.
- Dai, Y., and Y. Yuan, 2001, An efficient hybrid conjugate gradient method for unconstrained optimization: *Annals of Operations Research*, **103**, 33–47.
- Elliott, C., 2018, The simple essence of automatic differentiation: *Proceedings of the ACM on Programming Languages*, **2**, 1–29.
- Griewank, A., and A. Walther, 2000, Algorithm 799: revolve: an implementation of checkpointing for the reverse or adjoint mode of computational differentiation: *ACM Trans. Math. Softw.*, **26**, 19–45.
- Hager, W. W., and H. Zhang, 2006, A survey of nonlinear conjugate gradient methods: *Pacific journal of Optimization*, 35–58.
- Klein, W., A. Griewank, and A. Walther, 2002, *Differentiation methods for industrial strength problems*: Springer New York.
- Moses, W. S., V. Churavy, L. Paehler, J. Hüchelheim, S. H. K. Narayanan, M. Schanen, and J. Doerfert, 2021, Reverse-mode automatic differentiation and optimization of gpu kernels via enzyme: Presented at the Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, Association for Computing Machinery.
- Nocedal, J., and S. J. Wright, 2006, *Numerical optimization*: Springer.
- Plessix, R. E., 2006, A review of the adjoint-state method for computing the gradient of a functional with geophysical applications: *Geophysical Journal International*, **167**, 495–503.
- Sambridge, M., P. Rickwood, N. Rawlinson, and S. Sommacal, 2007, Automatic differentiation in geophysical inverse problems: *Geophysical Journal International*, **170**, 1–8.
- Sun, J., Z. Niu, K. Innanen, J. Li, and D. Trad, 2019, A deep learning perspective of the forward and inverse problems in exploration geophysics: Presented at the .
- Tarantola, A., 1984, Inversion of seismic reflection data in the acoustic approximation: *Geophysical*, **49**, 1259–1266.
- Virieux, J., and S. Operto, 2009, An overview of full-waveform inversion in exploration geophysics: *Geophysics*, **74**, WCC1–WCC26.
- Wang, Z., A. Bovik, H. Sheikh, and E. Simoncelli, 2004, Image quality assessment: from error visibility to structural similarity: *IEEE Transactions on Image Processing*, **13**, 600–612.
- Zhu, W., K. Xu, E. Darve, and G. C. Beroza, 2021, A general approach to seismic inversion with automatic differentiation: *Computers and Geosciences*, **151**, 104751.