

4

Metodologia e resultados preliminares para análise de velocidade utilizando o gradiente descendente

O processamento utilizando diferentes equações de sobretempo normal para a obtenção de análise de velocidade para meios com anisotropia VTI e a aplicação das equações em dados sísmicos sintéticos resultou na necessidade de tornar o processo de análise de velocidade mais simples e passível de ser adaptado a qualquer tipo de equação, visto que aparecem na literatura algumas sugestões de equações de NMO que possuem formatos diferentes da proposta por Taner & Koehler (1969), onde a função é baseada na série de Taylor. Propostas como Castle (1994) que propôs uma equação de NMO prática que corrige o sobretempo para dados com afastamentos longos, permite uma extensão de sua equação para anisotropia VTI, como as propostas de Siliqi & Bousquié (2000) e Elapavuluri (2003), e o formato de sua equação não possui termos “independentes” (como no modelo de equações baseadas na série de Taylor), tornando mais difícil o uso do método do *semblance* para esse tipo de equação.

O método proposto neste capítulo - que utiliza um minimizador de funções chamado gradiente descendente - permite a utilização de diferentes equações de NMO, com resultados obtidos de forma rápida e sem muita interferência do intérprete. Também possibilita que vários parâmetros sejam encontrados ao mesmo tempo, facilitando e agilizando ao usuário a utilização de equações envolvendo anisotropia como é o caso das equações de quarta ordem já utilizadas na prática.

Pelas vantagens citadas acima, além de equações mais complexas, a equação de correção de NMO de segunda ordem (equação isotrópica) também é testada e responde com bastante eficiência ao método, que pode ser utilizado inclusive para análise de velocidade convencional (abordagem isotrópica).

Pelo tipo de solução proposta, um minimizador de funções, este método poderia também ser testado/utilizado em outras aplicações envolvendo dados sísmicos que precisem ajustar os dados a uma equação e/ou realizar algum tipo de inversão ou ajuste de parâmetros. É necessário apenas saber o formato da equação e os dados que são utilizados no ajuste desta equação.

4.1. Funções discriminantes lineares

O termo funções discriminantes lineares em classificação de padrões é atribuído a funções de formato aproximadamente linear, que caracterizam um conjunto de pontos da forma (x, y) , sem a necessidade do conhecimento das distribuições probabilísticas dos dados, já que apenas uma análise não paramétrica é necessária para encontrar a função.

As funções discriminantes lineares assumem a forma:

$$g(y) = a \cdot y', \quad \text{Eq. 48}$$

onde y é função de x . Ou seja, se a função $g(x)$ é não linear, porém existe um vetor $y = f(x)$ que torne a função $g(y)$ linear, este problema pode ser resolvido a partir de funções discriminantes lineares.

As equações de NMO são funções matemáticas que se aproximam de funções polinomiais em afastamento (x) e tempo (t) pelo fato da grande maioria destas equações ser baseada na solução da série de Taylor (seção 2.4.1), que resulta numa função linear. Portanto, estas equações são boas candidatas a utilizarem funções discriminantes como método de solução.

Existem vários métodos de realizar essa análise não paramétrica e trata-se sempre de um problema de minimização de uma função de custo. São procedimentos de cálculo muito simples e são satisfatórios até quando existem poucos dados, principalmente pela sua característica de rápida convergência. A principal desvantagem do método é que a forma geral da função deve ser previamente conhecida (ou determinada) e que algumas operações matemáticas (gradientes) sejam determinadas e definidas no sistema para que os pontos existentes (x, y) consigam estimar os parâmetros desejados de modo satisfatório. Um estudo detalhado de convergência para essa análise não paramétrica pode ser vista em Duda et al. (2001).

Uma forma simples de resolver esta análise é uma metodologia conhecida como gradiente descendente, que reduz o problema à minimização de uma função escalar. Este método é explicado com mais detalhes a seguir.

4.1.1. Método do gradiente descendente

O método de minimização de funções se baseia no fato de uma função de custo (ou função de erro) ser zero quando a solução correta é encontrada. Se a função é linear (Eq. 48), este procedimento reduz o problema a uma minimização de uma função escalar (a) pela subtração do gradiente da função de custo, como pode ser visto na Eq. 49:

$$a(k+1) = a(k) - n(k)\nabla J(a(k)). \quad \text{Eq. 49}$$

A partir desta equação, observa-se que existem dois parâmetros que podem gerar problemas no uso do gradiente descendente: o parâmetro de aprendizado $n(k)$ e a função de custo $J(a)$. Ambos os parâmetros permitem um grau de escolha muito grande por serem parâmetros livres (podendo a princípio assumir qualquer valor) e podem definir a velocidade de convergência do processo tornando o incremento de $a(k)$ de maior ou menor intensidade.

O parâmetro de aprendizado é o peso que o gradiente da função de custo terá na atualização do parâmetro a (Eq. 49) e esta escolha deve ser sempre levada em consideração. Se n é muito pequeno, a convergência de $a(k+1)$ na Eq. 49 é desnecessariamente lenta pelo fato do incremento de $a(k)$ ser muito pequeno, caso contrário, se n é muito grande, o termo $a(k)$ recebe um incremento muito grande e o processo pode divergir (Duda, et al., 2001; Pedreira, 2003). Existem algumas propostas de função para $n(k)$, como uma função chamada “algoritmo de Newton” (Duda, et al., 2001), que ainda não foram testadas neste trabalho. A função utilizada na maioria dos exemplos realizados aqui é uma função tipo escada que diminui o valor com o aumento de iterações e possui valores entre 0,05 e 0,001. Esta função foi considerada satisfatória para todos os exemplos realizados.

A função de custo é uma função que deve ser escolhida com cuidado para que seu gradiente seja capaz de atualizar os parâmetros. Para isto, algumas funções podem ser escolhidas de modo que o método funcione de maneira mais correta.

Se o problema for o de resolver uma desigualdade do tipo $a \cdot y' > 0$, uma escolha interessante para a função de custo seria a função da Eq. 50, também chamada de função de custo de *Perceptron* (Pedreira., 2003). O gradiente desta função é um simples somatório em y :

$$J(a) = \sum_{j \in Y} (-a \cdot y^j). \quad \text{Eq. 50}$$

Para o caso utilizado neste trabalho (Eq. 48), uma equação de “relaxamento” (Duda, et al., 2001) é utilizada como função de custo (Eq. 51):

$$J(a) = \frac{1}{2} \sum_{j \in Y} \frac{(a \cdot y^j - g)^2}{\|y^j\|^2}. \quad \text{Eq. 51}$$

Esta função permite calcular o quanto os parâmetros $(a \cdot y^j)$ estão próximos do valor real (g) pela diferença entre eles, e assim aproximando a função de custo de zero quando os parâmetros estão satisfatórios.

A convergência do método pode ser verificada em Duda et al. (2001), para vários tipos de função utilizadas como parâmetro de aprendizado e como função de custo. Em todos os casos utilizados nesse trabalho, o gradiente descendente mostrou ser um método robusto e de convergência rápida.

A execução do método é simples e realizada de modo iterativo, como mostrado a seguir:

1. Escolher um valor inicial para $a = a(0)$, para a iteração $k = 0$;
2. Calcular ∇J para cada parâmetro a utilizando todos os pares (x, y) disponíveis para a minimização;
3. Atualizar o parâmetro $a(k+1)$ com o vetor calculado de $\nabla J(a(k))$ pela Eq. 49;
4. Repetir o procedimento a partir do item 2, considerar $k = k+1$ e repetir até o procedimento atingir um determinado número de iterações ou a diferença encontrada entre os parâmetro $a(k)$ e $a(k+1)$ seja menor que um determinado valor de erro determinado pelo usuário.

4.2. Equações de NMO adaptadas ao gradiente descendente

Após conhecido o método é preciso adaptar as equações que queremos usar ao método do gradiente descendente. Foram escolhidas inicialmente três equações para serem colocadas na forma de gradiente descendente. A equação mais simples e hiperbólica (Eq. 52) e duas equações utilizadas neste trabalho para correção de NMO de dados com anisotropia VTI: a equação desenvolvida por Alkhalifah & Tsvankin (1995) e Alkhalifah (1997) (Eq. 55) e a equação de hipérbole deslocada proposta por

Castle (1994) com solução que permite a utilização em meios anisotrópicos (Eq. 57) proposta por Siliqi & Bousquié (2000).

A equação hiperbólica, usualmente utilizada para realizar a análise de velocidade é a primeira equação inserida no método gradiente descendente:

$$t^2 = t_0^2 + \frac{x^2}{V_{nmo}^2}. \tag{Eq. 52}$$

Esta equação pode ser escrita da forma:

$$t^2 = a_1 + a_2 x^2 \tag{Eq. 53}$$

e na notação matricial da Eq. 48 $g(x) = t^2$, $a = \begin{bmatrix} t_0^2 & \frac{1}{V_{nmo}^2} \end{bmatrix}$ e $y = \begin{bmatrix} 1 & x^2 \end{bmatrix}$, onde

$a_1 = t_0^2$, $a_2 = 1/V_{nmo}^2$, $y_1 = 1$ e $y_2 = x^2$. Os gradientes da função de custo são então calculados a partir da Eq. 51:

$$\nabla J_{a1}(k) = \sum_{y \in Y} \frac{((a_1(k) \cdot y_1 + a_2(k) \cdot y_2) - g(x))}{y_1^2 + y_2^2} \cdot y_1 \text{ e} \tag{Eq. 54a}$$

$$\nabla J_{a2}(k) = \sum_{y \in Y} \frac{((a_1(k) \cdot y_1 + a_2(k) \cdot y_2) - g(x))}{y_1^2 + y_2^2} \cdot y_2. \tag{Eq. 54b}$$

Após calculadas as funções para esta equação, aplica-se o algoritmo e encontram-se os valores de a para os pontos (x, t) disponíveis.

A segunda equação, proposta por Alkhalifah & Tsvankin (1995) para anisotropia VTI:

$$t^2 = t_0^2 + \frac{x^2}{V_{nmo}^2} - \frac{2\eta x^4}{V_{nmo}^2 [t_0^2 V_{nmo}^2 + (1 + 2\eta)x^2]}, \tag{Eq. 55}$$

utiliza um termo de x^2 no denominador para melhorar a sua convergência quando os afastamentos são grandes (Eq. 55 e seção 2.4.1), fazendo com que a equação não se torne mais linear, ou seja, uma função que não se classifica mais como uma possível candidata às funções discriminantes lineares. Porém, segundo Pedreira (2004), funções não lineares também podem ser adaptadas ao método e tudo depende da escolha da função de custo e de como essa função de custo auxilia na otimização do problema (minimizar seu gradiente quando o erro se aproximar de zero). Portanto, o mesmo procedimento será utilizado para esta equação.

Para a Eq. 55, $g(x) = t^2$, $a = \begin{bmatrix} t_0^2 & \frac{1}{V_{nmo}^2} & \eta \end{bmatrix}$ e $y = [1 \quad x^2 \quad x^4]$ na notação

matricial aproximada da Eq. 48. Devido a equação não ser mais linear, o cálculo do gradiente da função de custo J (Eq. 56) é um pouco mais complexo (vide rotina desenvolvida no Matlab® e listada no Apêndice A):

$$J(a) = \frac{1}{2} \sum_{j \in Y} \frac{(a_1 \cdot y_1 + a_2 \cdot y_2 - \frac{2a_3 a_2 y_3}{[(a_1/a_2) + (1 + 2a_3) \cdot y_2]} - g)^2}{\|y\|^2}, \quad \text{Eq. 56}$$

porém o método se comportou muito bem e os resultados com dados sintéticos foram considerados satisfatórios (seção 4.3).

A última equação utilizada neste trabalho é a equação de Castle (1994), observada na Eq. 57 (seção 2.4.1 para a explicação da equação geral e seção 5.1 para utilização em modelos anisotrópicos):

$$t = \left(1 - \frac{1}{S}\right) \cdot t_0 + \frac{1}{S} \sqrt{t_0^2 + S \cdot \frac{x^2}{V_{nmo}^2}} \quad \text{Eq. 57}$$

onde $S = 1 + 8\eta$ (Siliqi & Bousquié, 2000) é uma consideração que permite que a mesma seja utilizada para um modelo VTI. Esta equação, segundo Grechka (2005), é uma equação imprecisa para modelos anisotrópicos com poucas camadas, mas se torna uma equação mais confiável quando o modelo possui várias camadas, devido ao fato das camadas anisotrópicas e de heterogeneidade vertical se intercalarem.

Para esta equação, assim como a anterior (Eq. 55), o gradiente da função de custo foi calculado para os parâmetros definidos como $g(x) = t$,

$$a = \begin{bmatrix} t_0^2 & \frac{1}{V_{nmo}^2} & S \end{bmatrix} \text{ e } y = [1 \quad x^2] \text{ na notação matricial aproximada da Eq. 48 (ver}$$

rotina desenvolvida no Matlab® e listada no Apêndice A).

Os exemplos em dados sintéticos podem ser vistos a seguir, com alguns testes em todas as equações para verificar a convergência, eficiência, robustez e precisão do método.

4.3. Exemplos com dados sintéticos

Nesta seção são testados vários exemplos com o intuito de verificar a robustez e precisão do método. O primeiro experimento (subseção 4.3.1) testa as equações citadas na seção anterior com um evento sintético, variando a quantidade de pontos e introduzindo um erro de marcação do evento. O segundo experimento (subseção 4.3.2) é em um sismograma sintético gerado a partir do *array* com mais de um evento ao mesmo tempo, simulando um problema real.

4.3.1. Dados sintéticos - eventos

Um mesmo exemplo é utilizado para testar as três equações, variando o número de pontos (x, t) e também simulando um erro na marcação dos eventos, que, em casos reais, deve ser um procedimento manual ou parcialmente manual (dependendo da existência de alguma opção de marcação automática de eventos disponível para o intérprete).

O evento proposto possui as seguintes propriedades: $t_0 = 1,2s$, $V_{NMO} = V_{RMS} = 2,8km/s$ e $\eta = 0,2$ e será simulado nas três equações abordadas na seção 4.2. Para a execução do gradiente descendente, os três parâmetros iterativos e livres do método foram escolhidos e considerados constantes para todas as simulações: A taxa de aprendizado foi escolhida para $n = 0,005$, determinada no primeiro exemplo da primeira equação e mantida constante (considerada satisfatória) no restante dos testes. O número máximo de iterações foi de $K = 10^8$ e o erro aceitável para todos os parâmetros $\epsilon = 10^{-8}$, sendo estes dois últimos parâmetros responsáveis pela quantidade de iterações a serem realizadas (tempo que o método gasta na execução).

A Figura 42 mostra um gráfico do evento gerado com 200 pontos, simulando um dado sísmico com intervalo de estação de 25m (afastamento máximo de 5km), com uma simulação sem ruído (curva azul) e com desvios (considerados aqui como ruídos) de até 1% (não mostrado) e de até 10% (curva vermelha) na escolha dos pontos (x, t) utilizados como entrada do método. Estes erros simulam a escolha, de forma imprecisa, dos eventos sísmicos pelo intérprete, fazendo com que os dados de entrada do método possuam ruído. A Figura 43 mostra um caso onde são gerados

apenas 10 pontos, simulando um intervalo de estações de 500m num modelo de mesmo afastamento máximo. Foi gerado ainda um modelo intermediário com 80 pontos no mesmo afastamento máximo.

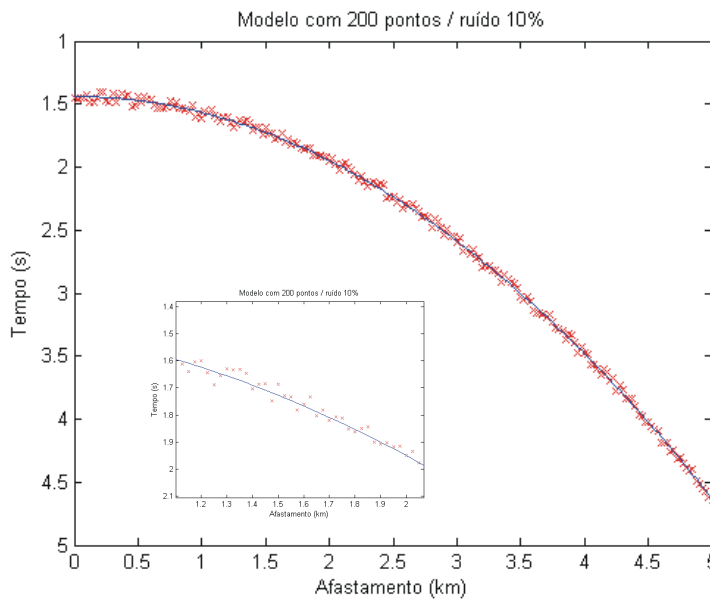


Figura 42: Evento sintético com 200 pontos utilizado para testes do gradiente descendente. Curva azul: original sem ruído; pontos vermelhos: original com ruído de até 10% na picagem (*picking*). Em detalhe, zoom de parte do gráfico mostrando os desvios.

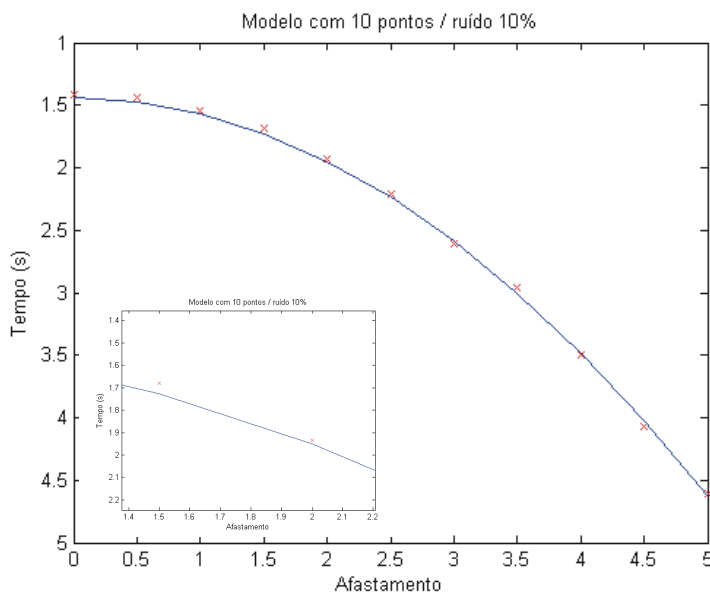


Figura 43: : Evento sintético com 10 pontos utilizado para testes do gradiente descendente. Curva azul: original sem ruído; pontos vermelhos: original com ruído de até 10% na picagem (*picking*). Em detalhe, zoom de parte do gráfico mostrando os desvios.

Tabela 7: Resultados de simulações para Eq. 52.

	t_0 (s) / erro (%)	V_{RMS} (km/s) / erro (%)
Valor exato	1,2	2,8
200 pontos / sem ruído	1,2 / 0	2,8 / 0
200 pontos / até 1% ruído	1,2 / 0	2,801/ -0,024
200 pontos / até 10% ruído	1,199/ 0,054	2,799/ 0,038
80 pontos / sem ruído	1,2 / 0	2,8 / 0
80 pontos / até 1% ruído	1,199 / 0,0057	2,799/ 0,009
80 pontos / até 10% ruído	1,206 / -0,51	2,814 / -0,529
10 pontos / sem ruído	1,2 / 0	2,8 / 0
10 pontos / até 1% ruído	1,202 / -0,2	2,801 / -0,029
10 pontos / até 10% ruído	1,201 / -0,076	2,791 / 0,303

A Tabela 7 mostra os resultados obtidos com o método do gradiente descendente aplicado à equação isotrópica (Eq. 52) nos seis modelos sintéticos utilizados como dados de entrada. O erro relativo novamente calculado pela Eq. 47 onde o valor de referência é o valor especificado na primeira linha da tabela (Valor exato). Observa-se a robustez e eficiência do método que mesmo com apenas 10 pontos de entrada e desvio de até 10%, obteve excelente resultados com erro máximo de 0,53% para V_{RMS} e 0,51% para t_0 .

Para todas as simulações da Tabela 7, o tempo de computação foi desprezível, chegando no máximo a aproximadamente 1 segundo para o Matlab[®] 7 sendo executado em uma máquina tipo PC Pentium[®] 4, CPU de 1,7MHz e 512Mb de memória RAM.

Na Tabela 8, podem ser analisados os resultados para o gradiente descendente aplicados à Eq. 55 nos nove conjuntos de pontos de entrada propostos. Novamente observa-se que para o primeiro parâmetro t_0 , o erro pode ser considerado insignificante para qualquer uma das simulações (erro máximo de 1,83%). O segundo parâmetro (V_{RMS}), gerou um erro máximo de 3,13% também considerado um resultado muito bom. O método começou a se tornar impreciso para o terceiro parâmetro (η) quando simulado com dez pontos e um erro de até 10% na marcação dos eventos.

As simulações para esta equação também foram realizadas a partir do mesmo equipamento computacional, e o tempo de execução para os cálculos do gradiente descendente utilizando esta equação duraram em média 30 segundos, tempo que pode ser explicado pelo cálculo dos gradientes da Eq. 55 serem significativamente mais complexos que da Eq. 52.

Tabela 8: Resultados de simulações para Eq. 55.

	t_0 (s) / erro (%)	V_{RMS} (km/s) / erro (%)	η / erro (%)
Valor exato	1,2	2,8	0,2
200 pontos / sem ruído	1,2 / 0	2,8 / 0	0,2 / 0
200 pontos / até 1% ruído	1,2 / 0	2,802 / -0,08	0,198 / 0,75
200 pontos / até 10% ruído	1,202 / -0,168	2,828 / -1,03	0,181 / 9,25
80 pontos / sem ruído	1,2 / 0	2,8 / 0	0,2 / 0
80 pontos / até 1% ruído	1,2 / 0	2,8 / 0	0,199 / 0,07
80 pontos / até 10% ruído	1,2004 / -0,032	2,82 / -0,73	0,186 / 7,02
10 pontos / sem ruído	1,2 / 0	2,799 / 0,01	0,2 / 0
10 pontos / até 1% ruído	1,199 / 0,013	2,789 / 0,38	0,208 / -4,37
10 pontos / até 10% ruído	1,198 / 0,121	2,887 / -3,13	0,135 / 32,11

Da mesma forma, a Tabela 9 mostra os resultados para o gradiente descendente aplicado à Eq. 57, com resultados bastante satisfatórios e erros obtidos aproximados aos da Tabela 8, que aumentam quando ocorre a situação onde existem poucos pontos como dados de entrada, e eles ainda possuem um erro de posição (Figura 43). Para este caso, o tempo de simulação também foi em média 30 segundos, e o erro máximo encontrado foi de 57,3% para o caso de 10 amostras com erro de 10% na posição dos pontos no parâmetro η , ou S (Eq. 57).

Tabela 9: Resultados de simulações para Eq. 57.

	t_0 (s) / erro (%)	V_{RMS} (km/s) / erro (%)	η / erro (%)
Valor exato	1,2	2,8	0,2
200 pontos / sem ruído	1,2 / 0	2,8 / 0	0,2 / 0
200 pontos / até 1% ruído	1,204 / -0,4	2,801 / -0,035	0,194 / 2,65
200 pontos / até 10% ruído	1,203 / -0,233	2,859 / -2,13	0,181 / 9,65
80 pontos / sem ruído	1,2 / 0	2,8 / 0	0,2 / 0
80 pontos / até 1% ruído	1,198 / 0,092	2,772 / 0,98	0,214 / -6,8
80 pontos / até 10% ruído	1,210 / -0,867	2,774 / 0,93	0,23 / -15,2
10 pontos / sem ruído	1,2 / 0	2,8 / 0	0,2 / 0
10 pontos / até 1% ruído	1,201 / -0,083	2,769 / 1,1	0,223 / -11,5
10 pontos / até 10% ruído	1,222 / -1,83	3,093 / -10,46	0,085 / 57,3

De acordo com os resultados deste primeiro experimento, observa-se que a Eq. 55 (Alkhalifah & Tsvankin, 1995) obteve erros menores na estimativa dos parâmetros utilizando o método do gradiente descendente que a Eq. 57 (Castle, 1994). Como os eventos utilizados nas simulações são criados a partir das próprias equações, conclui-se que, principalmente, com o aumento da complexidade da equação que se deseja estimar e também com o aumento da quantidade de ruído nos dados de entrada do método (eventos), o método torna-se mais impreciso.

4.3.2. Dados sintéticos - sismograma

Um outro exemplo realizado foi o de utilizar um sismograma sintético onde os dados são conhecidos para verificar como o método se comporta e como seria um procedimento para utilizar o gradiente descendente em dados reais. Para isso será utilizado o modelo sintético já utilizado no Capítulo 3, com o sismograma em mãos, será mostrado o procedimento para obter os pontos (x, t) necessários como dados de entrada para o gradiente descendente.

O ponto de partida para este exercício é um sismograma gerado no *array* (Figura 44) com as propriedades e características do meio definidas na Figura 16 e Tabela 1 com $d_1 = 1000m$ (águas ultra-profundas), $d_2 = 1500m$ (camada muito espessa de folhelhos), $\varepsilon = 0,1$ e $\delta = -0,1$, ($\eta = 0,25$). Este sismograma será utilizado

para realizar a análise de velocidades para meios anisotrópicos com o gradiente descendente.

Os eventos sísmicos do sismograma da Figura 44 devem ser selecionados para entrada no método do gradiente descendente, podendo ser utilizado qualquer programa de seleção de eventos para este procedimento. Neste caso específico, uma rotina foi desenvolvida no Matlab[®] para obter um conjunto de pontos a partir da marcação dos eventos nos gráficos com o *mouse* (Figura 45). A partir da obtenção dos pontos (100 pares de pontos (x, t) para este exemplo), é feita a entrada dos dados no algoritmo e os resultados para a Eq. 55 e a Eq. 57 (as duas equações para modelos VTI), comparados com o modelo teórico, podem ser observado na Tabela 10.

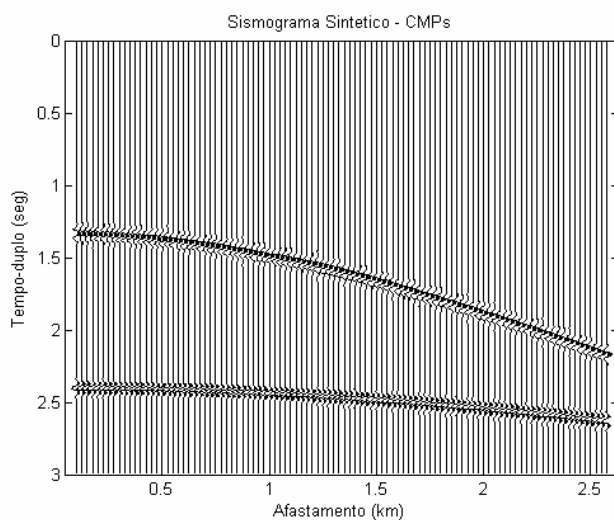


Figura 44: Sismograma sintético utilizado no teste.

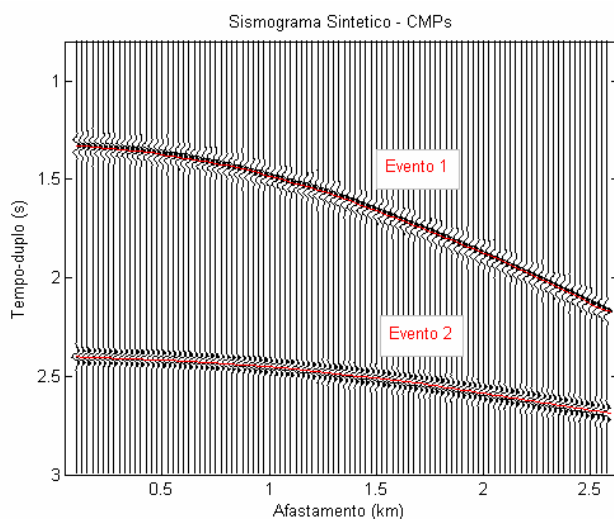


Figura 45: Seleção de eventos (curvas em vermelho) para entrada do gradiente descendente.

Pode-se observar na Tabela 10 que o primeiro evento obteve resultados muito próximos aos resultados teóricos para as duas equações. Porém, no evento 2, a Eq. 57 não conseguiu prever muito bem o parâmetro de anisotropia. Mais experimentos de processamento de dados anisotrópicos utilizando esta equação serão realizados no Capítulo 5.

Tabela 10: Tabela de resultados da estimativa do evento marcado na Figura 45. *Valor calculado utilizando a equação de Dix (Dix, 1955) para modelo isotrópico.

	t_0 (s)	V_{RMS} (km/s)	η
Evento 1			
Teórico	1,333	1,5	0
Estimado - Eq. 55	1,333	1,54	-0,03
Estimado - Eq. 57	1,329	1,491	0,01
Evento 2			
Teórico	2,405	2,177*	0,25
Estimado - Eq. 55	2,402	2,029	0,28
Estimado - Eq. 57	2,404	2,109	0,01

4.4. Exemplo com dado real

O último exercício do método será a utilização de um dado real. Foi escolhida uma pequena linha 2D chamada “Watson Rise” existente no banco de dados do programa ProMAX[®] como um dado exemplo ou tutorial, já com um processamento realizado proveniente do fabricante do programa.

Para tornar a comparação mais próxima possível do resultado de processamento já existente, o gradiente descendente será realizado nas mesmas posições onde a análise de velocidade convencional foi realizada (Figura 46) e a equação utilizada para o gradiente descendente é a equação hiperbólica ou isotrópica da Eq. 52.

Os CMPs da Figura 46 foram tratados para que os eventos importantes ficassem visíveis a ponto de serem selecionados para entrada do gradiente descendente (Figura 47). Para obter a imagem da Figura 46, foi aplicado um filtro

silenciador para a onda direta (*first break mute*) e o dado foi visualizado com um filtro passa-banda cuja janela é 13-40Hz.

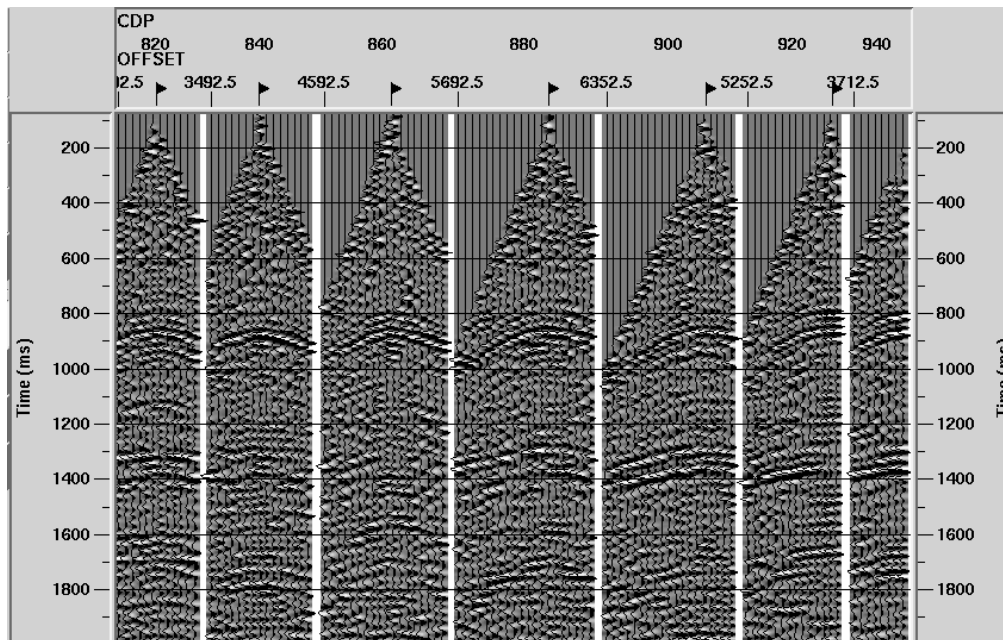


Figura 46: CMPs utilizados na análise de velocidade via gradiente descendente

A marcação dos eventos foi realizada no ProMAX[®], sendo escolhidos todos os eventos que possuem comportamento aproximadamente hiperbólico e se repetem nos CMPs adjacentes. Na média, foram selecionados 10 eventos por CMP como pode ser observado na Figura 47, com a quantidade máxima de amostras (cobertura de CMPs) de 20 amostras. Após salvos, os pontos (x, t) foram exportados do ProMAX[®] para o Matlab[®] para servirem de entrada do gradiente descendente. Os pontos obtidos como saída deste método (t_0, V_{RMS}) foram trazidos de volta para o ProMAX[®] onde foram comparadas as funções de velocidade RMS obtida pelos dois métodos (Figura 48 para a função proveniente do processamento anterior, e Figura 49 para o gradiente descendente), e os dados empilhados com ambas as funções (Figura 50).

A maior dificuldade na marcação dos eventos ocorreu nos eventos mais rasos (com profundidade de até 0,6s), devido ao filtro utilizado para silenciar a onda direta. Uma tentativa de marcar os eventos sem aplicar este filtro também foi feita, porém os sinais superpostos dificultaram a escolha dos eventos.

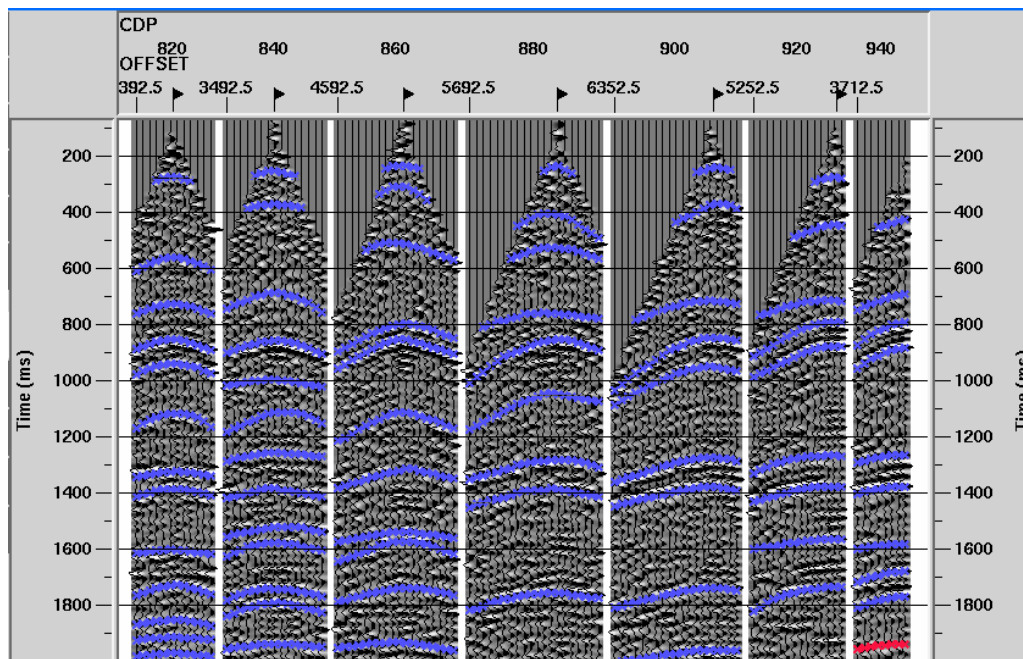


Figura 47: Marcação dos eventos nos CMPs para entrada do método do gradiente descendente.

Para as funções de velocidade RMS obtidas, comparando a Figura 48 com a Figura 49, observa-se que tendência lateral principal da velocidade RMS foi mantida (região entre 0,8s e 1,2s), porém observa-se um grande decréscimo da velocidade em aproximadamente 1,7s e entre os CMPs 815 e 870. Esta mudança na velocidade não pode ser considerada melhor ou pior, porque não se sabe nada sobre a real função de velocidade, pois a função da Figura 48 é resultado da interpretação de outra pessoa, como pode ser considerado qualquer resultado em processamento de dados sísmicos. Uma possível razão para esta diferença pode ter sido a escolha de um evento que seja uma múltipla de um evento real, ou seja, um sinal que ficou refletindo várias vezes dentro de algumas interfaces e foi captado pelos geofones ou hidrofones neste tempo. Uma outra possível razão para a diferença pode ser que o intérprete que realizou a análise de *semblance* não acreditou na inversão de velocidade e preferiu manter a função suave sem muitas variações laterais.

Apesar das diferenças, as duas funções são equivalentes no que diz respeito ao empilhamento, como pode ser observado na Figura 50. As duas sessões empilhadas são bastante equivalentes e todos os eventos principais podem ser observados tanto no processamento com a função de velocidade da Figura 48 quanto a função da Figura 49.

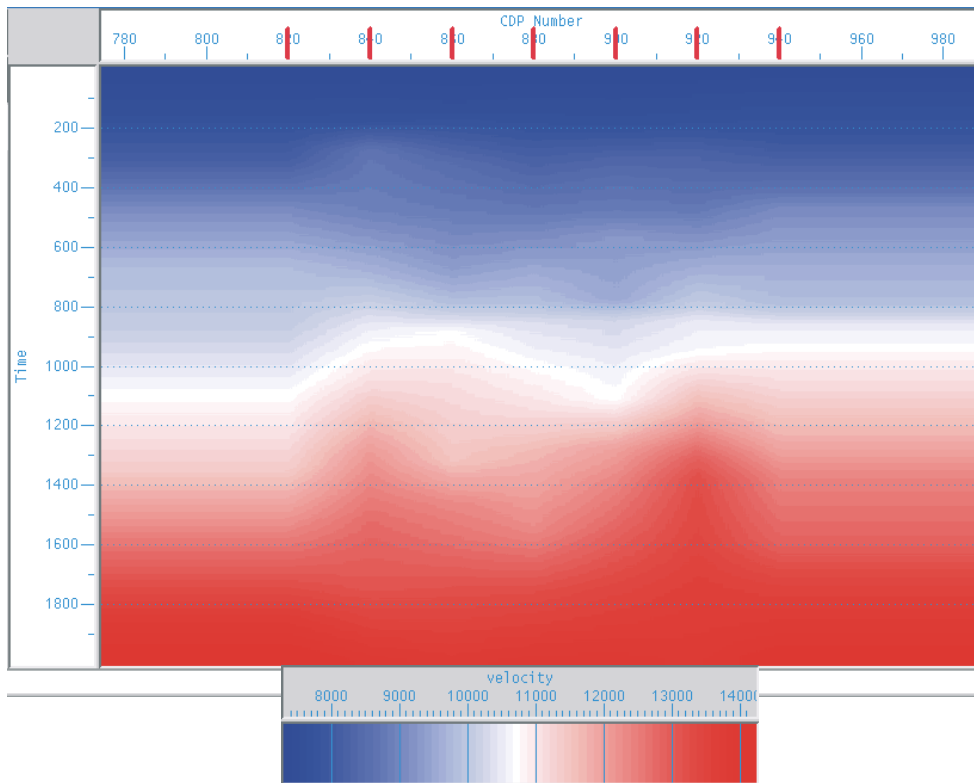


Figura 48: Função de velocidade RMS obtida no dado a partir do método convencional de análise de velocidade.

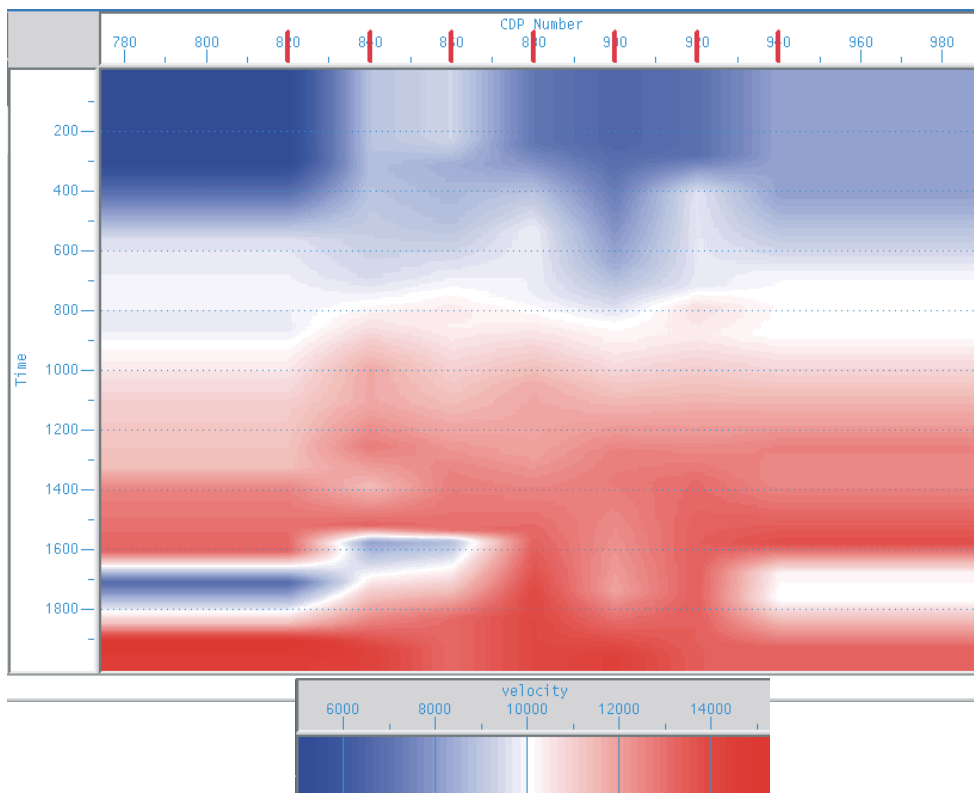


Figura 49: Função de velocidade RMS obtida no dado a partir do método gradiente descendente.

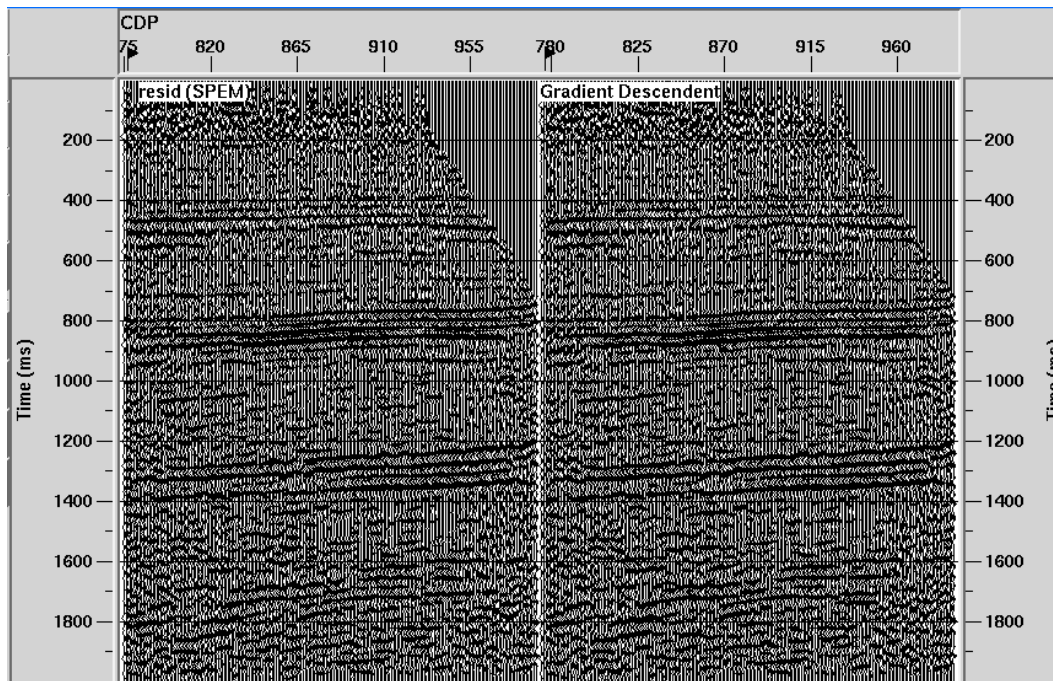


Figura 50: Comparação das sessões sísmicas empilhadas com a função de velocidade obtida pelo método convencional (esquerda) e pelo gradiente descendente (direita).

4.5. Conclusões e comentários

Um método de minimização de funções chamado gradiente descendente foi utilizado para adaptar diferentes equações de tempo de trânsito e realizar análise de velocidade a partir destas equações. As principais vantagens observadas no método foi a rapidez, eficácia e precisão dos resultados obtidos, inclusive quando existem poucas informações de entrada (x, t) . No caso de um processamento de dados anisotrópicos, uma outra vantagem do método é a obtenção de todos os parâmetros da equações de uma única vez $(t_0, V_{NMO}$ e $\eta)$, enquanto o método mais utilizado para este procedimento (*semblance*) só permite a solução de no máximo dois parâmetros ao mesmo tempo $(t_0$ e $V_{NMO})$.

Como aspecto negativo pode-se observar a dificuldade de calcular os gradientes da função de custo para equações de NMO mais complexas, ou não lineares. É preciso sempre ter em mãos um dado ou evento de referência para aplicar ao método para verificar se todas as derivadas estão corretas e foram definidas e aplicadas corretamente no algoritmo.

Por ser um método adaptativo, teoricamente ele pode ser utilizado com qualquer equação de tempo de trânsito, como funções que consideram anisotropia do tipo TTI, ortorrômbica ou monoclinica ou para funções que consideram onda convertida. Entretanto ainda é necessário testar a utilização de equações mais complexas, determinação otimizada das variáveis utilizadas pelo método e verificar a sua convergência para cada caso.

Pela sua própria característica, é um método que pode ser utilizado para estimar parâmetros de anisotropia ou parâmetros de processamento utilizando dados sísmicos.

Nesta fase do trabalho foram utilizadas três equações de tempo de trânsito, e os resultados confirmam que o método é promissor pela fácil manipulação e ajuste das suas variáveis, e a rápida convergência obtida nos poucos testes. O método foi utilizado em modelos sintéticos (eventos e sismogramas) e em um dado real. O tempo gasto pelo interprete é simplesmente o de marcar os eventos importantes no dado sísmico. Após este procedimento, a simulação é rápida (alguns segundos), e depende da quantidade de dados de entrada (número de pontos) e da equação que se deseja estimar com o método. O maior tempo de simulação registrado nos dados deste Capítulo foi de aproximadamente 30 segundos. Todos os resultados foram considerados bons, tornando o gradiente descendente uma alternativa muito interessante para a análise de velocidade em dados sísmicos.