



UNIVERSIDADE FEDERAL DO PARÁ
INSTITUTO DE GEOCIÊNCIAS
PROGRAMA DE PÓS-GRADUAÇÃO EM GEOFÍSICA

DISSERTAÇÃO DE MESTRADO

MIGRAÇÃO FD 3D EM PROFUNDIDADE USANDO
APROXIMAÇÃO DE PADÉ COMPLEXA

CARLOS ALEXANDRE NASCIMENTO DA COSTA

BELÉM

2011

CARLOS ALEXANDRE NASCIMENTO DA COSTA

**MIGRAÇÃO FD 3D EM PROFUNDIDADE USANDO
APROXIMAÇÃO DE PADÉ COMPLEXA**

Dissertação apresentada ao Programa de Pós-Graduação em Geofísica do Instituto de Geociências da Universidade Federal do Pará, para a obtenção do título de Mestre em Geofísica.

Área de Concentração: Métodos Sísmicos

Orientador: Prof. Dr. Jessé Carvalho Costa

BELÉM

2011

Dados Internacionais de Catalogação-na-Publicação(CIP)
Biblioteca Geólogo Raimundo Montenegro Garcia de Montalvão

C837m Costa, Carlos Alexandre Nascimento da
Migração FD 3D em profundidade usando aproximação de
Padé complexa / Carlos Alexandre Nascimento da Costa; Orien-
tador: Jessé Carvalho Costa - 2011.
75 fl.: il.

Dissertação (Mestrado em Geofísica) – Programa de Pós-
Graduação em Geofísica, Instituto de Geociências, Universidade
Federal do Pará, Belém, 2011.

1. Geofísica aplicada. 2. Diferenças finitas. 3. Migração
FD. 4. Aproximação de Padé complexa. 5. Método iterativo. I.
Costa, Jessé Carvalho, *orient.* II. Universidade Federal do Pará
III. Título.

CDD 22° ed.: 551

CARLOS ALEXANDRE NASCIMENTO DA COSTA

**MIGRAÇÃO FD 3D EM PROFUNDIDADE USANDO APROXIMAÇÃO DE PADÉ
COMPLEXA**

Dissertação apresentada ao Programa de Pós-Graduação em Geofísica do Instituto de Geociências da Universidade Federal do Pará, para a obtenção do título de Mestre em Geofísica.

Data da aprovação: 21 / 06 / 2011

BANCA EXAMINADORA:



Prof. Jesse Carvalho Costa (Orientador)
Doutor em Geofísica
CPGF/Universidade Federal do Pará



Prof.^a Maria Amélia Novais Schleicher
Doutora em Matemática Aplicada
IMECC/Universidade Estadual de Campinas



Prof. Cícero Roberto Teixeira Régis
Doutor em Geofísica
CPGF/Universidade Federal do Pará

CARLOS ALEXANDRE NASCIMENTO DA COSTA

**MIGRAÇÃO FD 3D EM PROFUNDIDADE USANDO APROXIMAÇÃO DE PADÉ
COMPLEXA**

Dissertação apresentada ao Programa de Pós-Graduação em Geofísica do Instituto de Geociências da Universidade Federal do Pará, para a obtenção do título de Mestre em Geofísica.

Data da aprovação: 21 / 06 / 2011

BANCA EXAMINADORA:

Prof. Jessé Carvalho Costa (Orientador)
Doutor em Geofísica
CPGF/Universidade Federal do Pará

Prof.^a Maria Amélia Novais Schleicher
Doutora em Matemática Aplicada
IMECC/Universidade Estadual de Campinas

Prof. Cícero Roberto Teixeira Régis
Doutor em Geofísica
CPGF/Universidade Federal do Pará

À minha querida mãe Maria Frâncisca.

AGRADECIMENTOS

Nenhum trabalho dessa natureza se completa pelo esforço ou talento de uma única pessoa. Ao longo do tempo que dedicamos a sua realização, diversas pessoas e entidades vão em maior ou menor grau contribuindo para que tal realização possa se dar da melhor forma possível. Abaixo, deixo um agradecimento sincero a algumas pessoas de quem me lembro agora. Se por ventura deixei alguém de fora dessa lista, peço desculpas, na certeza de que tal falha em nada diminui a importância de sua ajuda.

À minha mãe, Maria Francisca, por sempre se esforçar em me dar o melhor em estudo e pela confiança. Também aos meus irmãos e irmãs por sempre estarem prontos para me ajudar nos momentos difíceis e por entender minha “ausência” perante a família. Em especial as minhas irmãs Ana Maria, Maria de Lourdes e Soraya Maciel.

À minha amada, Eliane Epífane, que soube entender e aceitar pacientemente minhas “ausências” quando o tempo para finalizar este trabalho parecia que não ia ser suficiente.

Um especial agradecimento a todos os integrantes atuais e os que por lá passaram de um dos maiores responsáveis pelo meu sucesso, o Espaço Cultural “Nossa Biblioteca”. Em especial à Madalena Westerveld e Raimundo Oliveira.

Agradeço o Prof. Jessé Costa, não apenas por ter me apresentado ao mundo da geofísica, mas também, por sua orientação, paciência e por várias ideias que guiaram meu trabalho. Da mesma maneira, não posso deixar de reconhecer a importância de inúmeras e frutíferas discussões com o Prof. Francisco de Assis, que muito me ajudaram em diversas fases desse trabalho.

Agradeço a Itamara Campos por contribuir na realização desse trabalho.

Aos membros da banca examinadora Dr.^a Maria Amélia Novais Schleicher e Dr. Cícero Roberto Teixeira Régis pelas recomendações sugeridas.

Aos meus amigos e amigas que fiz na UFPA, pelo carinho, confiança e por me proporcionar momentos de felicidades. Em especial à Alessandra Braga, Hayane Maciel, Luís Tadaiesky, Márcio Moreira, Rodrigo Santos, Jonathas Maciel e Débora Rodrigues.

À Sra. Benilde Lopes e Suelen Buzaglo, a minha gratidão.

Agradeço ao CNPq, FAPESP, CAPES e PETROBRAS pelo apoio financeiro.

“Jamais considere seus estudos como uma obrigação, mas como uma oportunidade invejável para aprender a conhecer a influência libertadora da beleza do reino do espírito, para seu próprio prazer pessoal e para proveito da comunidade à qual seu futuro trabalho pertencer.”
(A. Einstein; Aos estudantes de Princeton, EUA)

RESUMO

A implementação convencional do método de migração por diferenças finitas 3D, usa a técnica de *splitting inline* e *crossline* para melhorar a eficiência computacional deste algoritmo. Esta abordagem torna o algoritmo eficiente computacionalmente, porém cria anisotropia numérica. Esta anisotropia numérica por sua vez, pode levar a falsos posicionamentos de refletores inclinados, especialmente refletores com grandes ângulos de mergulho. Neste trabalho, com o objetivo de evitar o surgimento da anisotropia numérica, implementamos o operador de extrapolação do campo de onda para baixo sem usar a técnica *splitting inline* e *crossline* no domínio frequência-espço via método de diferenças finitas implícito, usando a aproximação de Padé complexa. Comparamos a performance do algoritmo iterativo Bi-gradiente conjugado estabilizado (Bi-CGSTAB) com o *multifrontal massively parallel solver* (MUMPS) para resolver o sistema linear oriundo do método de migração por diferenças finitas. Verifica-se que usando a expansão de Padé complexa ao invés da expansão de Padé real, o algoritmo iterativo Bi-CGSTAB fica mais eficiente computacionalmente, ou seja, a expansão de Padé complexa atua como um condicionador para este algoritmo iterativo. Como consequência, o algoritmo iterativo Bi-CGSTAB é bem mais eficiente computacionalmente que o MUMPS para resolver o sistema linear quando usado apenas um termo da expansão de Padé complexa. Para aproximações de grandes ângulos, métodos diretos são necessários. Para validar e avaliar as propriedades desses algoritmos de migração, usamos o modelo de sal SEG/EAGE para calcular a sua resposta ao impulso.

Palavras-chave : Geofísica aplicada. Diferenças finitas. Migração por diferenças finitas. Aproximação de Padé complexa. Método iterativo.

ABSTRACT

Conventional implementations of 3D finite-difference (FD) migration use splitting techniques to accelerate performance and save computational cost. However, such techniques are plagued with numerical anisotropy that jeopardizes correct positioning of dipping reflectors in the directions not used for the splitted operators. We implement 3D downward continuation migration without splitting in the space coordinates using a complex Padé approximation and implicit finite differences. In this way, the numerical anisotropy is eliminated at the expense of a computationally more intensive solution of a large banded linear system. We compare the performance of the iterative stabilized biconjugate gradient (Bi-CGSTAB) and the multifrontal massively parallel direct solver (MUMPS). It turns out that the use of the complex Padé approximation provides an effective preconditioner for the Bi-CGSTAB, reducing the number of iterations relative to the real Padé expansion of the FD operator. As a consequence, the iterative Bi-CGSTAB method is more efficient than the direct MUMPS method when solving for a single term in the Padé expansion. For wide angle approximations direct methods are required. These algorithms are validated and the properties evaluated computing the migration impulse response in the SEG/EAGE salt model.

Keywords: Applied Geophysics. Finite difference. FD migration. Complex Padé approximation. Iterative method

LISTA DE ILUSTRAÇÕES

Figura 2.1	Linha de corte no plano complexo rotacionada por um ângulo θ	24
Figura 3.1	A esquerda: frequência limite, f_L , pelo ângulo de rotação da linha de corte em graus. A direita: número de iterações do algoritmo Bi-CGSTAB para cada frequência para $c = 1500m/s$ e $h = 10m$	33
Figura 3.2	A esquerda: frequência limite, f_L , pelo ângulo de rotação da linha de corte em graus. A direita: número de iterações do algoritmo Bi-CGSTAB para cada frequência para $c = 1500m/s$ e $h = 20m$	34
Figura 3.3	A esquerda: frequência limite, f_L , pelo ângulo de rotação da linha de corte em graus. A direita: número de iterações do algoritmo Bi-CGSTAB para cada frequência para $c = 4500m/s$ e $h = 10m$	34
Figura 3.4	A esquerda: número de iterações do algoritmo Bi-CGSTAB para cada frequência para $c = 1500m/s$, $h = 10m$ e dois termos da série de Padé complexa. A direita: número de iterações do algoritmo Bi-CGSTAB para cada frequência para $c = 1500m/s$, $h = 10m$ e um termo da série de Padé complexa	36
Figura 3.5	Número de iterações do algoritmo Bi-CGSTAB para cada frequência para $c = 1500m/s$ e $h = 10m$. A esquerda: Comparação entre FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte 25° com e sem PML nas bordas do plano horizontal. A direita: Comparação entre FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte 25° com otimização e sem otimização.	36
Figura 4.1	Seções da resposta ao impulso 3D no plano paralelo ao plano x_1x_2 em $x_3 = 700$ m, coluna da esquerda, e $x_3 = 1050$ m, coluna da direita. Em cima: FD usando <i>splitting inline</i> e <i>crossline</i> . No meio: FD sem <i>splitting inline</i> e <i>crossline</i> . Embaixo: RTM.	40
Figura 4.2	Seção horizontal em $x_3 = 1900$ m da resposta ao impulso 3D do algoritmo de migração FD iterativo usando 1 termo da série de Padé complexa com ângulo de rotação da linha de corte 45°	41
Figura 4.3	Seção horizontal em $x_3 = 1900$ m da resposta ao impulso 3D do algoritmo de migração FD iterativo-PML usando 1 termo da série de Padé complexa com ângulo de rotação da linha de corte 45°	41

Figura 4.4	Seção vertical da resposta ao impulso 3D no plano paralelo a direção <i>inline</i> em $x_2 = 3380$ m. Em cima: FD iterativo com 1 termo da série de Padé complexa com ângulo da linha de corte $\theta = 45^\circ$. No meio: FD iterativo com 1 termo da série de Padé real. Embaixo: RTM	43
Figura 4.5	Seção vertical da resposta ao impulso 3D no plano paralelo a direção <i>inline</i> em $x_2 = 3380$ m. Em cima: FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte $\theta = 45^\circ$. No meio: FD iterativo com 2 termo da série de Padé complexa com ângulo de rotação da linha de corte 45° . Embaixo: RTM	44
Figura 4.6	Seção vertical da resposta ao impulso 3D no plano paralelo a direção <i>inline</i> em $x_2 = 3380$ m. Em cima: FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte $\theta = 45^\circ$. No meio: FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte $\theta = 25^\circ$. Embaixo: RTM	45
Figura 4.7	Seção vertical da resposta ao impulso 3D no plano paralelo a direção <i>inline</i> em $x_2 = 3380$ m. Em cima: FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte $\theta = 25^\circ$. No meio: FD iterativo com 1 termo da série de Padé complexa otimizado com ângulo de rotação da linha de corte $\theta = 25^\circ$. Embaixo: RTM	47
Figura 5.1	Seções verticais do modelo de sal SEG/EAGE em planos paralelos a direção <i>inline</i> . Em cima: em $x_2 = 1690$ m. No meio: em $x_2 = 3380$ m. Embaixo: em $x_2 = 4160$ m.	49
Figura 5.2	Seções verticais do modelo de sal SEG/EAGE em planos paralelos a direção <i>crossline</i> . Em cima: em $x_1 = 1690$ m. No meio: em $x_1 = 3380$ m. Embaixo: em $x_1 = 4160$ m.	50
Figura 5.3	Seções horizontais do modelo de sal SEG/EAGE em planos paralelos ao plano x_1x_2 . Em cima na esquerda: em $x_3 = 550$ m. Em cima na direita: em $x_3 = 1050$ m. Embaixo na esquerda: em $x_3 = 1350$ m. Embaixo na direita: em $x_3 = 1550$ m.	51

Figura 5.4 Seções da resposta ao impulso 3D no plano paralelo ao plano x_1x_3 em $x_2 = 1690$ m. Em cima na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 45^\circ$. Em cima na direita: FD-MUMPS com 1 termo da série de Padé complexa e $\theta = 45^\circ$. No meio na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 25^\circ$. No meio na direita: FD iterativo com 1 termo da série de Padé complexa otimizado e $\theta = 25^\circ$. Embaixo na esquerda: FD iterativo com 2 termos da série de Padé complexa e $\theta = 45^\circ$. Embaixo na direita: RTM. 52

Figura 5.5 Seções da resposta ao impulso 3D no plano paralelo ao plano x_2x_3 em $x_1 = 1690$ m. Em cima na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 45^\circ$. Em cima na direita: FD-MUMPS com 1 termo da série de Padé complexa e $\theta = 45^\circ$. No meio na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 25^\circ$. No meio na direita: FD iterativo com 1 termo da série de Padé complexa otimizado e $\theta = 25^\circ$. Embaixo na esquerda: FD iterativo com 2 termos da série de Padé complexa e $\theta = 45^\circ$. Embaixo na direita: RTM. 53

Figura 5.6 Seções da resposta ao impulso 3D no plano paralelo ao plano x_1x_3 em $x_2 = 3380$ m. Em cima na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 45^\circ$. Em cima na direita: FD-MUMPS com 1 termo da série de Padé complexa e $\theta = 45^\circ$. No meio na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 25^\circ$. No meio na direita: FD iterativo com 1 termo da série de Padé complexa otimizado e $\theta = 25^\circ$. Embaixo na esquerda: FD iterativo com 2 termos da série de Padé complexa e $\theta = 45^\circ$. Embaixo na direita: RTM. 54

Figura 5.7 Seções da resposta ao impulso 3D no plano paralelo ao plano x_2x_3 em $x_1 = 3380$ m. Em cima na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 45^\circ$. Em cima na direita: FD-MUMPS com 1 termo da série de Padé complexa e $\theta = 45^\circ$. No meio na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 25^\circ$. No meio na direita: FD iterativo com 1 termo da série de Padé complexa otimizado e $\theta = 25^\circ$. Embaixo na esquerda: FD iterativo com 2 termos da série de Padé complexa e $\theta = 45^\circ$. Embaixo na direita: RTM. 55

Figura 5.8	Seções da resposta ao impulso 3D no plano paralelo ao plano x_1x_3 em $x_2 = 4160$ m. Em cima na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 45^\circ$. Em cima na direita: FD-MUMPS com 1 termo da série de Padé complexa e $\theta = 45^\circ$. No meio na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 25^\circ$. No meio na direita: FD iterativo com 1 termo da série de Padé complexa otimizado e $\theta = 25^\circ$. Embaixo na esquerda: FD iterativo com 2 termos da série de Padé complexa e $\theta = 45^\circ$. Embaixo na direita: RTM.	56
Figura 5.14	Tempo de CPU do algoritmo de migração FD para calcular a resposta ao impulso usando o modelo de sal da SEG/EAGE	56
Figura 5.9	Seções da resposta ao impulso 3D no plano paralelo ao plano x_2x_3 em $x_1 = 4160$ m. Em cima na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 45^\circ$. Em cima na direita: FD-MUMPS com 1 termo da série de Padé complexa e $\theta = 45^\circ$. No meio na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 25^\circ$. No meio na direita: FD iterativo com 1 termo da série de Padé complexa otimizado e $\theta = 25^\circ$. Embaixo na esquerda: FD iterativo com 2 termos da série de Padé complexa e $\theta = 45^\circ$. Embaixo na direita: RTM.	57
Figura 5.10	Seções da resposta ao impulso 3D no plano paralelo ao plano x_1x_2 em $x_3 = 550$ m. Em cima na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 45^\circ$. Em cima na direita: FD-MUMPS com 1 termo da série de Padé complexa e $\theta = 45^\circ$. No meio na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 25^\circ$. No meio na direita: FD iterativo com 1 termo da série de Padé complexa otimizado e $\theta = 25^\circ$. Embaixo na esquerda: FD iterativo com 2 termos da série de Padé complexa e $\theta = 45^\circ$. Embaixo na direita: RTM.	58
Figura 5.11	Seções da resposta ao impulso 3D no plano paralelo ao plano x_1x_2 em $x_3 = 1050$ m. Em cima na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 45^\circ$. Em cima na direita: FD-MUMPS com 1 termo da série de Padé complexa e $\theta = 45^\circ$. No meio na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 25^\circ$. No meio na direita: FD iterativo com 1 termo da série de Padé complexa otimizado e $\theta = 25^\circ$. Embaixo na esquerda: FD iterativo com 2 termos da série de Padé complexa e $\theta = 45^\circ$. Embaixo na direita: RTM.	59

Figura 5.12 Seções da resposta ao impulso 3D no plano paralelo ao plano x_1x_2 em $x_3 = 1350$ m. Em cima na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 45^\circ$. Em cima na direita: FD-MUMPS com 1 termo da série de Padé complexa e $\theta = 45^\circ$. No meio na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 25^\circ$. No meio na direita: FD iterativo com 1 termo da série de Padé complexa otimizado e $\theta = 25^\circ$. Embaixo na esquerda: FD iterativo com 2 termos da série de Padé complexa e $\theta = 45^\circ$. Embaixo na direita: RTM. 60

Figura 5.13 Seções da resposta ao impulso 3D no plano paralelo ao plano x_1x_2 em $x_3 = 1550$ m. Em cima na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 45^\circ$. Em cima na direita: FD-MUMPS com 1 termo da série de Padé complexa e $\theta = 45^\circ$. No meio na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 25^\circ$. No meio na direita: FD iterativo com 1 termo da série de Padé complexa otimizado e $\theta = 25^\circ$. Embaixo na esquerda: FD iterativo com 2 termos da série de Padé complexa e $\theta = 45^\circ$. Embaixo na direita: RTM. 61

SUMÁRIO

1	INTRODUÇÃO	16
2	MIGRAÇÃO COM A EQUAÇÃO DA ONDA ACÚSTICA UNIDIRE- CIONAL	19
2.1	EQUAÇÃO DA ONDA ACÚSTICA UNIDIRECIONAL	19
2.2	MIGRAÇÃO PELO MÉTODO POR DESLOCAMENTO DE FASE	21
2.3	MÉTODO DE MIGRAÇÃO FD IMPLÍCITO	23
3	MÉTODOS DE RESOLUÇÃO PARA SISTEMAS LINEARES	28
3.1	MÉTODOS DIRETOS VERSUS MÉTODOS ITERATIVOS	28
3.2	MÉTODOS ITERATIVOS	29
3.2.1	MÉTODOS ITERATIVOS BÁSICOS	29
3.2.2	MÉTODOS ITERATIVOS DO TIPO KRYLOV	29
3.2.2.1	MÉTODO ITERATIVO CG	29
3.2.2.2	MÉTODO ITERATIVO Bi-CGSTAB	30
3.2.2.3	ANALISANDO A CONVERGÊNCIA DO Bi-CGSTAB	31
4	ALGORITMO DE MIGRAÇÃO FD ITERATIVO	37
4.1	A ANISOTROPIA NUMÉRICA	37
4.2	IMPLEMENTANDO O ALGORITMO DE MIGRAÇÃO FD ITERATIVO .	38
4.2.1	REFLEXÕES NAS BORDAS DO PLANO HORIZONTAL	39

4.2.2	ABERTURA ANGULAR	42
5	RESULTADOS NUMÉRICOS USANDO MODELO DE SAL	48
6	CONCLUSÕES	62
	REFERÊNCIAS	65
	APÊNDICE	69
	APÊNDICE A – MÉTODOS ITERATIVOS	70

1 INTRODUÇÃO

Migração em profundidade usando o método de diferenças finitas (FD)¹ implícito para resolver a equação de onda (CLAERBOUT, 1985), é um dos mais adequados quando existem fortes variações laterais de velocidade no meio onde os dados sísmicos estão sendo migrados. Para fazermos migração em profundidade, é necessário extrapolar para baixo o campo de onda. No processo de extrapolação do campo de onda para baixo em um meio acústico, podemos usar a equação da onda acústica unidirecional (CLAERBOUT, 1985; BIONDI, 2006). Quando esta equação é em três dimensões (3D), usar o método FD implícito para resolvê-la, gera grandes sistemas lineares que precisam ser resolvidos para cada passo da extrapolação do campo de onda para baixo. Usar os métodos de solução direta (ISERLES, 1996) para resolver esses sistemas lineares, tem um alto custo computacional. Uma maneira eficiente computacionalmente de implementar este operador de extrapolação do campo de onda 3D, é usar a técnica de *splitting* nas direções *inline* e *crossline* (BROWN, 1983). Utilizar esta técnica de *splitting inline* e *crossline*, torna o algoritmo de extrapolação do campo para baixo muito eficiente computacionalmente, mas produz anisotropia numérica (LI, 1991; RISTOW; RÜHL, 1997; BIONDI, 2006). Devido a esta anisotropia numérica, os refletores inclinados podem ser erroneamente posicionados pelo processo de migração, principalmente quando estes refletores estão posicionado em um ângulo azimutal de 45° (LI, 1991; RISTOW; RÜHL, 1997; BIONDI, 2006).

Existem na literatura vários trabalhos com objetivo de resolver esta anisotropia numérica. Li (1991), propôs um operador de correção, que deve ser aplicado em alguns passos da extrapolação do campo de onda para baixo para corrigir os efeitos devido a anisotropia numérica. Ristow e Rühl (1997), propuseram aproximar o operador 3D em 3, 4 ou 6 direções para corrigir a anisotropia numérica. Os trabalhos de Li (1991), Ristow e Rühl (1997), além do objetivo de corrigir a anisotropia numérica, tem como objetivo preservar a eficiência computacional devido ao uso da técnica de *splitting inline* e *crossline*. Uma outra ideia para corrigir a anisotropia numérica e reduzir o custo computacional, é não usar a técnica de *splitting inline* e *crossline* e usar métodos iterativos para resolver os sistemas lineares. Cole (1989) usando essa ideia, investigou os métodos iterativos de Jacobi, Gauss-Seidel e sobre-relaxamento sucessivo (SOR)² (ISERLES, 1996) para resolver os sistemas lineares. Cole (1989) verificou que para baixas frequências esses métodos são muito lentos para convergir e em alguns casos podem até mesmo não convergir. Nichols

¹Resolvemos usar a sigla FD para denotar diferenças finitas devido ao nome em inglês *finite-difference*

²Resolvemos usar a sigla SOR para denotar sobre-relaxamento sucessivo devido ao nome em inglês *successive over-relaxation*

(1991) investigou o uso do gradiente conjugado (CG)³ (HESTENES; STIEFEL, 1952) para resolver os sistemas lineares. Apesar de ter convergência garantida, para matriz simétrica e positiva-definida, o CG também sofre do problema de baixa velocidade de convergência para baixas frequências, no entanto, Nichols (1991) ressalta que uma maneira de aumentar a velocidade de convergência do CG para baixas frequências, é evitar nos dados de entrada a propagação de energias relacionada aos grandes ângulos e de modos evanescentes.

Levando em consideração a observação de Nichols (1991), é interessante notar que Amazonas et al. (2007) usando a aproximação de Padé complexa (MILINAZZO; ZALA; BROOKE, 1997) no operador de extrapolação do campo de onda para baixo, que é essencialmente um meio de atenuar as energias de grandes ângulos e de modos evanescentes, melhorou a qualidade das imagens migradas. Outro fato que é importante ressaltar no trabalho de Nichols (1991), é a transformação da matriz dos coeficientes do sistema linear em uma matriz simétrica e positiva-definida, pois o CG não é adequado para resolver sistemas lineares complexos, e em alguns casos ele pode até mesmo divergir (ERLANGGA, 2005). Nesse processo de transformar a matriz de coeficientes em uma matriz simétrica e positiva-definida, a velocidade de convergência do CG fica mais lenta (ERLANGGA, 2005).

Nos últimos anos, com os avanços computacionais e com o desenvolvimento de métodos iterativos eficientes para resolver sistemas lineares complexos e não simétrico, tal como o Bi-gradiente conjugado estabilizado (Bi-CGSTAB)⁴ (VAN DE VORST, 1992) e o método de redução de dimensão induzida (IDR)⁵ (SONNEVELD; GIJZEN, 2008; SONNEVELD, 2010; GIJZEN; SONNEVELD, 2010; ONOUE; FUJINO; NAKASHIMA, 2009), impulsionaram o uso de métodos iterativos em geofísica de prospecção para resolver a equação de onda 3D, como podemos ver nos trabalhos de Neklyudov, Silvestrov e Tcheverda (2010), Plessix (2009, 2007), Riyanti et al. (2006).

No presente trabalho, implementamos a equação da onda acústica unidirecional 3D sem *splitting inline* e *crossline* no domínio frequência-espço, através do método FD implícito usando a expansão de Padé complexa para aproximar a raiz quadrada da equação da onda acústica unidirecional e o método iterativo Bi-CGSTAB para resolver os sistemas lineares. Dessa maneira, obtemos um algoritmo de migração 3D iterativo, que resolve o problema da anisotropia numérica, converge mais rápido que os algoritmos de migração iterativo 3D de Cole (1989), Nichols (1991) e é mais eficiente computacionalmente do que usando os métodos de solução direta para resolver o sistema linear quando usamos aproximação de primeira ordem da raiz quadrada da equação da onda acústica unidire-

³Resolvemos usar a sigla CG para denotar gradiente conjugado devido ao nome em inglês *conjugate gradient*

⁴Resolvemos usar a sigla Bi-CGSTAB para denotar Bi-gradiente conjugado estabilizado devido ao nome em inglês *Bi-conjugate gradient stabilized*

⁵Resolvemos usar a sigla IDR para denotar redução de dimensão induzida devido ao nome em inglês *induced dimension reduction*

cional. No decorrer desse trabalho, chamaremos esse algoritmo apenas de algoritmo de migração FD iterativo. Para validar o algoritmo de migração FD iterativo, comparamos a sua resposta ao impulso com as respostas ao impulso do algoritmo de migração reversa no tempo (RTM)⁶ e do algoritmo de migração FD usando o a versão serial do pacote *multifrontal massively parallel solver* (MUMPS) (AMESTOY et al., 2001, 2006) para resolver o sistema linear.

Esta dissertação está organizada em seis capítulos. Além desta introdução têm-se os seguintes capítulos:

No Capítulo 2, discutimos alguns conceitos de migração 3D em profundidade usando a equação da onda acústica unidirecional no domínio frequência-espaco e fazemos uma revisão do método de migração por deslocamento de fase e FD.

No Capítulo 3, discutimos os métodos de resolução para sistemas lineares em especial os métodos iterativos. Fazemos uma breve revisão do método iterativo Bi-CGSTAB e analisamos sua convergência em um modelos de velocidade homogêneo.

No Capítulo 4, discutimos a implementação do algoritmo de migração FD iterativo e analisamos a sua performance em modelos de velocidade homogêneo.

No Capítulo 5, mostramos os resultados numéricos obtido com o algoritmo de migração FD iterativo no modelo de sal SEG/EAGE.

No Capítulo 6, apresentamos conclusões desta dissertação, bem como algumas perspectivas de trabalhos futuros.

⁶Resolvemos usar a sigla RTM para migração reversa no tempo devido ao nome em inglês *reverse-time migration*

2 MIGRAÇÃO COM A EQUAÇÃO DA ONDA ACÚSTICA UNIDIRECIONAL

Este capítulo é dedicado a fazer uma revisão do método de migração FD implícito. Antes de revisarmos tal método, inicialmente discutiremos uma maneira de se obter a equação da onda acústica unidirecional no domínio frequência-espço e o método de migração por deslocamento de fase para introduzirmos o conceito de extrapolação do campo de onda para baixo.

2.1 EQUAÇÃO DA ONDA ACÚSTICA UNIDIRECIONAL

Com o objetivo de obter a equação da onda acústica unidirecional no domínio frequência-espço, começamos nossa discussão definindo a equação da onda acústica 3D em um domínio infinito $\Omega_0 \subseteq \mathbb{R}^3$ da seguinte maneira:

$$\nabla^2 P(\mathbf{x}, t) - \frac{1}{c(\mathbf{x})^2} \frac{\partial^2 P(\mathbf{x}, t)}{\partial t^2} = 0, \quad (2.1)$$

onde $\mathbf{x} = (x_1, x_2, x_3) \in \Omega_0$ denota um vetor no espço cartesiano com x_1 e x_2 as coordenadas laterais ao longo das direções *inline* e *crossline*, respectivamente, e x_3 a profundidade, $\nabla^2 = \left(\frac{\partial^2}{\partial x_1^2} + \frac{\partial^2}{\partial x_2^2} + \frac{\partial^2}{\partial x_3^2} \right)$ é o operador laplaciano em coordenadas cartesianas, t é a variável temporal, $c(\mathbf{x})$ é a velocidade de propagação da onda que varia com a posição e $P(\mathbf{x}, t)$ é o campo de pressão.

Usamos neste trabalho a seguinte convenção do par de transformada de Fourier em relação a variável temporal t :

$$\hat{P}(\mathbf{x}, \omega) = \int_{-\infty}^{+\infty} P(\mathbf{x}, t) e^{i\omega t} dt, \quad (2.2)$$

$$P(\mathbf{x}, t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \hat{P}(\mathbf{x}, \omega) e^{-i\omega t} d\omega, \quad (2.3)$$

onde $i = \sqrt{-1}$ é a unidade imaginária e $\omega \in \mathbb{R}$ é a frequência angular. A equação (2.2) denota a transformada de Fourier direta e a equação (2.3) denota a transformada de Fourier inversa.

Usando a transformada de Fourier inversa na equação (2.1), obtemos a equação da

onda acústica no domínio frequência-espço ($\omega - \Omega_0$) dada por

$$\nabla^2 \hat{P}(\mathbf{x}; \omega) + \frac{\omega^2}{c(\mathbf{x})^2} \hat{P}(\mathbf{x}, \omega) = 0. \quad (2.4)$$

A fim de obter a equação da onda acústica unidirecional, iremos considerar válida para descrever problemas de propagação de onda acústica no domínio frequência-espço ($\omega - \Omega_0$) a equação abaixo

$$\left\{ \left[\partial_3 + \frac{i\omega}{c(\mathbf{x})} \sqrt{1 + \frac{c^2(\mathbf{x})}{\omega^2} (\partial_1^2 + \partial_2^2)} \right] \left[\partial_3 - \frac{i\omega}{c(\mathbf{x})} \sqrt{1 + \frac{c^2(\mathbf{x})}{\omega^2} (\partial_1^2 + \partial_2^2)} \right] \right\} \hat{P}(\mathbf{x}, \omega) = 0, \quad (2.5)$$

com $\partial_j \equiv \frac{\partial_j}{\partial x_j}$, $j = 1, 2, 3$.

A partir da equação (2.5) obtemos as equações:

$$\left[\partial_3 - \frac{i\omega}{c(\mathbf{x})} \sqrt{1 + \frac{c^2(\mathbf{x})}{\omega^2} (\partial_1^2 + \partial_2^2)} \right] \hat{P}(\mathbf{x}, \omega) = 0 \quad (2.6)$$

e

$$\left[\partial_3 + \frac{i\omega}{c(\mathbf{x})} \sqrt{1 + \frac{c^2(\mathbf{x})}{\omega^2} (\partial_1^2 + \partial_2^2)} \right] \hat{P}(\mathbf{x}, \omega) = 0. \quad (2.7)$$

Se tentarmos a partir da equação (2.5) derivar a equação (2.4), obteremos:

$$\left\{ \partial_3^2 + \frac{\omega^2}{c(\mathbf{x})^2} \left[1 + \frac{c^2(\mathbf{x})}{\omega^2} (\partial_1^2 + \partial_2^2) \right] \right\} \hat{P}(\mathbf{x}, \omega) + \left\{ \left[\frac{\omega}{c(\mathbf{x})} \sqrt{1 + \frac{c^2(\mathbf{x})}{\omega^2} (\partial_1^2 + \partial_2^2)} \right] i\partial_3 - i\partial_3 \left[\frac{\omega}{c(\mathbf{x})} \sqrt{1 + \frac{c^2(\mathbf{x})}{\omega^2} (\partial_1^2 + \partial_2^2)} \right] \right\} \hat{P}(\mathbf{x}, \omega) = 0. \quad (2.8)$$

Observamos que para derivarmos a equação (2.4) da equação (2.8), o comutador presente na última equação entre os operadores $\frac{\omega}{c(\mathbf{x})} \sqrt{1 + \frac{c^2(\mathbf{x})}{\omega^2} (\partial_1^2 + \partial_2^2)}$ e ∂_3 tem que ser nulo, ou seja, eles têm que comutar. Uma maneira de comutar estes operadores é considerar a velocidade de propagação da onda constante. Neste caso, a equação (2.5) representa exatamente a equação (2.4). Sendo assim, podemos dizer que as equações (2.6) e (2.7) são equações exatas que governam a propagação da onda acústica para cima e para baixo respectivamente, no domínio frequência-espço com velocidade constante. Por outro lado, temos interesse em meios onde a velocidade varie com a posição \mathbf{x} . Nesse caso, anular o comutador da equação (2.8) é um problema que pode ser tratado usando a teoria dos operadores pseudodiferenciais (OP 'T ROOT; STOLK, 2010). Mas aqui, iremos ignorar esse comutador e considerar as equações (2.6) e (2.7) como sendo uma boa aproximação para

descrever problemas de onda acústica se propagando em uma única direção preferencial no domínio frequência-espço com a velocidade de propagação da onda acústica variando com a posição x .

As equações (2.6) e (2.7), descrevem precisamente o tempo de trânsito das ondas, porém elas precisam ser modificadas para descrever as amplitudes (ZHANG; ZHANG; BLEISTEIN, 2005; VIVAS; PESTANA, 2010; OP 'T ROOT; STOLK, 2010). Neste trabalho não temos preocupação de corrigir este problema de amplitude.

As equações (2.6) e (2.7) são válidas em um domínio espacial Ω_0 infinito. Porém, como iremos resolvê-las computacionalmente, precisamos limitar esse domínio, ou seja, temos que truncar o domínio Ω_0 . Sendo assim, a partir de agora temos interesse em encontrar a solução das equações (2.6) e (2.7) apenas no domínio limitado $\Omega_b \subset \Omega_0$. Para resolver estas equações usaremos o campo de pressão registrado na superfície de aquisição sísmica x_3^0 como uma condição de contorno.

A partir desse ponto, trabalhamos apenas com a equação que governa a propagação do campo de onda para baixo, equação (2.7), pois é esta equação que usaremos para extrapolar o campo de onda para baixo.

2.2 MIGRAÇÃO PELO MÉTODO POR DESLOCAMENTO DE FASE

Começamos esta seção ressaltando que a convenção do par de transformadas de Fourier direta e inversa sobre as coordenadas laterais x_1, x_2 que usamos neste trabalho são dadas respectivamente por

$$\hat{P}(k_1, k_2, x_3, t) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} P(x_1, x_2, x_3, t) e^{-i(k_1 x_1 + k_2 x_2)} dx_1 dx_2, \quad (2.9)$$

$$P(x_1, x_2, x_3, t) = \frac{1}{(2\pi)^2} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \hat{P}(k_1, k_2, x_3, t) e^{i(k_1 x_1 + k_2 x_2)} dk_1 dk_2, \quad (2.10)$$

onde k_1 e k_2 são os números de onda nas direções *inline* e *crossline*, respectivamente.

Considerando um meio onde a velocidade de propagação da onda acústica seja constante, usando a transformada de Fourier inversa, equação (2.10), na equação (2.4) e fazendo a fatoração como foi feito para obter as equações (2.6) e (2.7), obtemos como

resultado a equação:

$$\partial_3 \hat{P}(\mathbf{K}, x_3, \omega) = -i \frac{\omega}{c} \sqrt{1 - \frac{c^2}{\omega^2} (k_1^2 + k_2^2)} \hat{P}(\mathbf{K}, x_3, \omega), \quad (2.11)$$

onde $\mathbf{K} = (k_1, k_2)$ é o número de onda no plano horizontal.

Como falamos na seção anterior, usaremos o campo de pressão registrado na superfície de aquisição sísmica x_3^0 , P_0 , como uma condição de contorno para a equação (2.11), ou seja, $P(x_1, x_2, x_3 = x_3^0, t) = P_0$. Sendo assim, resolvendo a equação (2.11) e usando a transformada de Fourier de P_0 como uma condição de contorno, obtemos a seguinte solução:

$$\hat{P}(\mathbf{K}, x_3, \omega) = \hat{P}_0 \exp\left(-i \frac{\omega}{c} k_3 (x_3 - x_3^0)\right), \quad (2.12)$$

onde $k_3 \equiv \sqrt{1 - \frac{c^2}{\omega^2} K^2}$.

Como podemos ver na equação acima, a exponencial $\exp(-i \frac{\omega}{c} k_3 (x_3 - x_3^0))$, apenas desloca o campo \hat{P}_0 na direção x_3 e o sinal negativo indica que esse deslocamento é para baixo, ou seja, estamos extrapolando o campo para baixo.

O objetivo de extrapolar o campo para baixo, é para formar uma imagem da superfície refletora que refletiu a onda registrada em x_3^0 . Considerando que a onda que foi registrada em x_3^0 foi refletida no instante $t = 0$, para posicionarmos o campo P_0 coincidindo com a superfície refletora, basta retropropagar este campo no tempo, ou seja, ir no sentido contrário no eixo temporal e avaliá-lo em $t = 0$. Esta é uma condição necessária para formar a imagem de tal superfície. Por exemplo, para um par fonte-receptor, extrapolando para baixo os seus campos de ondas e fazendo a correlação cruzada entres eles avaliado em $t = 0$, obtemos a imagem onde ocorreu a reflexão. Esta correlação cruzada avaliada em $t = 0$, é uma condição de imagem.

Pelo que foi discutido acima, podemos dizer que migração em profundidade é dividida em dois processos: (i) a continuação do campo de onda para baixo, que pode ser entendido como uma transformação do campo de onda medido em x_3^0 em um campo coincidente com a superfície refletora e (ii) a condição de imagem, que é a operação responsável pela formação da imagem da superfície refletora na posição x .

Este método de migração através do deslocamento do campo de onda via rotação da fase da solução da equação de onda acústica em uma única direção preferencial, é conhecida como método por deslocamento de fase (GAZDAG, 1978).

Para usar o método de migração por deslocamento de fase em um meio verticalmente heterogêneo, ou seja, com a velocidade variando apenas com a profundidade, $c(x_3)$,

dividimos o eixo x_3 em L camadas de velocidade constante de espessura Δx_3 de modo que $l\Delta x_3 \equiv x_3^l$ denota a camada l , tal que $x_3^{l-1} \leq x_3^l < x_3^{l+1}$ onde x_3^{l-1} e x_3^{l+1} são os limites superior e inferior da camada l , respectivamente, com $l = 1, 2, \dots, L - 1$. Adotando essa estratégia, definimos as camadas da seguinte maneira:

$$x_3^l = x_3^{l-1} + \Delta x_3 \quad (2.13)$$

$$x_3 = \sum_{l=1}^{L-1} x_3^0 + x_3^l. \quad (2.14)$$

Usando a equação (2.13) na equação (2.12), obtemos a equação:

$$\hat{P}(x_3^l, \mathbf{K}, \omega) = \hat{P}(x_3^{l-1}, \mathbf{K}) \exp\left(-i \frac{\omega}{c} k_3 \Delta x_3\right), \quad (2.15)$$

que extrapola o campo para a camada l , usando o campo da camada $l - 1$.

Antes de concluir essa seção, queremos ressaltar que para derivar o método de migração por deslocamento de fase, nenhuma aproximação foi necessária, desta maneira, a princípio esperamos que esse método seja preciso para migrar refletores com mergulhos de até 90° .

2.3 MÉTODO DE MIGRAÇÃO FD IMPLÍCITO

O método de migração FD implícito é um dos mais adequados quando existem fortes variações laterais de velocidade no meio onde os dados sísmicos estão sendo migrados. Para resolver a equação (2.7) usando o método FD implícito, precisamos expandir a raiz quadrada que aparece nesta equação. Essa expansão é necessária, pois neste caso não podemos adotar a estratégia de usar a transformada de Fourier nas direções *inline* e *crossline* como foi feito para o método por deslocamento de fase, dado que a transformada de Fourier neste caso envolve uma convolução. Neste trabalho, optamos por usar a expansão de Padé complexa para aproximar raiz quadrada da equação (2.7) (AMAZONAS et al., 2007; MILINAZZO; ZALA; BROOKE, 1997), pois além de atenuarmos os modos evanescentes, ainda mantemos a ordem dos operadores de derivadas. No Capítulo 3 analisamos a necessidade de evitar energias de modos evanescentes. Sendo assim, temos a seguinte expansão da raiz quadrada da equação (2.7) em série de Padé complexa:

$$\sqrt{1 + \frac{c^2(\mathbf{x})}{\omega^2} (\partial_1^2 + \partial_2^2)} \approx \left[C_0 + \sum_{n=1}^N \frac{A_n \frac{c^2(\mathbf{x})}{\omega^2} D}{1 + B_n \frac{c^2(\mathbf{x})}{\omega^2} D} \right], \quad (2.16)$$

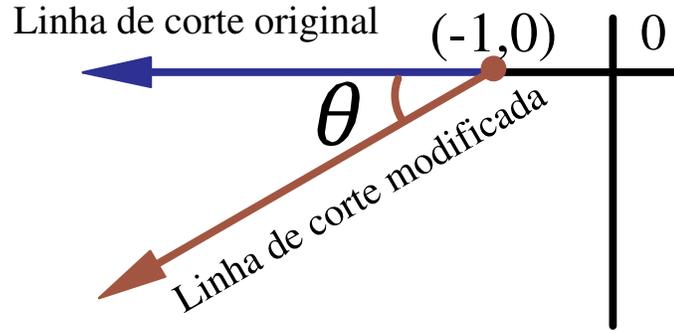


Figura 2.1: Linha de corte no plano complexo rotacionada por um ângulo θ .

com $D \equiv \partial_1 + \partial_2$. Onde N indica a ordem da aproximação, C_0 , A_n e B_n são os coeficientes de Padé complexo dados por:

$$\begin{aligned} C_0 &= e^{i\theta/2} \left[1 + \sum_{n=1}^N \frac{a_n (e^{-i\theta} - 1)}{1 + b_n (e^{-i\theta} - 1)} \right] \\ A_n &= \frac{a_n e^{-i\theta/2}}{[1 + b_n (e^{-i\theta} - 1)]^2} \\ B_n &= \frac{b_n e^{-i\theta}}{1 + b_n (e^{-i\theta} - 1)}. \end{aligned} \quad (2.17)$$

Onde θ é o ângulo de rotação da linha de corte modificada no plano complexo, mostrado na Figura (2.1), doravante chamaremos apenas ângulo de rotação da linha de corte, a_n e b_n são os coeficientes de Padé real dados por:

$$\begin{aligned} a_n &= \frac{2}{2N+1} \text{sen}^2 \left(\frac{n\pi}{2N+1} \right) \\ b_n &= \cos^2 \left(\frac{n\pi}{2N+1} \right). \end{aligned} \quad (2.18)$$

Utilizando esta aproximação da raiz quadrada da equação (2.7) temos como resultado

$$\partial_3 \hat{P}(\mathbf{x}, \omega) \approx -i \frac{\omega}{c(\mathbf{x})} \left[C_0 + \sum_{n=1}^N \frac{A_n \frac{c^2(\mathbf{x})}{\omega^2} D}{1 + B_n \frac{c^2(\mathbf{x})}{\omega^2} D} \right] \hat{P}(\mathbf{x}, \omega). \quad (2.19)$$

Uma maneira mais eficiente computacionalmente de implementar a equação anterior é usar a técnica de *splitting* (CLAERBOUT, 1985). Usando esta técnica, para cada passo x_3^l da extrapolação do campo para baixo, temos que resolver $N + 1$ equações. Primeiro resolvemos:

$$\partial_3 \hat{P}(\mathbf{x}, \omega) = -i \frac{\omega C_0}{c(\mathbf{x})} \hat{P}(\mathbf{x}, \omega) \quad (2.20)$$

e depois para $n = 1, \dots, N$ equações

$$\partial_3 \hat{P}(\mathbf{x}, \omega) = -i \frac{\omega}{c(\mathbf{x})} \left[\frac{A_n \frac{c^2(\mathbf{x}) D}{\omega^2}}{1 + B_n \frac{c^2(\mathbf{x}) D}{\omega^2}} \right] \hat{P}(\mathbf{x}, \omega), \quad (2.21)$$

onde a solução de cada equação é usada como condição de contorno para a próxima equação até o total de $N + 1$ equações para cada passo x_3^l da extrapolação do campo. A equação (2.20), que chamaremos de termo analítico, é fácil para resolver, no entanto as N últimas equações (2.21), que chamaremos de termo FD, é preciso aproximar as derivadas por operadores de diferenças finitas. Para facilitar nosso trabalho, faremos essa aproximação por operadores de diferenças finitas usando apenas uma das N equações.

Como vamos aproximar as derivadas das equações (2.21) através de operadores de diferenças finitas, precisamos discretizar o domínio Ω_b . O domínio discretizado será denotado por Ω_b^h . Discretizando os eixos x_1, x_2 e x_3 em tamanho h_1, h_2 e h_3 , respectivamente. Obtemos uma malha cúbica através dos pontos $(x_{1p}, x_{2q}, x_3^l) = (ph_1, qh_2, lh_3)$ com $N_{x_1}^i \leq p \leq N_{x_1}^f, N_{x_2}^i \leq q \leq N_{x_2}^f$ e $N_{x_3}^i \leq l \leq N_{x_3}^f$, onde $N_{x_j}^i$ e $N_{x_j}^f, j = 1, 2, 3$ são números inteiros que indicam o ponto inicial e final em cada eixo na malha respectivamente.

Usando a notação $P(ph_1, qh_2, lh_3, \omega) = P_{pq}^l$ e o método Crank-Nicholson, podemos discretizar a equação (2.21) obtendo a seguinte equação discretizada no domínio Ω_b^h

$$\left\{ \delta_{sp} \delta_{rq} + \eta_{pq}^l \left[\frac{1}{h_1^2} (\delta_{sp-1} \delta_{rq} - 2\delta_{sp} \delta_{rq} + \delta_{sp+1} \delta_{rq}) + \frac{1}{h_2^2} (\delta_{sp} \delta_{rq-1} - 2\delta_{sp} \delta_{rq} + \delta_{sp} \delta_{rq+1}) \right] \right\} P_{sr}^{l+1} = \left\{ \delta_{sp} \delta_{rq} + \eta_{pq}^{*l} \left[\frac{1}{h_1^2} (\delta_{sp-1} \delta_{rq} - 2\delta_{sp} \delta_{rq} + \delta_{sp+1} \delta_{rq}) + \frac{1}{h_2^2} (\delta_{sp} \delta_{rq-1} - 2\delta_{sp} \delta_{rq} + \delta_{sp} \delta_{rq+1}) \right] \right\} P_{sr}^l, \quad (2.22)$$

onde δ_{sp}, δ_{rq} são as deltas de Kronecker, η_{pq}^l e $\eta_{p,q}^{*l}$ são dados por

$$\eta_{pq}^l = \left(B_n + i \frac{\omega h_3}{2c_{pq}^l} A_n \right) \left(\frac{c_{pq}^l}{\omega} \right)^2 \quad (2.23)$$

e

$$\eta_{pq}^{*l} = \left(B_n - i \frac{\omega h_3}{2c_{pq}^l} A_n \right) \left(\frac{c_{pq}^l}{\omega} \right)^2. \quad (2.24)$$

Da equação (2.22), vemos que para calcular a função $P(ph_1, qh_2, lh_3, \omega)$ para cada nível l , é necessário resolvê-la para todos os pontos da malha do plano horizontal. Por outro lado, para calcular esta função em um ponto da malha no plano horizontal, são necessários quatro outros pontos vizinhos, exceto para os pontos das bordas da malha.

Então para cada ponto da malha do plano horizontal obtemos uma equação, logo temos um sistema linear de $M = ((N_{x_1}^f - N_{x_1}^i + 1)(N_{x_2}^f - N_{x_2}^i + 1))$ equações com o mesmo número de incógnitas, exceto para os pontos das bordas da malha, para cada nível l . Para obter uma representação matricial para esse problema discretizado, é preciso escolher uma ordenação para os pontos da malha. Neste trabalho escolhemos a seguinte ordenação para representar a equação (2.22) de forma matricial: os pontos da malha do plano horizontal são percorridos coluna por coluna da esquerda para a direita de cima para baixo para cada nível l tomando os devidos cuidados com as bordas dessa malha. Dessa maneira, obtemos a seguinte representação matricial para a equação (2.22):

$$(\mathbf{I} + \alpha \mathbf{D})\mathbf{P}^{l+1} = (\mathbf{I} + \alpha^* \mathbf{D})\mathbf{P}^l, \quad (2.25)$$

onde \mathbf{I} é a matriz identidade, α , α^* representam os valores do mapeamento de η_{pq}^l e η_{pq}^{*l} , respectivamente, para cada ponto da malha, \mathbf{D} representa a matriz de derivadas ao longo das coordenadas do plano horizontal e tem dimensão M , $\mathbf{P}^l \in \mathbb{C}^M$ e $\mathbf{P}^{l+1} \in \mathbb{C}^M$ representam os vetores campos de pressão na malha do plano horizontal em nível de profundidades sucessivas.

Para facilitar nosso trabalho, vamos reescrever a equação (2.25) da seguinte maneira:

$$\mathbf{A}\mathbf{P}^{l+1} = \mathbf{A}^*\mathbf{P}^l, \quad (2.26)$$

onde $\mathbf{A} \equiv \mathbf{I} + \alpha \mathbf{D} \in \mathbb{C}^{M \times M}$ e $\mathbf{A}^* \equiv \mathbf{I} + \alpha^* \mathbf{D} \in \mathbb{C}^{M \times M}$ representam as matrizes de coeficientes complexos.

Nas matrizes da equação (2.26) para cada uma das M linhas, exceto para as linhas que representam as bordas da malha, temos apenas 5 elementos diferentes de zero. Para a malha que usamos em nossos testes, tanto no modelo homogêneo quanto no heterogêneo, temos um M da ordem 457 mil linhas, o que torna essas matrizes esparsas.

Para cada nível da extrapolação do campo para baixo, temos que resolver M equações do sistema linear para cada frequência. O custo computacional para usar o método de solução direta é alto, tanto no sentido de tempo de CPU quanto na quantidade de memória que deve ser alocada. Uma maneira de reduzir este custo computacional, é fazer *splitting* nas direções *inline* (x_1) e *crossline* (x_2) da equação (2.22), resultando no par de equações

$$\begin{aligned} & \left\{ \delta_{sp} \delta_{rq} + \eta_{pq}^l \left[\frac{1}{h_1^2} (\delta_{sp-1} \delta_{rq} - 2\delta_{sp} \delta_{rq} + \delta_{sp+1} \delta_{rq}) \right] \right\} P_{sr}^{l+1} = \\ & \left\{ \delta_{sp} \delta_{rq} + \eta_{pq}^{*l} \left[\frac{1}{h_1^2} (\delta_{sp-1} \delta_{rq} - 2\delta_{sp} \delta_{rq} + \delta_{sp+1} \delta_{rq}) \right] \right\} P_{sr}^l \end{aligned} \quad (2.27)$$

$$\left\{ \delta_{sp} \delta_{rq} + \eta_{pq}^l \left[\frac{1}{h_2^2} (\delta_{sp} \delta_{rq-1} - 2\delta_{sp} \delta_{rq} + \delta_{sp} \delta_{rq+1}) \right] \right\} P_{sr}^{l+1} = \left\{ \delta_{sp} \delta_{rq} + \eta_{pq}^{*l} \left[\frac{1}{h_2^2} (\delta_{sp} \delta_{rq-1} - 2\delta_{sp} \delta_{rq} + \delta_{sp} \delta_{rq+1}) \right] \right\} P_{sr}^l. \quad (2.28)$$

As equações (2.27) e (2.28) também podem ser representadas de forma matricial assim como foi feito para obter a forma matricial da equação (2.22).

Se usarmos as equações (2.27) e (2.28) em vez da equação (2.22) para extrapolar o campo para baixo, temos que resolvê-las sucessivamente para cada passo da extrapolação. Sendo assim, para cada passo dessa extrapolação temos que resolver para cada frequência um total de $M^s = ((N_{x_1}^f - N_{x_1}^i + 1) + (N_{x_2}^f - N_{x_2}^i + 1))$ equações lineares. Este método é conhecido como método de *splitting inline* e *crossline*. Ele é super eficiente do ponto de vista computacional, porém produz erros no processo de migração, como veremos no Capítulo 4.

Resolvendo sucessivamente para cada passo da extrapolação do campo para baixo a equação (2.20) e o sistema linear sem *splitting inline* e *crossline* equação (2.22) ou com *splitting inline* e *crossline* equações (2.27) e (2.28), encontramos um campo de pressão se propagando continuamente para baixo. Esta solução numérica da equação da onda acústica unidirecional através de aproximações das derivadas em operadores de diferenças finitas é conhecido como método de migração FD implícito. O método de migração FD implícito, como foi falado antes, não tem restrições em relação ao modelo de velocidade, porém como é necessário realizar uma aproximação da raiz quadrada da equação de onda unidirecional, ele tem dificuldade de migrar refletores com fortes mergulhos.

3 MÉTODOS DE RESOLUÇÃO PARA SISTEMAS LINEARES

Como foi visto no capítulo anterior, para fazer migração usando o método FD implícito é necessário resolver um grande número de sistemas lineares esparsos. Sendo assim, neste capítulo discutiremos alguns métodos de resolução para sistemas lineares. Em geral esses métodos são divididos em métodos diretos e métodos iterativos.

3.1 MÉTODOS DIRETOS VERSUS MÉTODOS ITERATIVOS

Métodos diretos são métodos baseados em eliminação Gaussiana (ERLANGGA, 2005). Eles calculam a solução exata de um sistema linear, a menos de erro de arredondamento, com um número finito e bem determinado de operações aritméticas. São adequados para resolver sistemas lineares de tamanhos moderados e densos. Como os sistemas lineares que precisamos resolver nesta dissertação são grandes e esparsos, equação (2.26), usar métodos diretos não é recomendado, pois no processo de eliminação Gaussiana o sistema linear perde sua característica de esparsidade, exigindo muita memória alocada e o tempo de CPU para encontrar a solução do sistema linear também é alto.

Métodos iterativos calculam a solução aproximada de um sistema linear, se beneficia da esparsidade da matriz, exigindo pouca memória alocada, porém tem baixa velocidade de convergência para encontrar a solução do sistema linear, dependendo do condicionamento da matriz, e em alguns casos pode divergir. Este problema de de baixa velocidade de convergência dos métodos iterativo pode ser contornado com um bom preconditionador para a matriz do sistema linear, ou seja, deixando a matriz original melhor condicionada. Para os sistemas lineares que precisamos resolver nesta dissertação, o uso dos métodos iterativos são mais adequados que os métodos diretos, pois a matriz do sistema linear (2.26) tem um grande número equações lineares e é esparso.

Pelo que foi exposto nos dois últimos parágrafos anteriores, o uso de métodos iterativos são mais adequado para o nosso problema, então deixamos o que restante deste capítulo para estudarmos alguns métodos iterativos e analisar suas características.

3.2 MÉTODOS ITERATIVOS

Nesta seção iremos introduzir alguns métodos de solução iterativo para resolver sistemas lineares. Em especial, daremos uma atenção maior para o método iterativo Bi-CGSTAB e estudaremos o comportamento de sua convergência para o sistema linear (2.26) em um modelo de velocidade homogêneo.

3.2.1 MÉTODOS ITERATIVOS BÁSICOS

Métodos iterativos básicos como de Jacobi, Gauss-Seidel e SOR (ISERLES, 1996), normalmente são muito lentos para convergir (dependendo do condicionamento da matriz de coeficientes) e em alguns casos (quando a matriz não é diagonalmente dominante) eles podem não convergir (COLE, 1989). Na seção A.1 do Apêndice A, discutimos como derivar de uma maneira intuitiva os métodos de Jacobi e Gauss-Seidel.

3.2.2 MÉTODOS ITERATIVOS DO TIPO KRYLOV

Métodos iterativos do tipo Krylov, são métodos de projeção para encontrar a solução aproximada de sistemas lineares em um subespaço específico, o subespaço de Krylov. Segue abaixo uma definição do subespaço de Krylov:

Definição 3.1 *Seja uma matriz $\mathbf{B} \in \mathbb{C}^{M \times M}$ não singular e dado um vetor $\mathbf{v} \neq \mathbf{0} \in \mathbb{C}^M$, o subespaço de Krylov $\mathcal{K}_l(\mathbf{B}, \mathbf{v})$ gerado de \mathbf{v} é dado por*

$$\mathcal{K}_l(\mathbf{B}, \mathbf{v}) \equiv \text{span} \{ \mathbf{v}, \mathbf{B}\mathbf{v}, \dots, \mathbf{B}^{l-1}\mathbf{v} \}. \quad (3.1)$$

Na seção A.2 do Apêndice A, introduzimos esta definição de maneira mais intuitiva.

MÉTODO ITERATIVO CG

O método iterativo CG (HESTENES; STIEFEL, 1952) é um método do tipo Krylov. Ele é muito eficiente para resolver sistemas lineares onde a matriz de coeficiente é simétrica e positiva-definida. Para este tipo de matriz, ele tem convergência garantida em um número finito de iterações. Veja na seção A.2 do Apêndice A, o algoritmo CG.

MÉTODO ITERATIVO Bi-CGSTAB

Como os elementos da matriz de coeficientes, do sistema linear que precisamos resolver (2.26), são complexos e para algumas combinações de parâmetros, como veremos mais tarde neste capítulo, ela não é diagonalmente dominante, optamos por usar o método iterativo Bi-CGSTAB para resolvê-lo. Pois ele é apropriado para resolver sistemas lineares onde os elementos da matriz de coeficiente são complexos.

O Bi-CGSTAB é um método iterativo do tipo Krylov para resolver sistemas lineares como

$$\mathbf{B}\mathbf{u} = \mathbf{b}, \quad (3.2)$$

onde $\mathbf{B} \in \mathbb{C}^{M \times M}$ é uma matriz não singular, $\mathbf{u} \in \mathbb{C}^M$ e $\mathbf{b} \in \mathbb{C}^M$.

Uma generalização do CG para resolver sistemas lineares como o (3.2) é o algoritmo BI-CG de Lanczos (1950). A ideia básica desse algoritmo é a construção de duas bases $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_l\} \subset \mathcal{K}_l(\mathbf{B}, \mathbf{r}_0)$ e $\{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_l\} \subset \mathcal{L}_l(\mathbf{B}^\dagger, \hat{\mathbf{r}}_0)$ de tal maneira que elas sejam bi-ortogonais. Onde $\mathcal{K}_l(\mathbf{B}, \mathbf{r}_0)$ e $\mathcal{L}_l(\mathbf{B}^\dagger, \hat{\mathbf{r}}_0)$ são subespaços de Krylov e \mathbf{B}^\dagger denota a matriz transposta conjugada de \mathbf{B} .

Essa condição de bi-ortogonalidade é necessária para calcular o α_l e β_l para cada iteração l como foi feito no CG (veja Apêndice A). Porém fazendo isso, estamos resolvendo dois sistemas lineares da forma (3.2). O $\mathbf{B}\mathbf{u} = \mathbf{b}$ e seu dual $\mathbf{B}^\dagger \hat{\mathbf{u}} = \hat{\mathbf{b}}$.

Devido as observações de Sonneveld (1987) temos o algoritmo CGS onde não precisamos ter que resolver o sistema $\mathbf{B}^\dagger \hat{\mathbf{u}} = \hat{\mathbf{b}}$, porém o comportamento da sua convergência é instável. Esse problema de instabilidade na convergência é resolvido por (VAN DE VORST, 1992) dando origem ao algoritmo Bi-CGSTAB.

Segue a baixo o algoritmo **Bi-CGSTAB**:

Algoritmo 3.1 (Bi-CGSTAB)

1. Inicialize com o vetor inicial \mathbf{u}_0 , o vetor conhecido \mathbf{b} e um vetor $\hat{\mathbf{r}}$ arbitrário.
2. Calcule o resíduo inicial $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{u}_0$
3. $\mathbf{p}_0 = \mathbf{r}_0$
4. Calcule a tolerância da aproximação e atribua o número máximo de iterações permitida
5. Início do laço

6. Faça até que o limite de tolerância seja atingido ou até o número máximo de iterações:

$$7. \alpha_l = \langle \mathbf{r}_l, \hat{\mathbf{r}} \rangle / \langle \mathbf{B}\mathbf{p}_l, \hat{\mathbf{r}} \rangle$$

$$8. \mathbf{s}_l = \mathbf{r}_l - \alpha_l \mathbf{B}\mathbf{p}_l$$

$$9. \omega_l = \langle \mathbf{B}\mathbf{s}_l, \mathbf{s}_l \rangle / \langle \mathbf{B}\mathbf{s}_l, \mathbf{B}\mathbf{s}_l \rangle$$

$$10. \mathbf{u}_{l+1} = \mathbf{u}_l + \alpha_l \mathbf{p}_l + \omega_l \mathbf{s}_l$$

$$11. \mathbf{r}_{l+1} = \mathbf{s}_l - \omega_l \mathbf{B}\mathbf{s}_l$$

$$12. \beta_l = \frac{\langle \mathbf{r}_{l+1}, \hat{\mathbf{r}} \rangle}{\langle \mathbf{r}_l, \hat{\mathbf{r}} \rangle} \times \alpha_l / \omega_l$$

$$13. \mathbf{p}_{l+1} = \mathbf{r}_{l+1} + \beta_l (\mathbf{p}_l - \omega_l \mathbf{B}\mathbf{p}_l)$$

14. Fim do Laço

ANALISANDO A CONVERGÊNCIA DO Bi-CGSTAB

Nesta seção estudaremos como se comporta a convergência do algoritmo Bi-CGSTAB na obtenção da solução aproximada do sistema linear (2.26) usando um modelo de velocidade homogêneo com uma malha regular. Denotamos o espaçamento entre os pontos por h , ou seja, $h_1 = h_2 = h_3 = h$. A precisão da solução do sistema linear calculado com o algoritmo iterativo Bi-CGSTAB, é de 10^{-5} ou até atingir um número máximo de 3500 iterações.

Cole (1989) investigou o comportamento da convergência dos métodos iterativos de Jacobi, Gauss-Seidel e do método de sobre-relaxamento sucessivo na obtenção da solução do sistema linear (2.26) usando a primeira ordem da expansão de Padé real. Ele notou que para frequências que não satisfazem a inequação

$$\omega^2 \geq \frac{c^2}{h^2} 8b_1, \quad (3.3)$$

a matriz de coeficientes deixa de ser diagonalmente dominante, e com isso a velocidade de convergências desses métodos fica lenta e pode até mesmo não convergir.

Nichols (1991) para resolver o sistema linear (2.26) usando a primeira ordem da expansão de Padé real e a equação normal de \mathbf{A} , $\mathbf{A}^\dagger \mathbf{A}$, usou o método CG devido sua convergência ser garantida em um número finito de passos (para matriz simétrica e positiva-definida). Ele observa que a velocidade de convergência do CG é lenta para baixas frequências, assim como os métodos investigados por Cole (1989), mas reporta que evitando as energias dos modos evanescentes e de grandes ângulos de propagação, a convergência para baixas frequências fica mais eficiente.

Levando em consideração as observações de Nichols (1991), que evitando as energias de grandes ângulos e de modos evanescentes a convergência do método CG para baixas frequências fica mais eficiente, ao invés de usar a expansão de Padé real, usamos a expansão de Padé complexa, pois ela atenua as energias de grandes ângulos e de modos evanescentes (MILINAZZO; ZALA; BROOKE, 1997). Apesar de usarmos o método Bi-CGSTAB que é mais adequado que o método CG para resolver sistemas lineares onde os elementos da matriz de coeficientes são complexos, esperamos que a velocidade de convergência do Bi-CGSTAB deva variar com o modelo de velocidade e espaçamento da malha. Para verificar isso, obtemos uma versão da inequação (3.3), usando os coeficientes de Padé complexo.

Para a matriz \mathbf{A} ser diagonalmente dominante, tem que satisfazer a condição

$$|A_{jj}| \geq \sum_{l=1}^M |A_{jl}|_{j \neq l} \quad j = 1, \dots, M. \quad (3.4)$$

Vamos verificar o quanto essa matriz, usando apenas um termo da série de Padé complexa em um meio homogêneo e com uma malha regular, satisfaz essa condição. Para esse caso, temos que para cada linha da matriz \mathbf{A} , o elemento da diagonal principal será dado por:

$$A_{jj} = \left[\left(\frac{\omega h}{c} \right)^2 - 4 \left(B_1 + i \frac{\omega h}{2c} A_1 \right) \right] \quad (3.5)$$

e fora da diagonal principal, temos apenas 4 elementos diferentes de zero, todos da mesma forma dados por:

$$A_{jl} = \left(B_1 + i \frac{\omega h}{2c} A_1 \right) \quad j \neq l. \quad (3.6)$$

Assim, temos que para essa matriz ser diagonalmente dominante tem que satisfazer a seguinte inequação:

$$\left(\frac{\omega h}{c} \right)^2 - 8 \left(\frac{\omega h}{c} \right) I_m \{A_1\} - 8 R_e \{B_1\} \geq 0, \quad (3.7)$$

cujas as soluções são dadas pelas frequências limite inferior, ω_{L_i} , e superior, ω_{L_s}

$$\omega_{L_i} \geq \left(\frac{c}{h} \right) \left[-2 I_m \{A_1\} + \sqrt{(2 I_m \{A_1\})^2 + 8 R_e \{B_1\}} \right], \quad (3.8)$$

$$\omega_{L_s} \leq \left(\frac{c}{h} \right) \left[-2 I_m \{A_1\} - \sqrt{(2 I_m \{A_1\})^2 + 8 R_e \{B_1\}} \right], \quad (3.9)$$

onde $I_m \{A_1\}$ e $R_e \{B_1\}$ denotam as partes imaginária de A_1 e parte real B_1 , respectivamente.

Estas inequações podem nos fornecer uma estimativa do quanto a matriz A é bem condicionada. Observando só a frequência limite inferior (3.8), temos que quanto maior ela for, pior condicionada será a matriz. Como consequência, menos eficiente será os métodos iterativos para resolver o sistema linear. Porém, para valores de frequências acima da frequência limite inferior (3.8) o método iterativo é mais eficiente.

Embora as condições (3.8) e (3.9) sejam apenas condições necessárias para o método iterativo ser mais eficiente, pois elas estimam o quanto a matriz é bem condicionada, podemos avaliar dessas condições como espaçamento da malha, a velocidade e os coeficientes de Padé complexo afetam a convergência do algoritmo Bi-CGSTAB. Vemos das inequações (3.8) e (3.9) que quando maior for a razão entre velocidade e espaçamento da malha (c/h) pior condicionada será a matriz, e com isso mais lenta será a velocidade de convergência do algoritmo Bi-CGSTAB. Na Figura (3.1), mostramos como a frequência limite inferior $f_L = \omega_{L_i}/2\pi$ varia com ângulo de rotação da linha de corte θ (na esquerda) e o número de iterações do algoritmo Bi-CGSTAB como função da frequência (na direita), para $h = 10m$ e $c = 1500m/s$. O gráfico na esquerda indica que aumentando θ a eficiência do método iterativo deve melhorar. No gráfico da direita, vemos que o número de iterações do algoritmo Bi-CGSTAB usando coeficientes de Padé real, é mais que o dobro do número de iterações quando se usa os coeficientes de Padé complexo tanto com $\theta = 25^\circ$ ou com $\theta = 45^\circ$. Aumentando o ângulo de rotação da linha de corte, de $\theta = 25^\circ$ para $\theta = 45^\circ$, o número de iterações diminui, porém essa diminuição tem pouco efeito no comportamento da convergência.

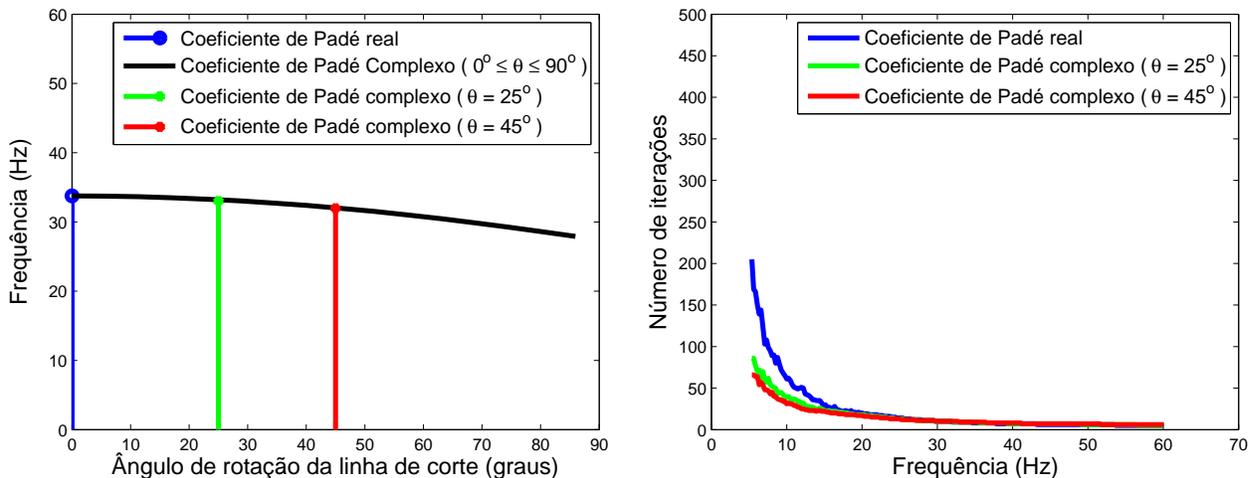


Figura 3.1: A esquerda: frequência limite, f_L , pelo ângulo de rotação da linha de corte em graus. A direita: número de iterações do algoritmo Bi-CGSTAB para cada frequência para $c = 1500m/s$ e $h = 10m$

Na Figura (3.2), mostramos a frequência limite inferior $f_L = \omega_{L_i}/2\pi$ (na esquerda) e o número de iterações do algoritmo Bi-CGSTAB como função da frequência (na direita),

para $h = 20m$ e $c = 1500m/s$. No gráfico da esquerda, podemos ver que a frequência limite inferior, $f_L = \omega_{L_i}/2\pi$, diminuiu com o aumento de h , deixando a matriz A melhor condicionada. Podemos ver no gráfico da direita, o efeito dessa melhora no condicionamento da matriz. Vemos que o número de iterações do algoritmo Bi-CGSTAB diminuiu comparando com a Figura (3.1). Observa-se também que usando coeficientes de Padé complexo, o número de iterações do algoritmo Bi-CGSTAB diminui em relação ao real, porém essa diminuição tem menos impacto na convergência do algoritmo Bi-CGSTAB do que quando usamos $h = 10m$. Aumentando o ângulo de rotação da linha de corte, de $\theta = 25^\circ$ para $\theta = 45^\circ$, o número de iterações praticamente se mantém o mesmo nesse caso. Na Figura (3.3), mostramos frequência limite inferior $f_L = \omega_{L_i}/2\pi$ (na esquerda) e

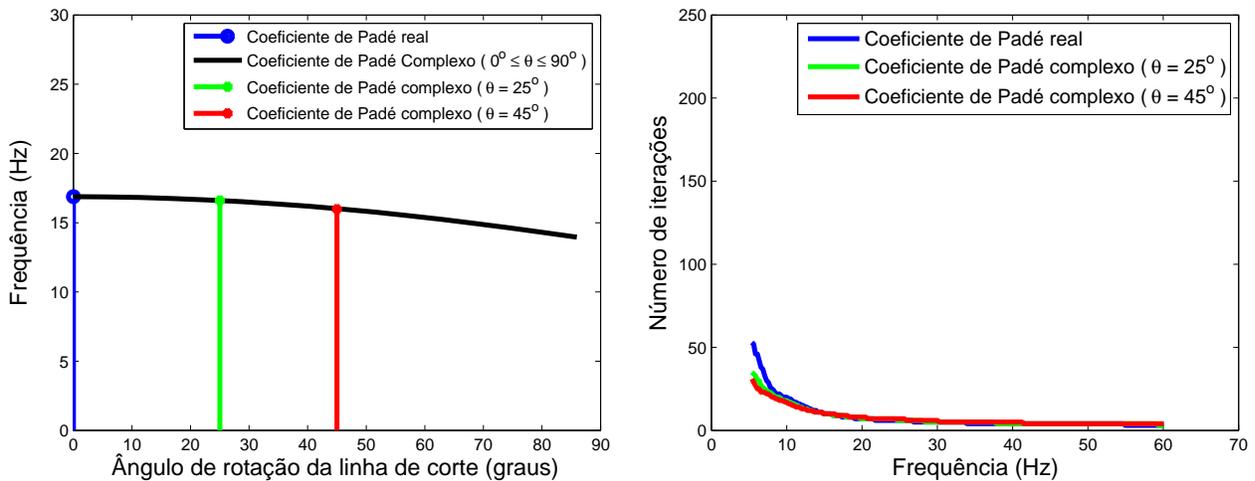


Figura 3.2: A esquerda: frequência limite, f_L , pelo ângulo de rotação da linha de corte em graus. A direita: número de iterações do algoritmo Bi-CGSTAB para cada frequência para $c = 1500m/s$ e $h = 20m$

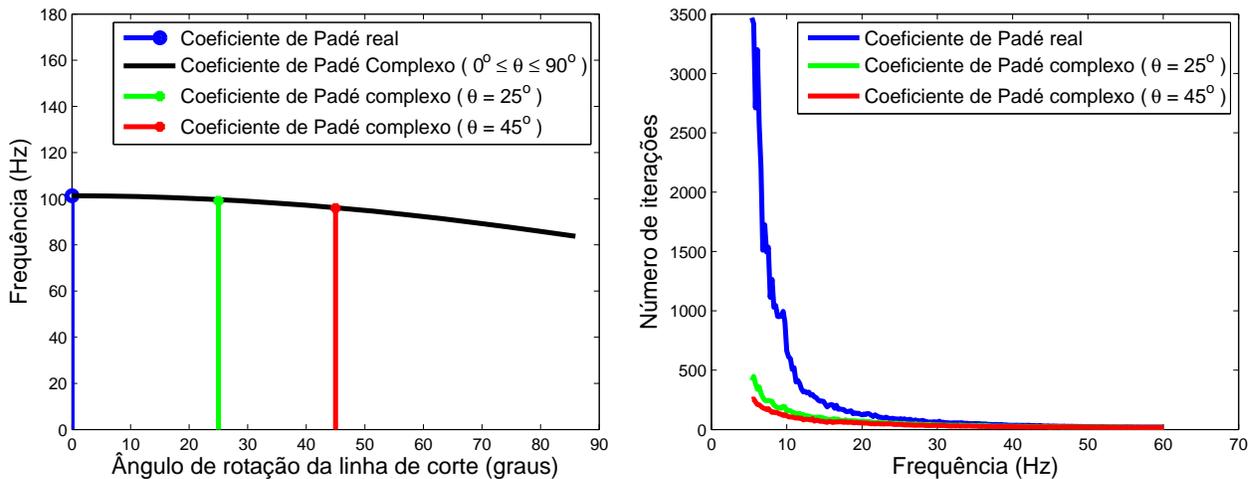


Figura 3.3: A esquerda: frequência limite, f_L , pelo ângulo de rotação da linha de corte em graus. A direita: número de iterações do algoritmo Bi-CGSTAB para cada frequência para $c = 4500m/s$ e $h = 10m$

o número de iterações do algoritmo Bi-CGSTAB como função da frequência (na direita),

para $h = 10m$ e $c = 4500m/s$. No gráfico da esquerda, podemos ver que a frequência limite inferior, $f_L = \omega_{L_i}/2\pi$, aumentou bruscamente com o aumento de c , deixando a matriz \mathbf{A} mal condicionada. Podemos ver no gráfico da direita, o efeito desse mau condicionamento da matriz. Vemos que o número de iterações do algoritmo Bi-CGSTAB aumentou bastante comparando com a Figura (3.1). Observa-se também que usando coeficientes de Padé complexo, o número de iterações do algoritmo Bi-CGSTAB diminui bruscamente em relação ao real, este efeito tem grande impacto na convergência do algoritmo Bi-CGSTAB. Aumentando o ângulo de rotação da linha de corte, de $\theta = 25^\circ$ para $\theta = 45^\circ$, o número de iterações tem uma diminuição da ordem de 200 iterações.

Da comparação das Figuras (3.1), (3.2) e (3.3) percebemos que quanto mais mal condicionada for a matriz \mathbf{A} , maior é o impacto do uso dos coeficientes de Padé complexo sobre o comportamento da convergência do algoritmo Bi-CGSTAB. Ressaltamos que nestas figuras, estamos usando apenas um termo da série de Padé tanto complexa como real. Se usarmos mais termos da série de Padé tanto complexa como real, menos eficiente será o método iterativo. Na Figura (3.4), mostramos o número de iterações em função da frequência com $c = 1500m/s$, $h = 10m$ usando dois termos da série de Padé (gráfico da esquerda) e um termo da série de Padé (gráfico da direita). Comparando o gráfico da esquerda como o da direita, podemos ver que o número das iterações do algoritmo Bi-CGSTAB quando usamos 2 termos da série de Padé aumenta bastante comparado quando usamos 1 termo. Porém o aumento do número das iterações quando usamos 2 termos dos coeficientes de Padé real é bem mais elevado comparado com o aumento quando usamos 2 termos dos coeficientes de Padé complexo. Aumentando o ângulo de rotação da linha de corte de $\theta = 25^\circ$ para $\theta = 45^\circ$ o número de iterações cai quase pela metade.

No próximo capítulo, iremos discutir o uso de *perfectly matched layer* (PML)(BÉRENGER, 1994) em nosso algoritmo para evitar reflexões nas bordas do plano horizontal da malha e uma técnica de otimização dos coeficientes de Padé complexo, para melhorar a abertura angular do nosso algoritmo usando apenas um termo da série de Padé complexa. Na Figura (3.5) mostramos como o uso do PLM (gráfico da esquerda) e a técnica de otimização (gráfico da direita) afetam a convergência do algoritmo Bi-CGSTAB. Estamos usando $c = 1500m/s$ e $h = 10m$. Como podemos ver nos gráficos, tanto o uso do PML quanto o uso de coeficientes otimizados, aumentam o número de iterações do Bi-CGSTAB.

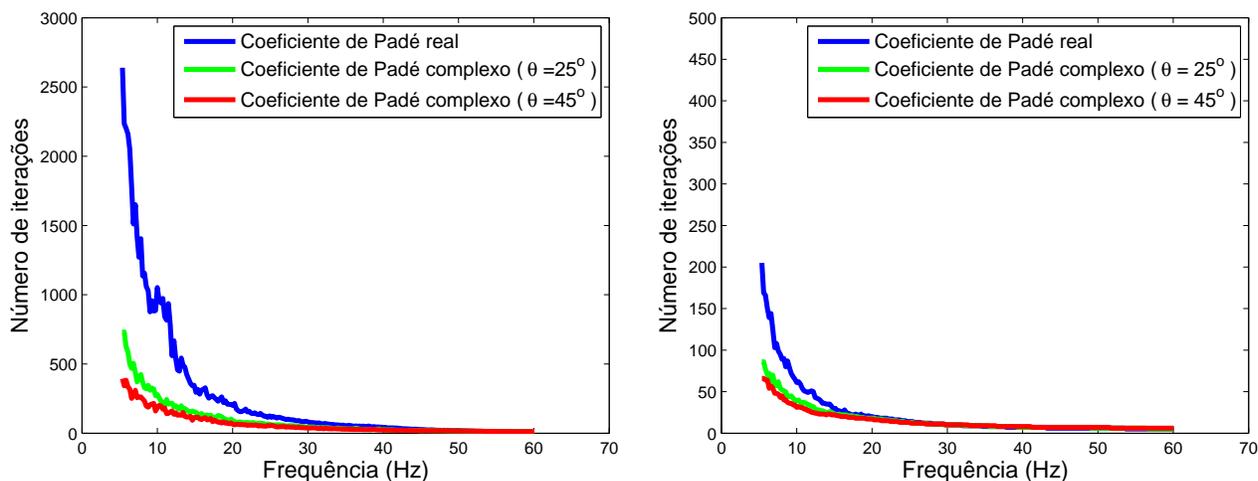


Figura 3.4: A esquerda: número de iterações do algoritmo Bi-CGSTAB para cada frequência para $c = 1500m/s$, $h = 10m$ e dois termos da série de Padé complexa. A direita: número de iterações do algoritmo Bi-CGSTAB para cada frequência para $c = 1500m/s$, $h = 10m$ e um termo da série de Padé complexa

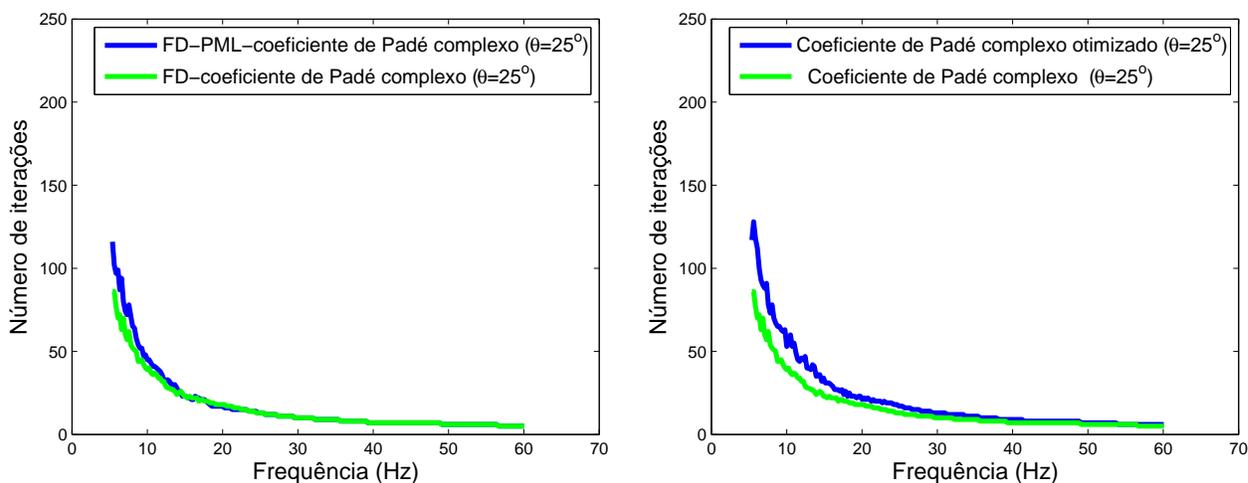


Figura 3.5: Número de iterações do algoritmo Bi-CGSTAB para cada frequência para $c = 1500m/s$ e $h = 10m$. A esquerda: Comparação entre FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte 25° com e sem PML nas bordas do plano horizontal. A direita: Comparação entre FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte 25° com otimização e sem otimização.

4 ALGORITMO DE MIGRAÇÃO FD ITERATIVO

Neste capítulo discutiremos a causa da anisotropia numérica, a implementação e as características do algoritmo de migração FD iterativo. Para avaliar essas características, usamos um modelo de velocidade homogêneo de 1500 m/s e uma malha regular de dimensões $676 \times 676 \times 210$ ao longo das direções x_1 , x_2 e x_3 , respectivamente, com espaçamento de 10 m. A fonte é colocada na superfície no meio da malha. O sinal da fonte é um pulso Ricker com a frequência de pico de 25 Hz e uma amostragem no tempo de 8 ms. Estamos usando uma banda de frequência de 5 Hz a 60 Hz. A precisão da solução do sistema linear calculado com o algoritmo Bi-CGSTAB é de 10^{-5} ou até atingir um número máximo de 2000 iterações.

4.1 A ANISOTROPIA NUMÉRICA

Quando queremos implementar uma equação, a nossa preocupação é encontrar uma maneira simples e eficiente do ponto de vista computacional. Levando em consideração esta simplicidade e eficiência, frequentemente quando se faz migração em profundidade 3D através da equação da onda unidirecional via método FD implícito, se usa o método *splitting inline* e *crossline*, ou seja, as equações (2.27) e (2.28) em vez da equação (2.22) para extrapolar o campo de onda para baixo. Por um lado, ganhamos eficiência computacional, mas por outro, usar o método *splitting inline* e *crossline*, produz anisotropia numérica. Esse problema de anisotropia, ocorre porque quando o operador que governa a propagação da onda é dividido em dois, um que governa a propagação da onda na direção *inline* e outro na direção *crossline*, estamos supondo que eles são independentes um do outro, e isso só é verdade para ondas que se propagam nas direções *inline* e *crossline*, então para qualquer outra onda que se propague com um ângulo azimutal diferente de 0° ou 90° , essa divisão deixa de ser exata, e tendo como consequência a anisotropia numérica. Devido a esta anisotropia, os refletores inclinados podem ser erroneamente posicionados pelo processo de migração, principalmente quando estes refletores estão posicionado em um ângulo azimutal de 45° (LI, 1991; RISTOW; RÜHL, 1997; BIONDI, 2006).

Na parte de cima da Figura (4.1) mostramos a resposta ao impulso do método FD implícito usando as equações (2.27) e (2.28) com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte de 45° nas profundidades $x_3 = 700$ m (a esquerda) e $x_3 = 1050$ m (a direita). O círculo vermelho nesta figura, indica a posição exata que deveria está a frente de onda se não existisse anisotropia numérica, como podemos ver na parte de

baixo dessa figura, onde mostramos a resposta ao impulso do algoritmo de migração RTM nas profundidades $x_3 = 700$ m (a esquerda) e $x_3 = 1050$ m (a direita) . É visível, na resposta ao impulso mostrada na parte de cima dessa figura, a presença da anisotropia numérica em ambas as profundidades, porém quanto maior a profundidade mais acentuada ela fica.

4.2 IMPLEMENTANDO O ALGORITMO DE MIGRAÇÃO FD ITERATIVO

Com os avanços computacionais dos últimos anos, hoje em dia podemos trocar os benefícios computacionais do método de *splitting inline* e *crossline* por imagens mais precisas. Com esse compromisso de obter melhores imagens, para extrapolar o campo de onda para baixo, optamos por resolver a equação (2.22) ao invés das equações (2.27) e (2.28), ou seja, não usamos a técnica de *splitting inline* e *crossline*, e assim evitaremos o surgimento da anisotropia numérica.

Como precisamos resolver um sistema linear onde os elementos da matriz de coeficientes equação (2.26) são complexos, optamos por usar o algoritmo iterativo Bi-CGSTAB discutido no capítulo anterior, pois ele é mais adequado para esse tipo de matriz. Para resolver este sistema linear usando o algoritmo iterativo Bi-CGSTAB, precisamos armazenar as matrizes \mathbf{A} e \mathbf{A}^* na memória do computador, porém para encontrar a solução do sistema linear, precisamos apenas dos elementos diferentes de zero dessas matrizes, sendo assim, com o objetivo de usar o menor espaço possível da memória do computador, usamos o método de armazenamento *compressed row storage* para armazenar apenas os elementos diferentes de zero das referidas matrizes.

A maneira como implementamos o algoritmo de migração FD iterativo pode ser descrito da seguinte maneira:

Algoritmo 4.1 (FD iterativo)

1. **Entrando com o campo de onda conhecido na superfície**, $P(x_1, x_2, x_3 = x_3^0, t)$;
2. **Laço para todos os níveis em profundidade**;
3. **Laço na frequência (para toda a banda de frequência)**;
4. *Resolvendo o termo analítico*;
5. *Resolvendo o termo FD*;
6. *Condição de imagem*;
7. **Fim do laço na frequência**;
8. **Fim do laço em profundidade**;

Um teste numérico preliminar do algoritmo de migração FD iterativo pode ser visto na parte do meio da Figura (4.1), onde estamos mostrando a resposta ao impulso do algoritmo FD iterativo com 1 termo da série de Padé complexa e ângulo de rotação da linha de corte de 45° nas profundidades $x_3 = 700$ m (a esquerda) e $x_3 = 1050$ m (a direita). Percebemos nessas imagens a ausência da anisotropia numérica, mostrando assim, a eficácia do algoritmo FD iterativo para resolver o problema da anisotropia numérica.

Na implementação computacional das equações que resultou nas respostas ao impulso da Figura (4.1), não nos preocupamos com as reflexões nas bordas do plano horizontal. Na subseção que segue, iremos dar um tratamento melhor para essas reflexões nessas bordas.

4.2.1 REFLEXÕES NAS BORDAS DO PLANO HORIZONTAL

Como estamos resolvendo a equação (2.19) em um domínio finito Ω_b^h , e como a onda no plano horizontal pode se propagar em dois sentidos, é de se esperar que haja reflexões nas bordas desse plano.

Para avaliar esse fenômeno de reflexão nas bordas do plano horizontal, usamos os mesmo dados descritos no início deste capítulo, com exceção do espaçamento da malha que é de 5 m e do modelo de velocidade que é constituído de duas camadas planos horizontais, onde a primeira camada, que vai até 1000 m em profundidade, tem um valor de 1500 m/s e a segunda camada, que vai até o final do modelo em profundidade, tem um valor 2500 m/s.

Na Figura (4.2), mostramos a seção horizontal em $x_3 = 1900$ m da resposta ao impulso calculado com algoritmo de migração FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte de 45° sem um tratamento adequado nas bordas do plano horizontal. Podemos observar que existem reflexões nas bordas desse plano. Para evitar ou diminuir o máximo possível esse tipo de fenômeno nessas bordas, usamos a técnica do PML, inicialmente introduzida por Bérenger (1994).

A ideia básica do PML é adicionar nas bordas do domínio físico Ω_b^h , uma faixa artificial, de tal maneira que na interface entre o domínio físico e a faixa artificial, não haja reflexão e que a energia da onda caia exponencialmente dentro dessa faixa (COLLINO; TSOGKA, 2001). Para o nosso caso, onde só existem reflexões nas bordas do plano horizontal, teremos que fazer a seguinte extensão de x_1 e x_2 :

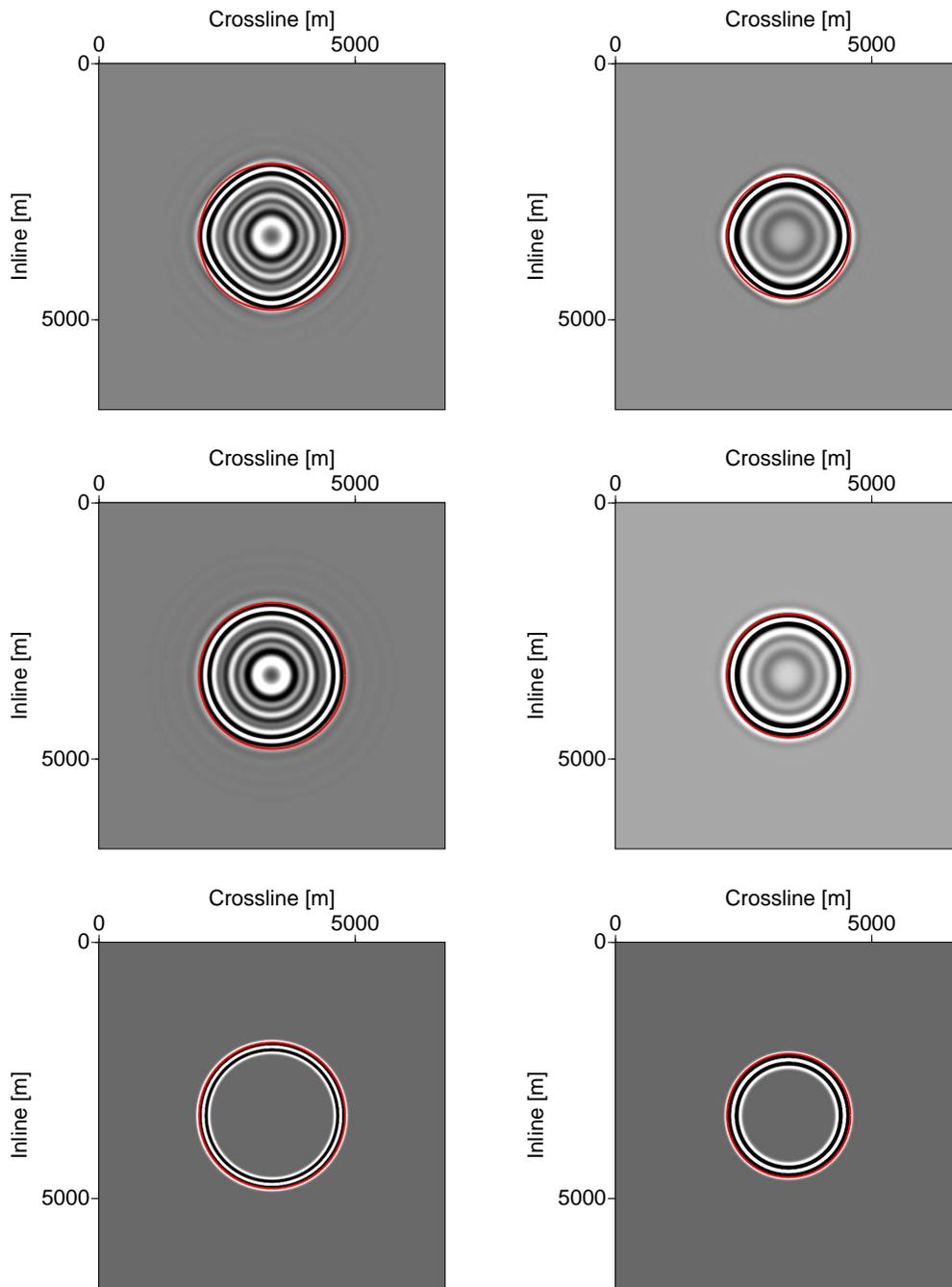


Figura 4.1: Seções da resposta ao impulso 3D no plano paralelo ao plano x_1x_2 em $x_3 = 700$ m, coluna da esquerda, e $x_3 = 1050$ m, coluna da direita. Em cima: FD usando *splitting inline* e *crossline*. No meio: FD sem *splitting inline* e *crossline*. Embaixo: RTM.

$$\tilde{x}_j = x_j - \frac{i}{\omega} \int_0^x d(s) ds; j = 1, 2. \quad (4.1)$$

Onde $d(s)$ é o fator de amortecimento dado por

$$d(x_j) = d_0 \left(\frac{x_j}{\delta} \right)^2, \quad (4.2)$$

que é nulo fora da faixa artificial e positivo dentro dela. Onde d_0 é o amortecimento máximo e δ é a largura da faixa artificial. Usando essa transformação, obtemos as seguintes derivadas parciais:

$$\frac{\partial}{\partial x_j} \rightarrow \frac{\partial}{\partial \tilde{x}_j} = \frac{i\omega}{i\omega + d(x_j)} \frac{\partial}{\partial x_j}. \quad (4.3)$$

Substituindo a equação (4.3) na equação (2.19) e discretizando-a como foi feito no Capítulo 2, obteremos outro sistema linear

$$\widetilde{\mathbf{A}}\mathbf{P}^{k+1} = \widetilde{\mathbf{A}}^*\mathbf{P}^k, \quad (4.4)$$

com $\widetilde{\mathbf{A}}$ e $\widetilde{\mathbf{A}}^*$ o mesmo domínio e mesma dimensão de \mathbf{A}

Na Figura (4.3), mostramos a seção horizontal em $x_3 = 1900$ m da resposta ao impulso calculado com algoritmo de migração FD iterativo com 1 termo da série de Padé complexa e ângulo de rotação da linha de corte de 45° com PML nas bordas do plano horizontal. Usamos uma faixa artificial para o PML de 25 pontos com um espaçamento entre os pontos de 10 m. Observamos nesta figura que usando PML, praticamente eliminamos as reflexões nas bordas do plano horizontal.

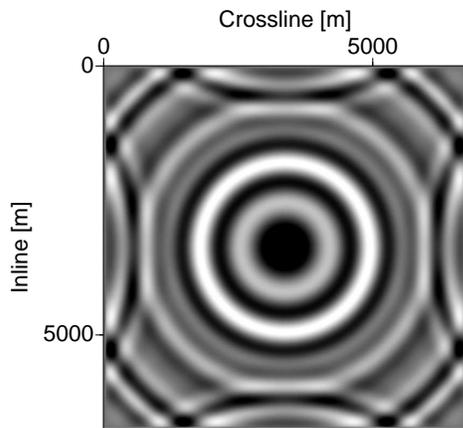


Figura 4.2: Seção horizontal em $x_3 = 1900$ m da resposta ao impulso 3D do algoritmo de migração FD iterativo usando 1 termo da série de Padé complexa com ângulo de rotação da linha de corte 45° .

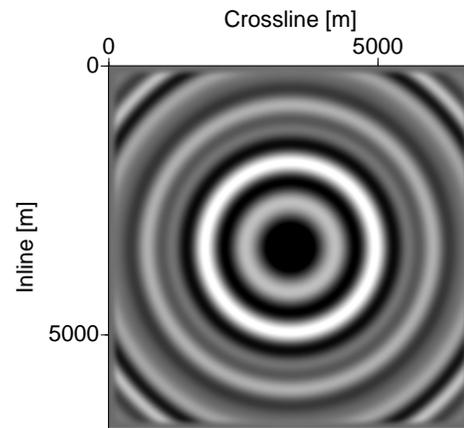


Figura 4.3: Seção horizontal em $x_3 = 1900$ m da resposta ao impulso 3D do algoritmo de migração FD iterativo-PML usando 1 termo da série de Padé complexa com ângulo de rotação da linha de corte 45° .

4.2.2 ABERTURA ANGULAR

Esta subseção é dedicada a analisar a abertura angular do algoritmo FD iterativo. Usamos a resposta ao impulso calculado com o algoritmo de migração RTM como referência para compararmos com a resposta ao impulso do algoritmo FD iterativo.

Como estamos usando um modelo de velocidade homogêneo, para fazer a análise da abertura angular, mostraremos apenas a resposta ao impulso do algoritmo FD iterativo e do algoritmo RTM no plano vertical paralelo a direção *inline* em $x_2 = 3380$ m.

Na Figura (4.4), em cima temos a resposta ao impulso do algoritmo FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte de 45° . No meio temos a resposta ao impulso do algoritmo FD iterativo com 1 termo da série de Padé real. Embaixo temos a resposta ao impulso do RTM. A curva em vermelho, indica a posição exata da resposta ao impulso do RTM e os raios em azul tem uma abertura angular com a vertical de 45° .

Podemos perceber na Figura (4.4) que o algoritmo FD iterativo com 1 termo da série de Padé complexa e ângulo de rotação da linha de corte de 45° é um bom algoritmo para imagear refletores com ângulo de mergulho de até 45° . Comparando a resposta ao impulso de cima (usando aproximação de Padé complexa) com a do meio (usando aproximação de Padé real), percebemos que em relação a abertura angular elas são equivalentes, porém em relação a qualidade da imagem, a imagem de cima é bem melhor. Pois ela não apresenta dispersão numérica e nem efeitos artificiais, como esses artefatos que aparecem após a frente de onda. Na comparação das duas primeiras imagens com a terceira, percebemos que a amplitude do sinal são diferentes. O algoritmo FD iterativo apresenta uma maior amplitude relativa do que o RTM. Essa diferença de amplitude é devido ao fato que a equação da onda unidirecional é exata para descrever o tempo de trânsito da onda, porém não trata corretamente a amplitude da mesma. Nessa comparação com o algoritmo de migração RTM, percebemos também que a frente de onda do algoritmo de migração FD iterativo está adiantada em relação a frente de onda o algoritmo de migração RTM, isso é devido ao fato que no algoritmo FD iterativo, o pulso fonte é um pulso causal e no algoritmo RTM o pulso é não causal.

Ressaltamos que a frente de onda adiantada da resposta ao impulso do algoritmo FD iterativo em relação a resposta ao impulso do RTM e o problema de amplitude da equação de onda unidirecional será visto em todas as respostas ao impulso do algoritmo FD iterativo deste capítulo.

Para melhorar a abertura angular do algoritmo FD iterativo, nesta dissertação,

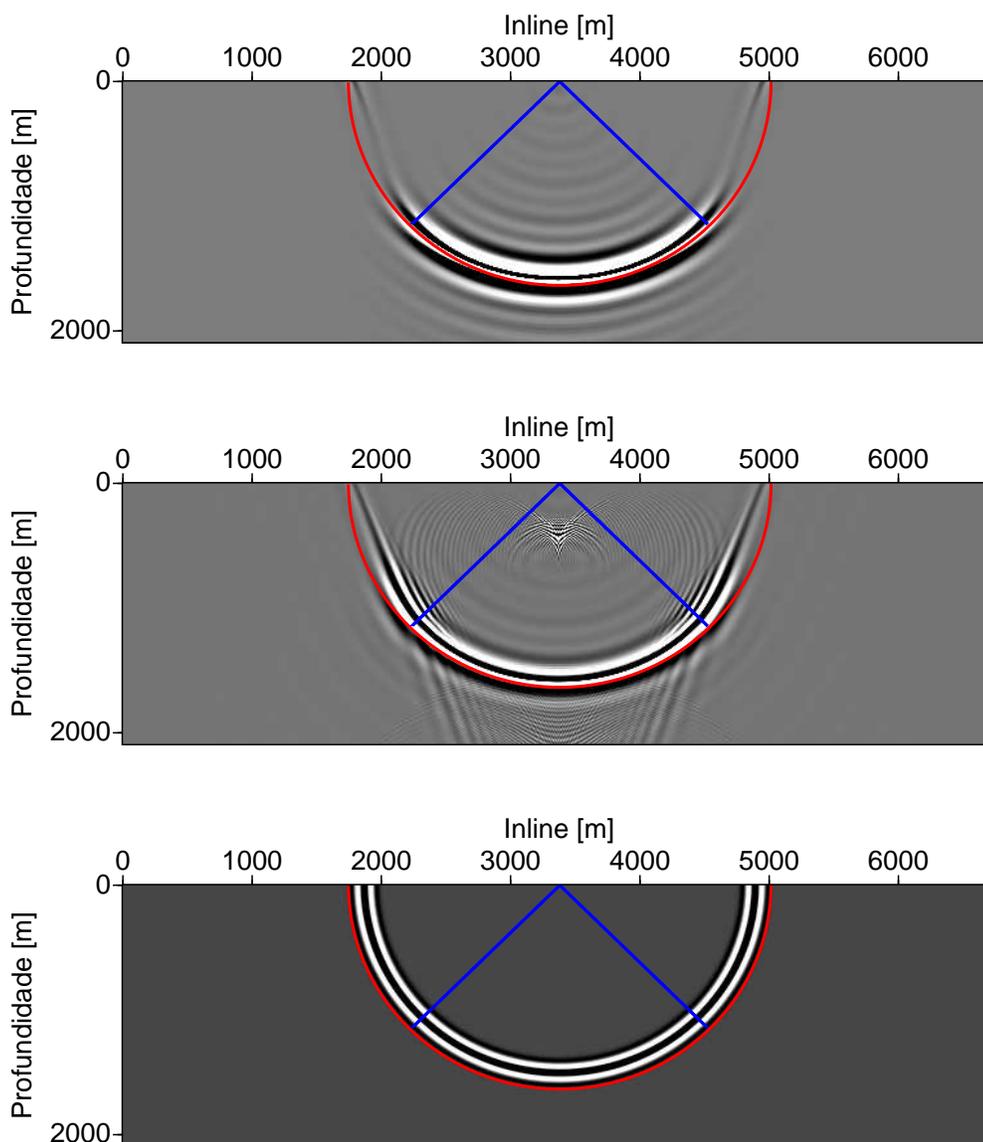


Figura 4.4: Seção vertical da resposta ao impulso 3D no plano paralelo a direção *inline* em $x_2 = 3380$ m. Em cima: FD iterativo com 1 termo da série de Padé complexa com ângulo da linha de corte $\theta = 45^\circ$. No meio: FD iterativo com 1 termo da série de Padé real. Embaixo: RTM

analisamos três maneiras:

(i) usando mais termos da série de Padé complexa:

Quanto mais termos da série de Padé usarmos no algoritmo FD iterativo, melhor ele será para imagear refletor com ângulos de mergulho acima de 45° . Por exemplo, usando dois termos da série de Padé complexa com ângulo de rotação da linha de corte de 45° , obtemos uma resposta ao impulso bem melhor que usando um único termo, como podemos observar na Figura (4.5). Onde em cima é a resposta ao impulso do algoritmo FD

iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte de 45° . No meio é a resposta ao impulso do algoritmo FD iterativo com 2 termo da série de Padé complexa com ângulo de rotação da linha de corte de 45° . Embaixo a resposta ao impulso do RTM.

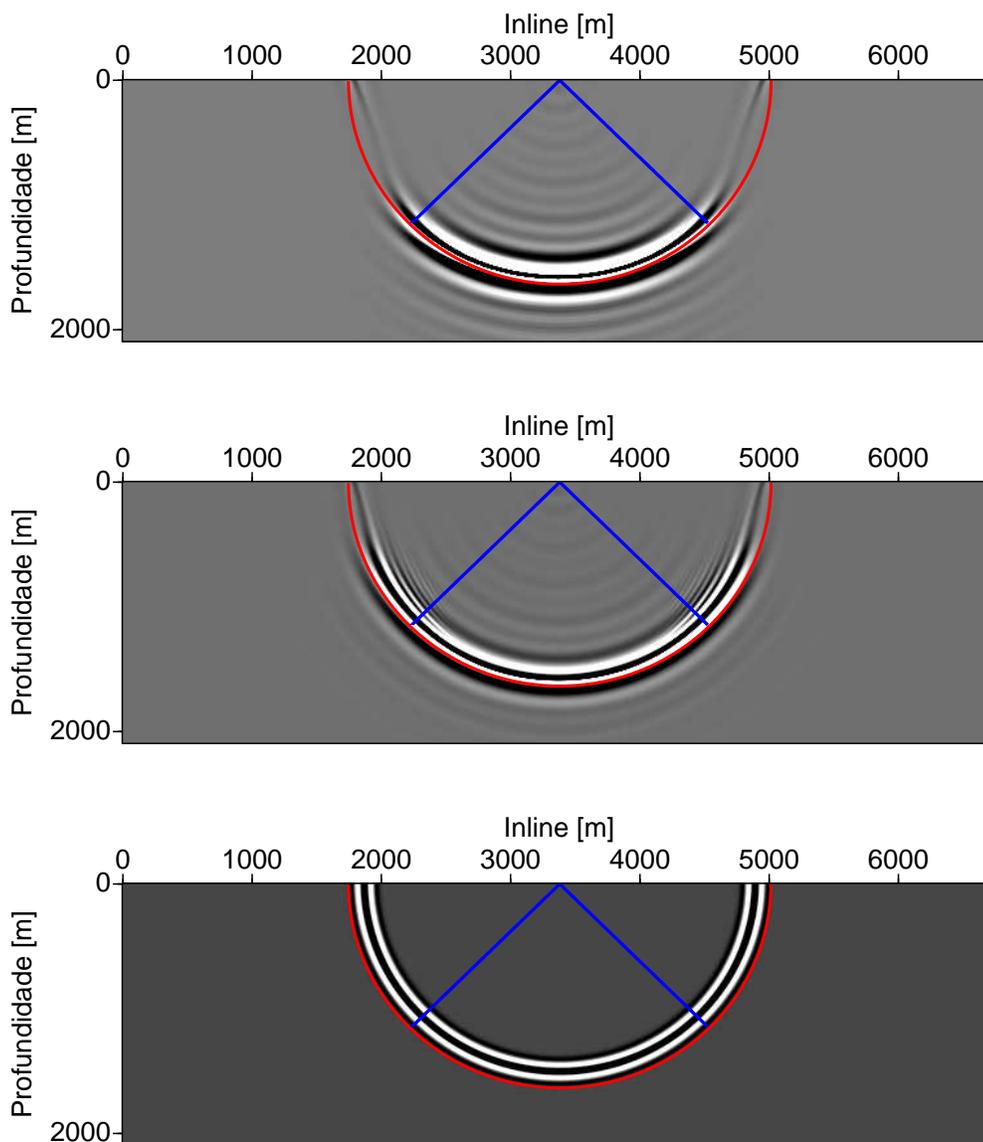


Figura 4.5: Seção vertical da resposta ao impulso 3D no plano paralelo a direção *inline* em $x_2 = 3380$ m. Em cima: FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte $\theta = 45^\circ$. No meio: FD iterativo com 2 termo da série de Padé complexa com ângulo de rotação da linha de corte 45° . Embaixo: RTM

(ii) usando menores ângulos de rotação da linha de corte:

Uma outra maneira de melhorar a abertura angular do algoritmo FD iterativo, é diminuindo o ângulo de rotação da linha de corte. Como, por exemplo, um ângulo de

rotação de 25° , como podemos observar na Figura (4.6). Onde em cima é a resposta ao impulso do algoritmo FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte de 45° . No meio é a resposta ao impulso do algoritmo FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte de 25° . Embaixo é a resposta ao impulso do RTM.

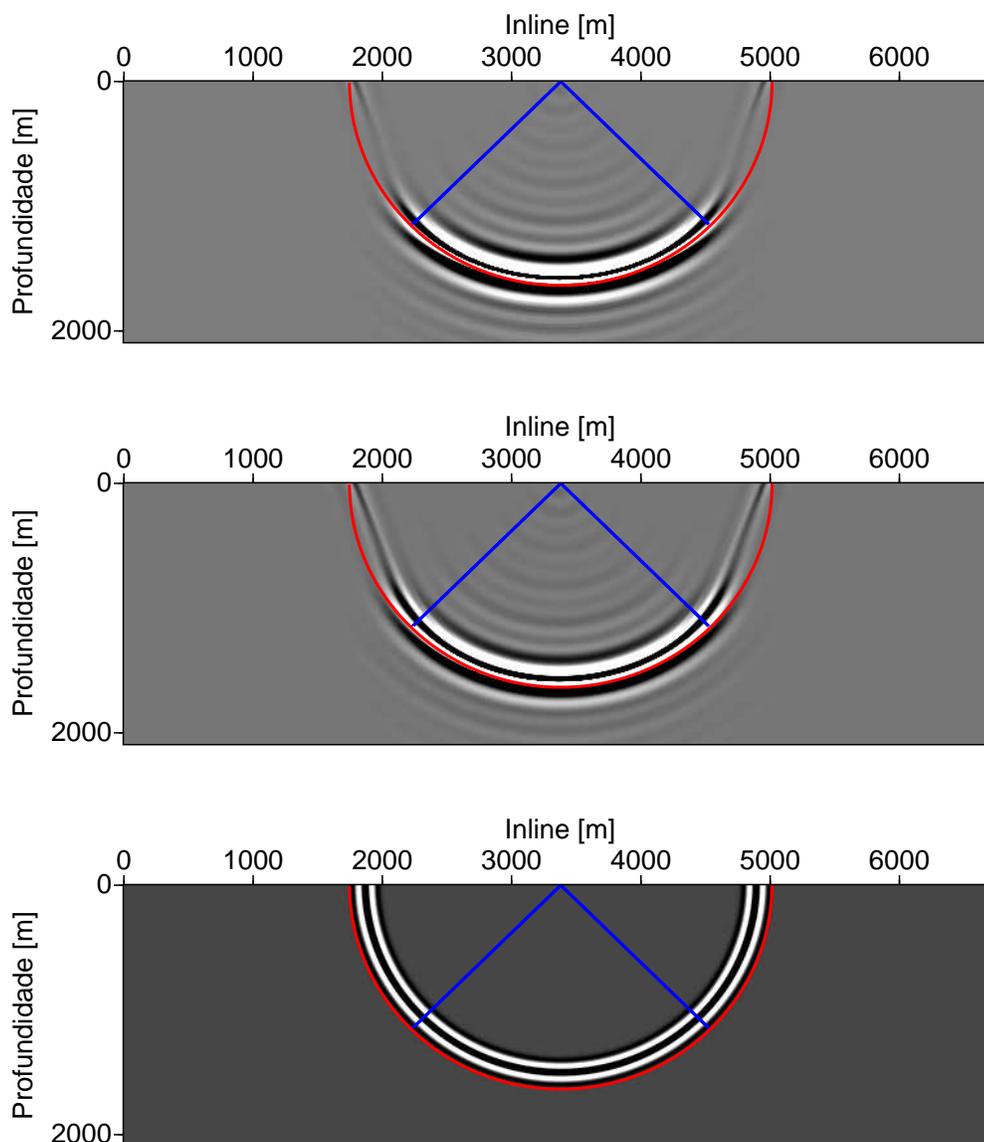


Figura 4.6: Seção vertical da resposta ao impulso 3D no plano paralelo a direção *inline* em $x_2 = 3380$ m. Em cima: FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte $\theta = 45^\circ$. No meio: FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte $\theta = 25^\circ$. Embaixo: RTM

(iii) Usando coeficientes de Padé complexo otimizados:

Uma outra estratégia que usamos para melhorar a abertura angular do algoritmo

FD iterativo foi otimizando os coeficientes de Padé complexo. A ideia dessa otimização é obter a maior abertura possível do operador de propagação da onda, usando um número reduzido de coeficientes de Padé complexo. No presente trabalho, para otimizar os coeficientes de Padé complexo, usamos o método dos mínimos quadrados para otimizar os coeficientes de Padé real e a partir desses coeficientes reais otimizados, calculamos os coeficientes de Padé complexo. Mostramos o resultado dessa otimização na Figura (4.7). Onde em cima é a resposta ao impulso do algoritmo FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte de 25° . No meio é a resposta ao impulso do algoritmo FD iterativo com 1 termo da série de Padé complexa otimizado com ângulo de rotação da linha de corte de 25° . Embaixo é a resposta ao impulso do RTM.

Todas as três maneiras usadas aqui para melhorar a abertura angular do algoritmo FD iterativo tem um custo computacional adicional, como foi visto no capítulo anterior quando estávamos analisando a velocidade de convergência do algoritmo iterativo Bi-CGSTAB.

ÚLTIMAS CONSIDERAÇÕES DO CAPÍTULO

Neste capítulo discutimos o problema da anisotropia numérica e mostramos que o algoritmo de migração FD iterativo consegue resolver tal problema. Além disso, tivemos sucesso no uso do PML para evitar reflexões nas bordas do plano horizontal. Observamos que a qualidade da resposta ao impulso obtida com algoritmo FD iterativo usando aproximação de Padé complexa, evita certos problemas que o uso da aproximação de Padé real trás para a imagem. Discutimos também, estratégias de melhorar a abertura angular do algoritmo de migração FD iterativo.

Todos os teste numéricos do presente capítulo, foram feitos em modelos de velocidades sem variações laterais de velocidades. Para testa a eficiência deste algoritmo em um modelo de velocidade com fortes variações laterais de velocidades, no próximo capítulo iremos realizar testes numéricos usando o modelo de sal da SEG/EAGE.

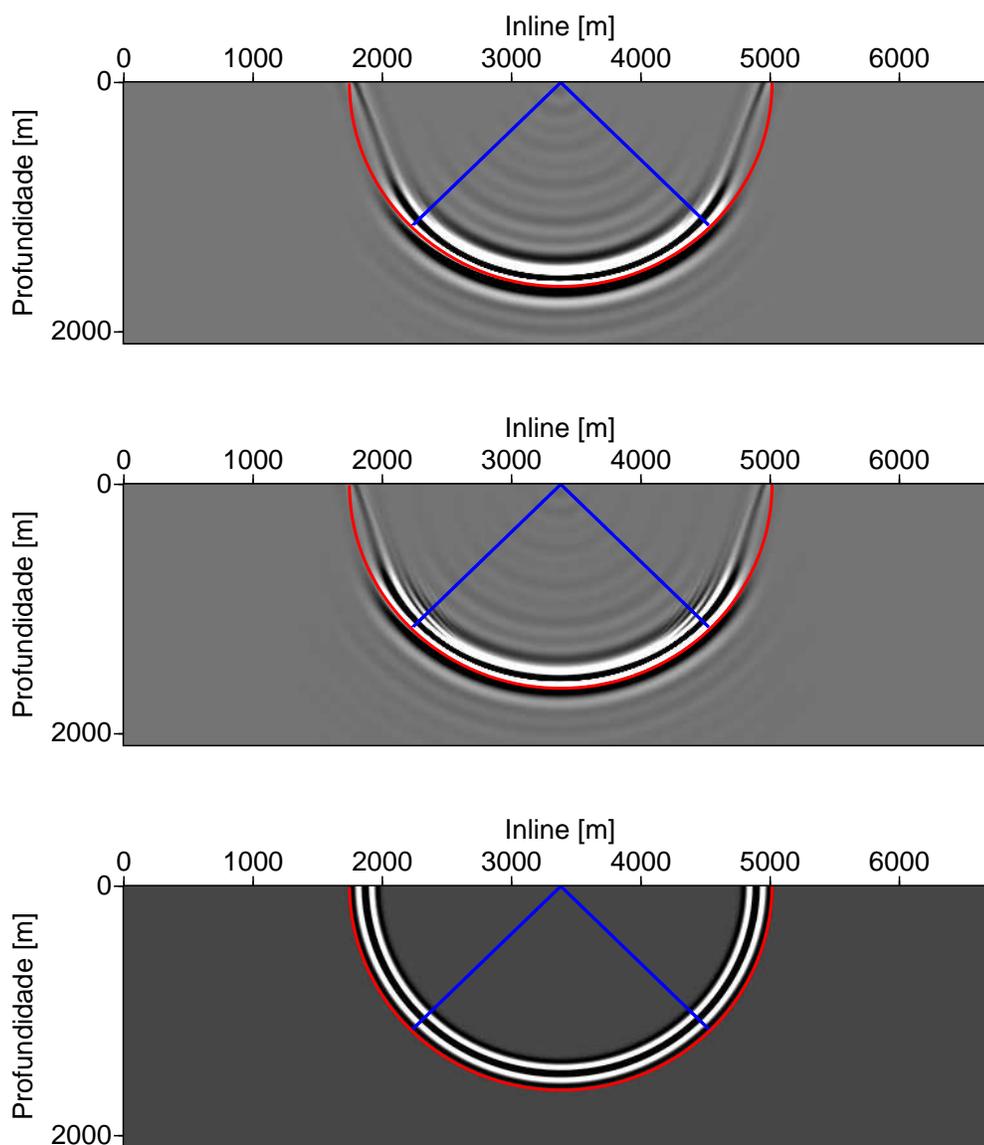


Figura 4.7: Seção vertical da resposta ao impulso 3D no plano paralelo a direção *inline* em $x_2 = 3380$ m. Em cima: FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte $\theta = 25^\circ$. No meio: FD iterativo com 1 termo da série de Padé complexa otimizado com ângulo de rotação da linha de corte $\theta = 25^\circ$. Embaixo: RTM

5 RESULTADOS NUMÉRICOS USANDO MODELO DE SAL

Neste capítulo usamos o modelo de sal SEG/EAGE para calcular a resposta ao impulso com algoritmo de migração FD iterativo com PML nas bordas do plano horizontal. Usamos uma faixa artificial para o PML de 25 pontos com um espaçamento entre os pontos de 10 m. A precisão do método iterativo é de 10^{-5} ou até atingir um número máximo de 2000 iterações. Para validar esta resposta ao impulso, a comparamos com a resposta ao impulso calculado com o algoritmo de migração FD usando a versão serial do pacote MUMPS (AMESTOY et al., 2001, 2006) para resolver o sistema linear, o qual chamaremos algoritmo de migração FD MUMPS, e com a resposta ao impulso do algoritmo de migração RTM. Para filtrar os constaste de velocidade *spike* usado pra simular os refletores no modelo original, aplicamos um filtro mediano de dimensões $7 \times 7 \times 7$. Usamos uma malha de dimensões $651 \times 651 \times 210$ ao longo das coordenadas x_1 , x_2 e x_3 , respectivamente. O espaçamento da malha é uniforme e igual a 10 m. A fonte injetada, é localizada na superfície e no centro da malha. O sinal da fonte é um pulso Ricker com frequência de pico de 25 Hz, e um intervalo temporal de 8 ms. Usamos uma banda de frequência de 5 Hz a 60 Hz.

Todas as seções verticais e horizontais, da resposta ao impulso, mostradas neste capítulo, são nas posições das Figuras (5.1), (5.2) e (5.3), onde estamos mostrando os cortes verticais em planos paralelos a direção *inline*, cortes verticais em planos paralelos a direção *crossline* e os cortes horizontais em planos paralelos ao plano x_1x_2 , respectivamente, do modelo de sal SEG/EAGE.

Nas Figuras (5.4) e (5.5) mostramos as seções verticais da resposta ao impulso calculado com o algoritmo de migração FD iterativo, FD MUMPS e RTM em planos paralelos as direções *inline* em $x_2 = 1690$ m (Figura 5.4) e *crossline* em $x_1 = 1690$ m (Figura 5.5). Na parte de cima de ambas as figuras, a esquerda, mostramos a resposta ao impulso do algoritmo de migração FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte de 45° , a direita, a resposta ao impulso com o algoritmo de migração FD MUMPS com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte de 45° . No meio de ambas as figuras, a esquerda, mostramos a resposta ao impulso do algoritmo de migração FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte de 25° e a direita com ângulo de rotação da linha de corte de 25° otimizado. Na parte de baixo de ambas as figuras, a esquerda, mostramos a resposta ao impulso do algoritmo de migração FD iterativo com 2 termos da série de Padé complexa com ângulo de rotação da linha de corte de 45° , a direita, a resposta ao impulso do algoritmo de migração RTM. Comparando as duas seções verticais

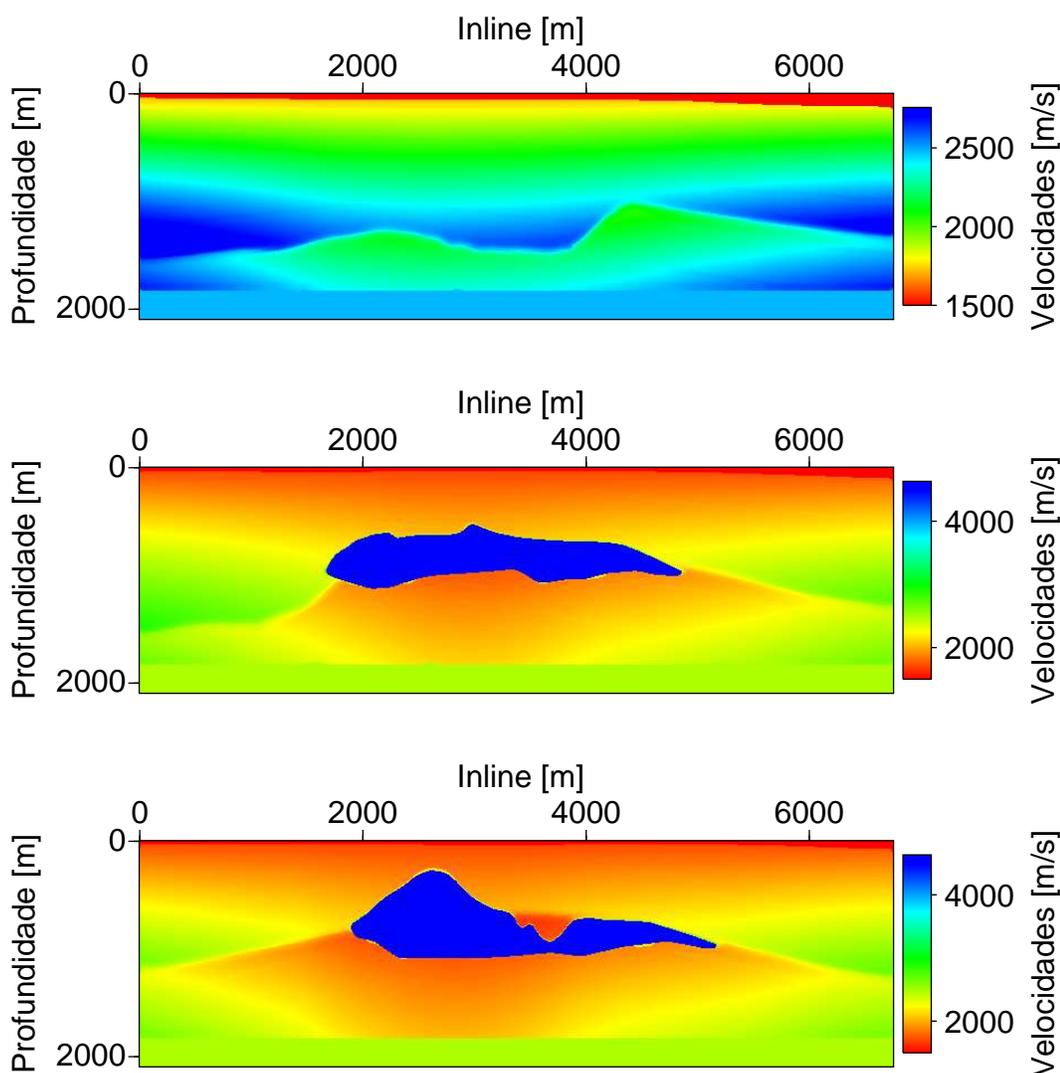


Figura 5.1: Seções verticais do modelo de sal SEG/EAGE em planos paralelos a direção *inline*. Em cima: em $x_2 = 1690$ m. No meio: em $x_2 = 3380$ m. Embaixo: em $x_2 = 4160$ m.

da resposta ao impulso do algoritmo de migração FD iterativo e FD MUMPS (parte de cima de ambas as figuras), uma com a outra, vemos que elas são iguais. Mostrando assim, que o algoritmo iterativo Bi-CGSTAB é tão bom quanto o método direto para resolver o sistema linear do algoritmo de migração FD. Comparando essas mesmas seções verticais, com a suas respectivas resposta ao impulso do algoritmo de migração RTM, vemos que os eventos migrado com ângulos de mergulho abaixo de 45° , correspondem aos eventos migrado com esse mesmo ângulo de mergulho usando o algoritmo de migração RTM. Essa restrição em relação ao ângulo de mergulho dos algoritmos de migração FD iterativo e FD MUMPS com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte de 45° são intrínsecos desses algoritmos. Como vimos no capítulo anterior, temos

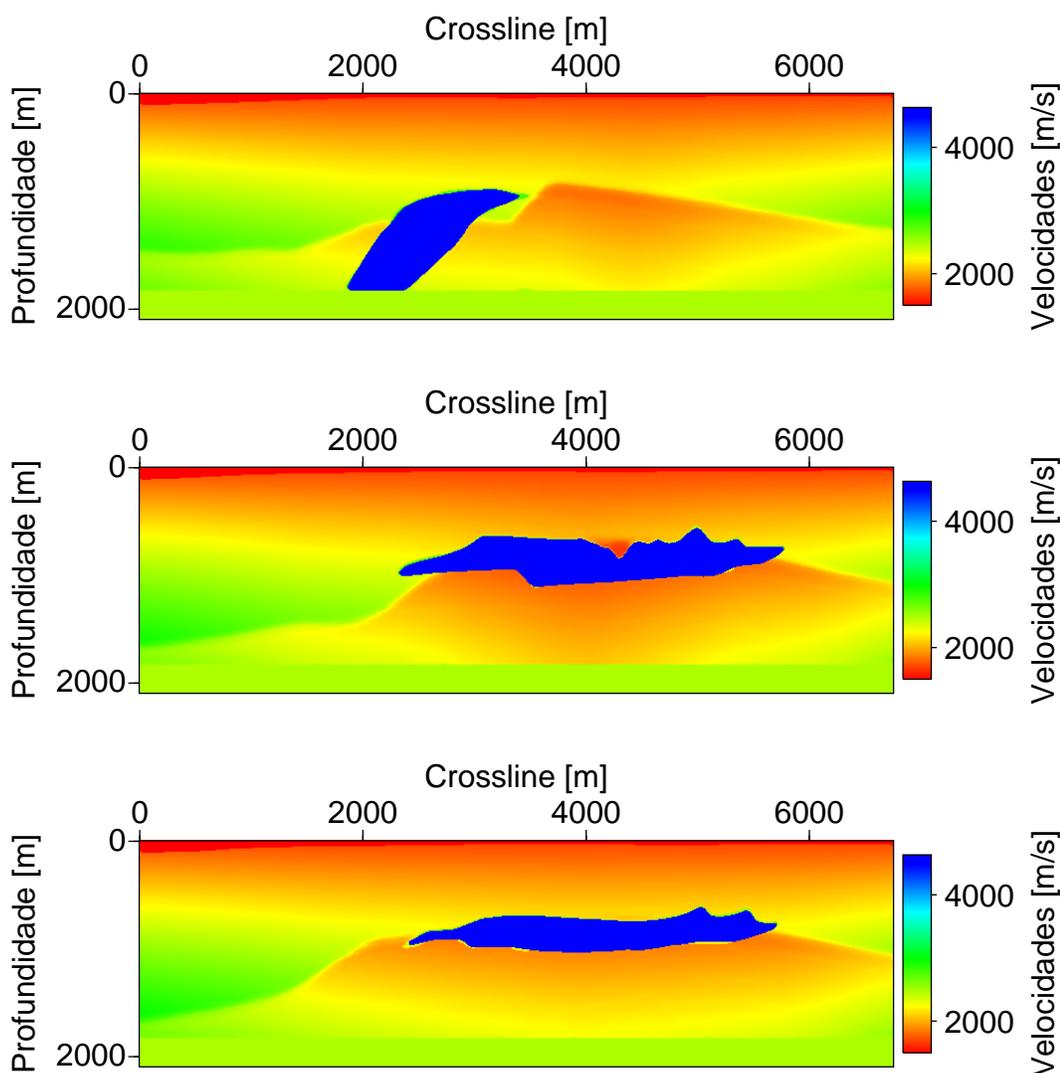


Figura 5.2: Seções verticais do modelo de sal SEG/EAGE em planos paralelos a direção *crossline*. Em cima: em $x_1 = 1690$ m. No meio: em $x_1 = 3380$ m. Embaixo: em $x_1 = 4160$ m.

as três seguintes maneiras de melhorar o alcance do algoritmo de migração FD iterativo para migrar eventos com ângulos de mergulho acima de 45° : usando menores ângulos de rotação da linha de corte, otimizando os coeficientes de Padé complexo ou usando mais termos da série de Padé complexa. Essa melhora para migrar eventos com ângulos de mergulho acima de 45° usando essas três maneiras, se confirma na comparação das seções verticais das resposta ao impulso do algoritmo de migração FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte de 25° (no meio e a esquerda de ambas as figuras) e 25° otimizado (no meio e a direita de ambas as figuras), e do algoritmo de migração FD iterativo com 2 termos da série de Padé complexa com ângulo de rotação da linha de corte de 45° (embaixo a esquerda de ambas as figuras)

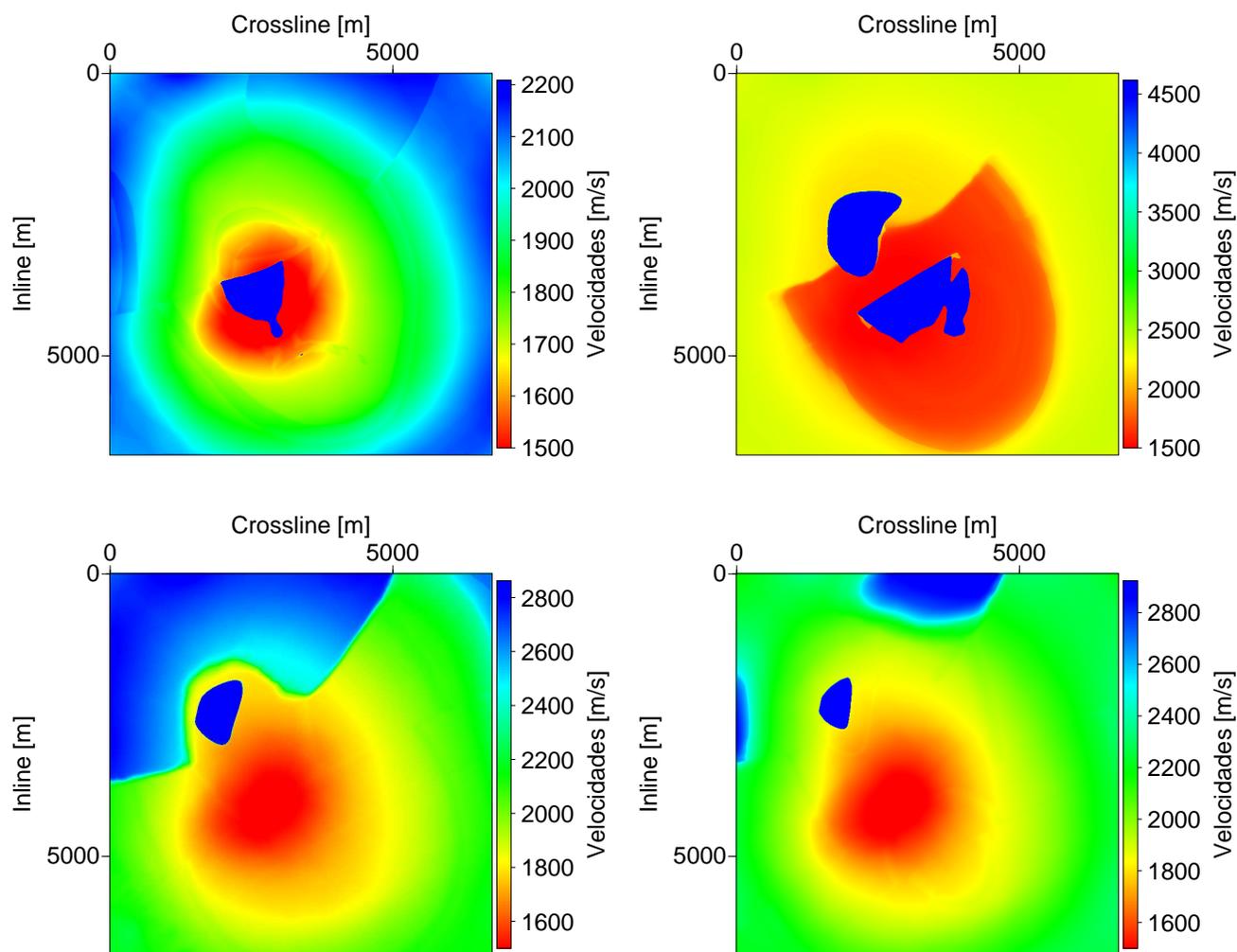


Figura 5.3: Seções horizontais do modelo de sal SEG/EAGE em planos paralelos ao plano x_1x_2 . Em cima na esquerda: em $x_3 = 550$ m. Em cima na direita: em $x_3 = 1050$ m. Em baixo na esquerda: em $x_3 = 1350$ m. Em baixo na direita: em $x_3 = 1550$ m.

com as suas respectivas respostas ao impulso do algoritmo de migração RTM (embaixo a direita de ambas as figuras). Observamos também que as amplitudes relativas dos eventos migrados com o algoritmo FD iterativo e FD MUMPS de ambas as figuras, não correspondem as amplitudes relativas dos eventos migrados com algoritmo RTM. Como falamos nos capítulos anteriores, isso é devido ao fato que a equação da onda unidirecional não é correta para fornecer as amplitudes das ondas.

As conclusões da análise que foi realizada no par de Figura (5.4), (5.5), se aplicam para os pares de Figuras (5.6), (5.7) e (5.8), (5.9). Onde no par de Figuras (5.6), (5.7), estamos mostrando as seções verticais da resposta ao impulso calculado com o algoritmo de migração FD iterativo, FD MUMPS e RTM em planos paralelos as direções *inline* em $x_2 = 3380$ m (Figura 5.6) e *crossline* em $x_1 = 3380$ m (Figura 5.7). Nas Figuras (5.8) e (5.9),

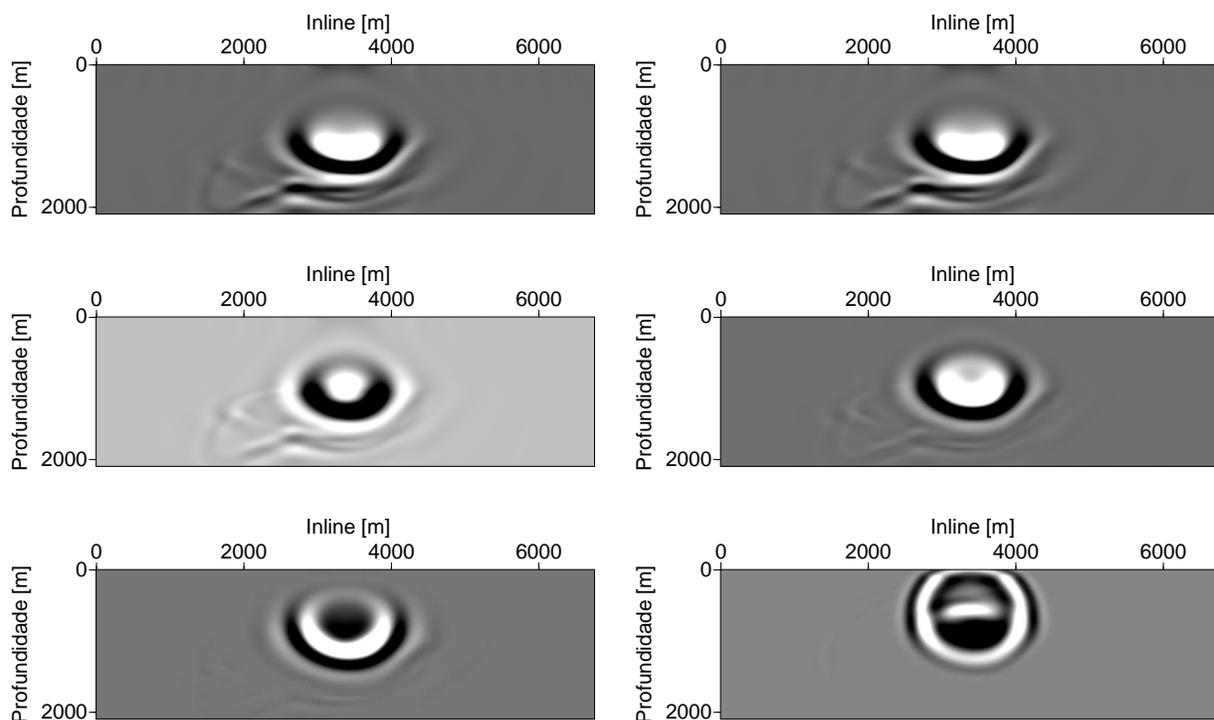


Figura 5.4: Seções da resposta ao impulso 3D no plano paralelo ao plano x_1x_3 em $x_2 = 1690$ m. Em cima na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 45^\circ$. Em cima na direita: FD-MUMPS com 1 termo da série de Padé complexa e $\theta = 45^\circ$. No meio na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 25^\circ$. No meio na direita: FD iterativo com 1 termo da série de Padé complexa otimizada e $\theta = 25^\circ$. Embaixo na esquerda: FD iterativo com 2 termos da série de Padé complexa e $\theta = 45^\circ$. Embaixo na direita: RTM.

estamos mostrando as seções verticais da resposta ao impulso calculado com o algoritmo de migração FD iterativo, FD MUMPS e RTM em planos paralelos as direções *inline* em $x_2 = 4160$ m (Figura 5.8) e *crossline* em $x_1 = 4160$ m (Figura 5.9). A forma como estas seções verticais da resposta ao impulso dos algoritmos de migração FD iterativo, FD MUMPS e RTM, estão organizadas nos pares de Figuras (5.6), (5.7) e (5.8), (5.9), seguem o mesmo padrão do par de Figuras (5.4), (5.5).

Nas Figuras (5.10), (5.11), (5.12) e (5.13), mostramos as seções horizontais da resposta ao impulso calculado com os algoritmos de migração FD iterativo, FD MUMPS e RTM nas profundidades $x_3 = 550$ m (Figura 5.10), $x_3 = 1050$ m (Figura 5.11), $x_3 = 1350$ m (Figura 5.12) e $x_3 = 1550$ m (Figura 5.13). Na parte de cima das 4 figuras, a esquerda, mostramos a resposta ao impulso do algoritmo de migração FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte de 45° , a direita, a resposta ao impulso com o algoritmo de migração FD MUMPS com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte de 45° . No meio das 4 figuras, a esquerda, mostramos a resposta ao impulso do algoritmo de migração FD iterativo

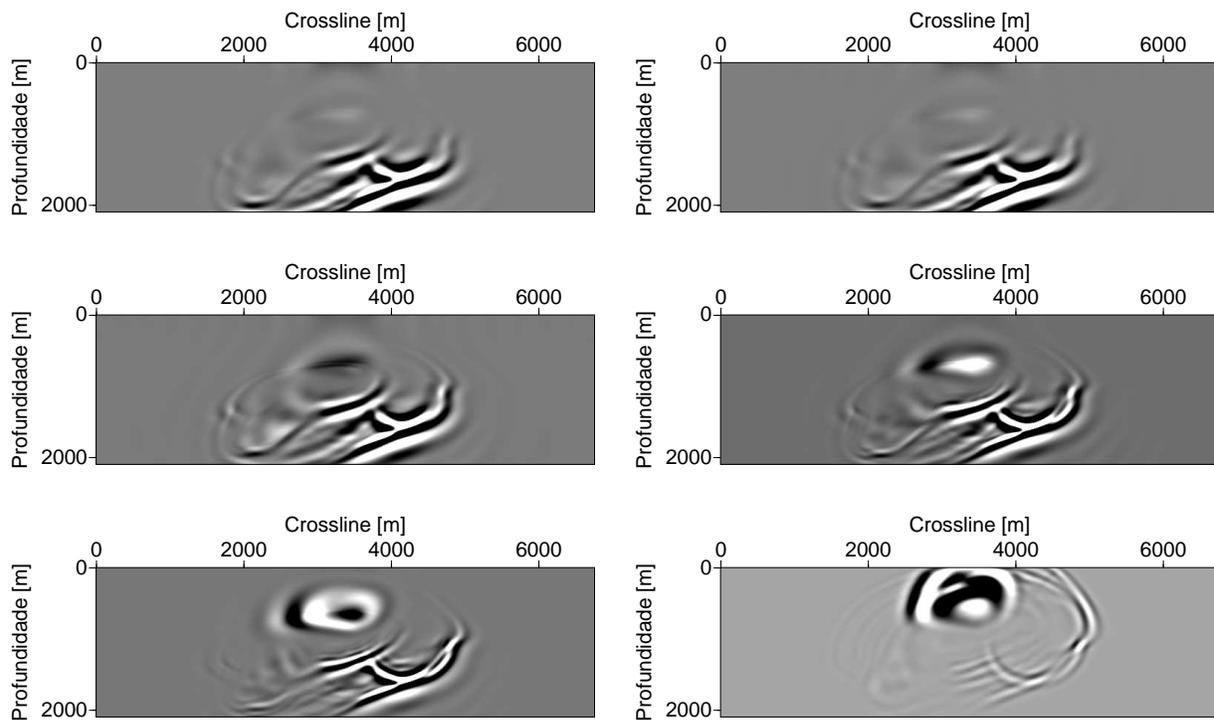


Figura 5.5: Seções da resposta ao impulso 3D no plano paralelo ao plano x_2x_3 em $x_1 = 1690$ m. Em cima na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 45^\circ$. Em cima na direita: FD-MUMPS com 1 termo da série de Padé complexa e $\theta = 45^\circ$. No meio na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 25^\circ$. No meio na direita: FD iterativo com 1 termo da série de Padé complexa otimizado e $\theta = 25^\circ$. Embaixo na esquerda: FD iterativo com 2 termos da série de Padé complexa e $\theta = 45^\circ$. Embaixo na direita: RTM.

com 1 termo da série de Padé complexa e ângulo de rotação da linha de corte de 25° e a direita com ângulo de rotação da linha de corte de 25° otimizado. Na parte de baixo das 4 figuras, a esquerda, mostramos a resposta ao impulso do algoritmo de migração FD iterativo com 2 termos da série de Padé complexa com ângulo de rotação da linha de corte de 45° , a direita, a resposta ao impulso do algoritmo de migração RTM. Vemos na Figura (5.10), que as resposta ao impulso calculado com os algoritmos de migração FD iterativo (em cima a esquerda) e FD MUMPS (em cima a direita), ambos com 1 termo da série de Padé complexa com ângulos de rotação da linha de corte de 45° e o algoritmo de migração FD iterativo com 1 termo da série de Padé complexa com ângulo da linha de corte de 25° (no meio a direita), não migraram os eventos que aparecem na resposta ao impulso calculado com o algoritmo de migração RTM (embaixo a direita). Isso é devido a limitação desses algoritmos em relação ao ângulo de mergulho dos refletores, pois na profundidade que foi feito essa seção horizontal, o ângulo de mergulho dos refletores é bem maior que 45° . Por outro lado, as seções da resposta ao impulso do algoritmo de migração FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte de 25° otimizado e do algoritmo de migração FD iterativo com 2 termos da

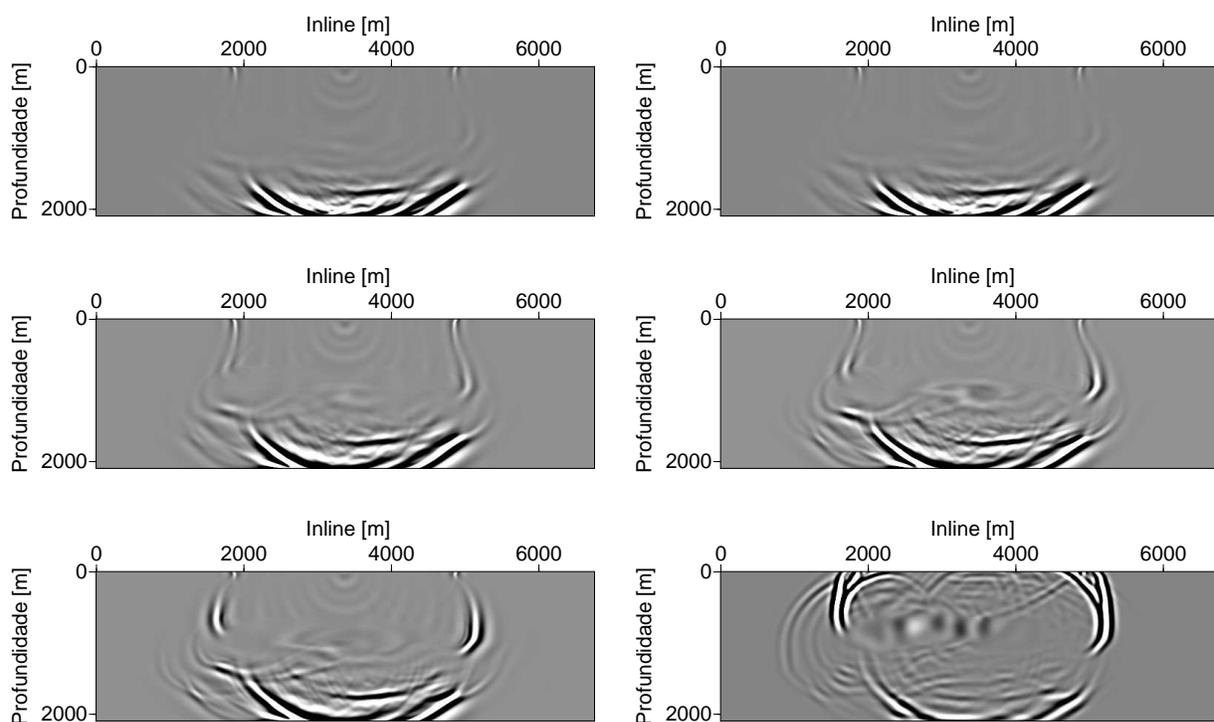


Figura 5.6: Seções da resposta ao impulso 3D no plano paralelo ao plano x_1x_3 em $x_2 = 3380$ m. Em cima na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 45^\circ$. Em cima na direita: FD-MUMPS com 1 termo da série de Padé complexa e $\theta = 45^\circ$. No meio na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 25^\circ$. No meio na direita: FD iterativo com 1 termo da série de Padé complexa otimizado e $\theta = 25^\circ$. Embaixo na esquerda: FD iterativo com 2 termos da série de Padé complexa e $\theta = 45^\circ$. Embaixo na direita: RTM.

série de Padé complexa com ângulo de rotação da linha de corte de 45° , que são melhores para migrar refletores com maiores mergulho, conseguem imagear melhor os eventos que aparecem na resposta ao impulso calculado com o algoritmo de migração RTM. Nas outras 3 figuras restantes, vemos que os eventos que existem na resposta ao impulso calculado com o algoritmo de migração RTM são também migrados nos algoritmos de migração FD iterativo e FD MUMPS. Porém, percebemos que usando o algoritmo de migração FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte de 25° os eventos são mais parecidos com os eventos do RTM, por sua vez, usando esse mesmo algoritmo de migração, mas agora com ângulo de rotação da linha de corte de 25° otimizado, os eventos migrados são ainda mais parecidos com os eventos migrados com o algoritmo RTM do que usando ângulo de rotação da linha de corte de 25° . Por fim, vemos que a melhor migração é usando o algoritmo de migração FD iterativo com 2 termos da série de Padé complexa com ângulo de rotação da linha de corte de 45° . Isso mais uma vez, vem confirmar a eficácia das três técnicas discutidas no capítulo anterior para melhorar o alcance do algoritmo de migração FD iterativo para migrar eventos com ângulo de mergulho acima de 45° . Nessas figuras, também observamos o problema que

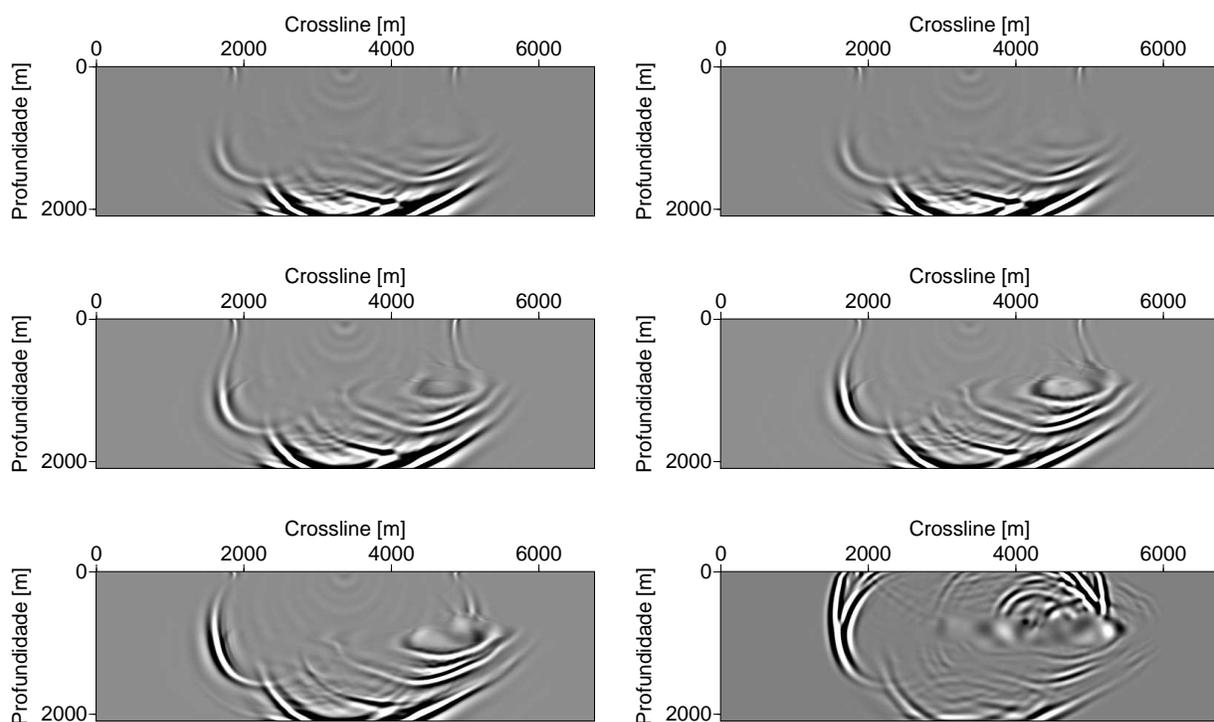


Figura 5.7: Seções da resposta ao impulso 3D no plano paralelo ao plano x_2x_3 em $x_1 = 3380$ m. Em cima na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 45^\circ$. Em cima na direita: FD-MUMPS com 1 termo da série de Padé complexa e $\theta = 45^\circ$. No meio na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 25^\circ$. No meio na direita: FD iterativo com 1 termo da série de Padé complexa otimizado e $\theta = 25^\circ$. Embaixo na esquerda: FD iterativo com 2 termos da série de Padé complexa e $\theta = 45^\circ$. Embaixo na direita: RTM.

foi ressaltado sobre as amplitudes relativas dos eventos migrados.

ÚLTIMAS CONSIDERAÇÕES DO CAPÍTULO

Neste capítulo, avaliamos a performance do algoritmo de migração FD iterativo usando o modelo de sal SEG/EAGE. Vimos que calculando a resposta ao impulso usando o algoritmo de migração FD iterativo com 1 termo da série de Padé complexa com ângulo de rotação da linha de corte de 45° , esse algoritmo consegue migrar eventos que tem ângulo de mergulho abaixo de 45° . Isso mostra que esse algoritmo cumpriu seu objetivo, pois é uma limitação intrínseca desse algoritmo, com esses parâmetros, migrar eventos com ângulo de mergulho abaixo de 45° . Avaliamos também as três técnicas discutidas no capítulo anterior para melhorar o alcance do algoritmo para migrar eventos com maiores ângulos de mergulho. As três técnicas se mostraram eficazes, principalmente a que usa dois termos da série de Padé complexa. Por outro lado, essas técnicas acarretam um custo computacional maior, como vimos no Capítulo 2, na

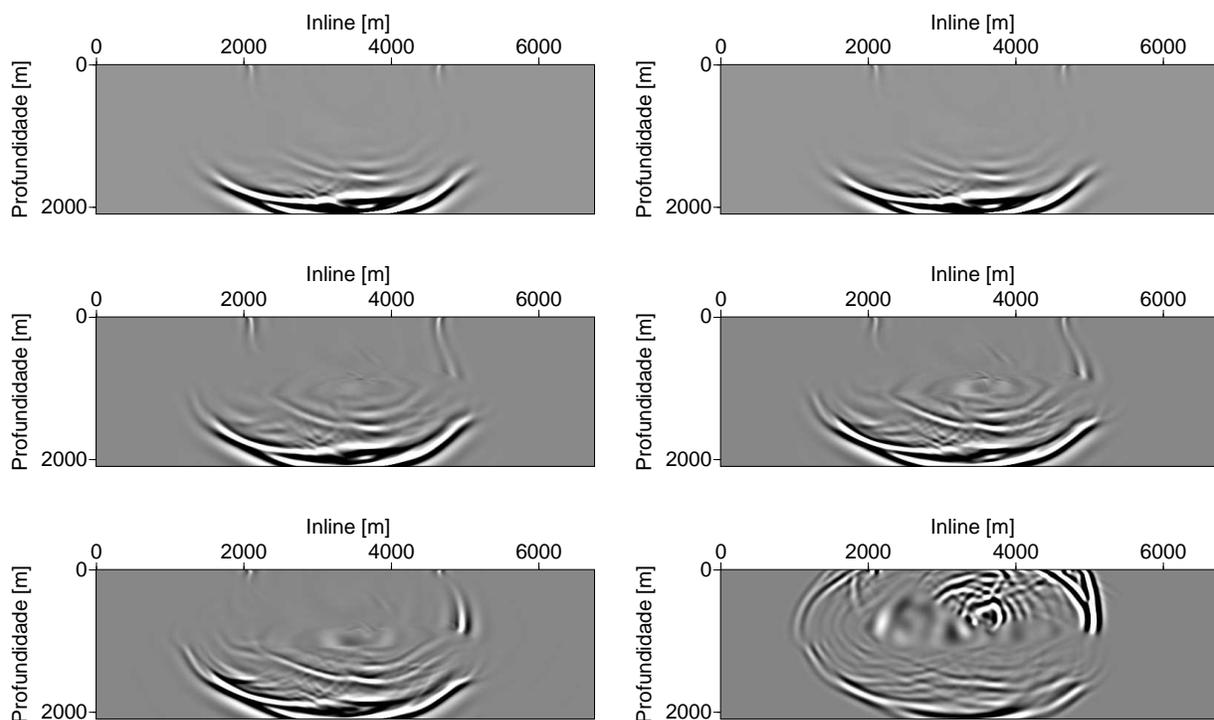


Figura 5.8: Seções da resposta ao impulso 3D no plano paralelo ao plano x_1x_3 em $x_2 = 4160$ m. Em cima na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 45^\circ$. Em cima na direita: FD-MUMPS com 1 termo da série de Padé complexa e $\theta = 45^\circ$. No meio na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 25^\circ$. No meio na direita: FD iterativo com 1 termo da série de Padé complexa otimizado e $\theta = 25^\circ$. Embaixo na esquerda: FD iterativo com 2 termos da série de Padé complexa e $\theta = 45^\circ$. Embaixo na direita: RTM.

análise da convergência do algoritmo Bi-CGSTAB. Na Figura (5.14), mostramos o tempo de CPU que os algoritmos de migração FD MUMPS e FD iterativo levaram para calcular as respostas ao impulso analisadas neste capítulo. Avaliando esse gráfico, podemos ver que o algoritmo FD iterativo usando 1 termo da série de Padé complexa com ângulo de rotação de linha de corte de 45° , 25° ou 25° otimizado é bem mais eficiente computacionalmente que o algoritmo FD MUMPS. Principalmente na comparação entre o tempo de CPU que os algoritmos de migração FD MUMPS e FD iterativo levaram para calcular a resposta ao impulso usando 1 termo da série de Padé complexa com ângulo de rotação da linha de corte de 45° . Onde nessa comparação, podemos ver que o tempo

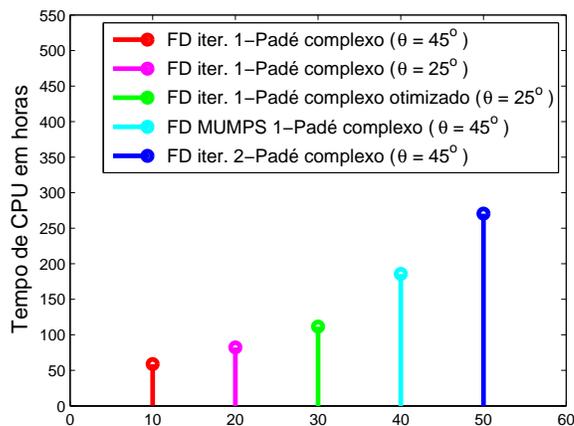


Figura 5.14: Tempo de CPU do algoritmo de migração FD para calcular a resposta ao impulso usando o modelo de sal da SEG/EAGE

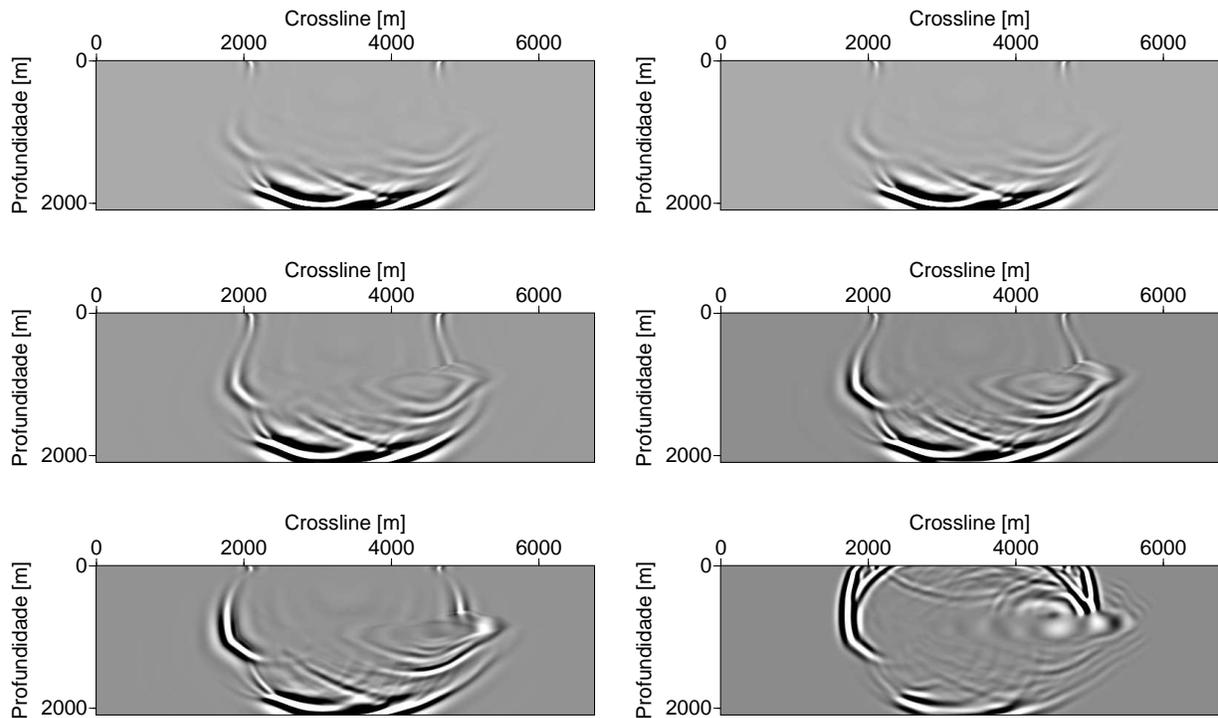


Figura 5.9: Seções da resposta ao impulso 3D no plano paralelo ao plano x_2x_3 em $x_1 = 4160$ m. Em cima na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 45^\circ$. Em cima na direita: FD-MUMPS com 1 termo da série de Padé complexa e $\theta = 45^\circ$. No meio na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 25^\circ$. No meio na direita: FD iterativo com 1 termo da série de Padé complexa otimizado e $\theta = 25^\circ$. Embaixo na esquerda: FD iterativo com 2 termos da série de Padé complexa e $\theta = 45^\circ$. Embaixo na direita: RTM.

de CPU do algoritmo FD MUMPS é mais que três vezes maior que o tempo de CPU do FD iterativo. Vemos também que o algoritmo de migração FD iterativo com 1 termo da série de Padé complexa otimizado com ângulo de rotação da linha de corte de 25° , é um algoritmo que tem uma resposta ao impulso quase tão boa quanto usando 2 termos da série de Padé complexa, mas tem um custo computacional bem menor que usando dois termos da série de Padé complexa.

Ressaltamos que não fizemos nenhum teste usando 3 termos da série de Padé, pois apesar do algoritmo convergir, há uma grande perda da eficiência do algoritmo iterativo Bi-CGSTAB para calcular a solução do sistema linear do algoritmo de migração FD iterativo.

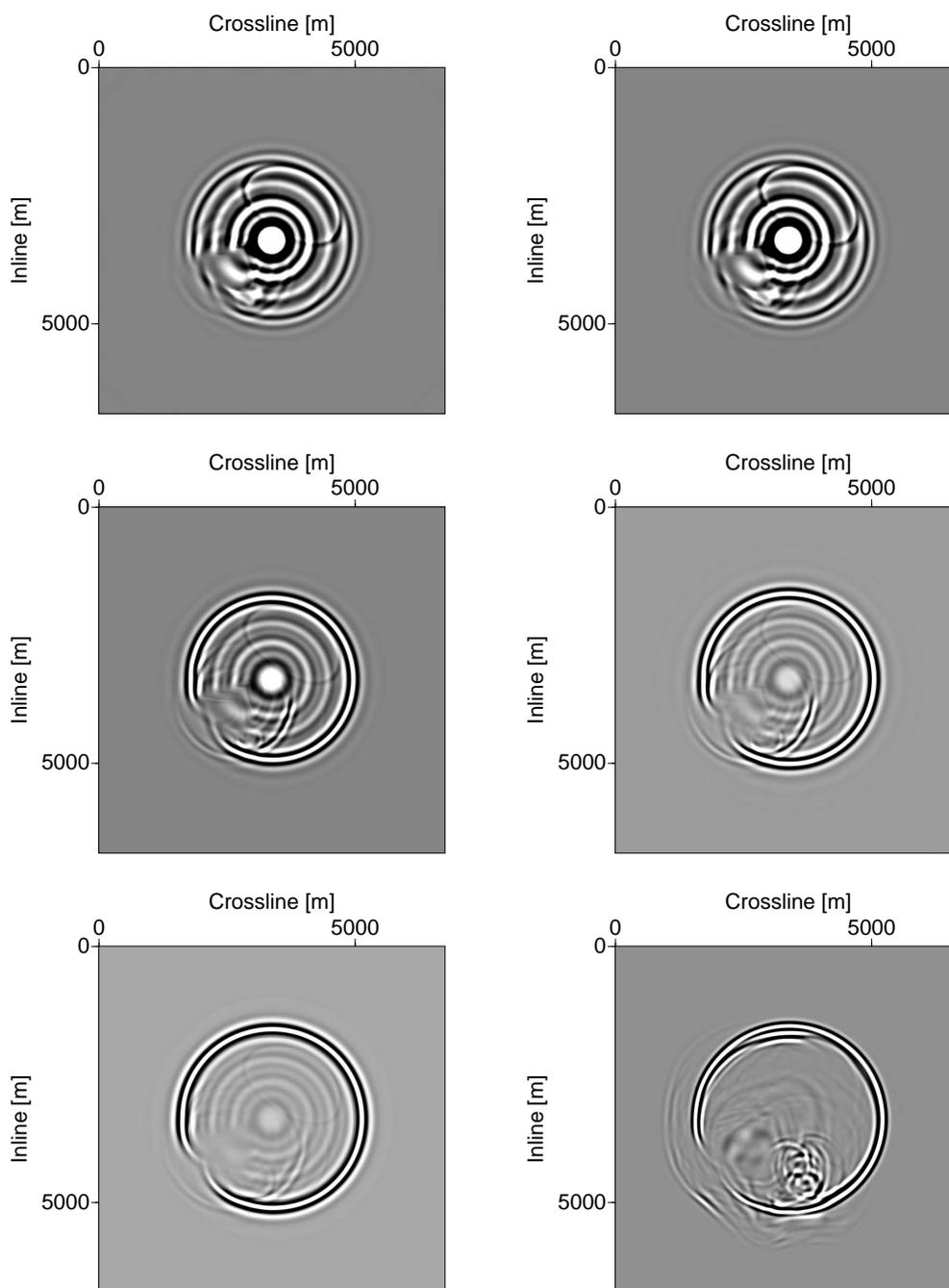


Figura 5.10: Seções da resposta ao impulso 3D no plano paralelo ao plano x_1x_2 em $x_3 = 550$ m. Em cima na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 45^\circ$. Em cima na direita: FD-MUMPS com 1 termo da série de Padé complexa e $\theta = 45^\circ$. No meio na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 25^\circ$. No meio na direita: FD iterativo com 1 termo da série de Padé complexa otimizado e $\theta = 25^\circ$. Embaixo na esquerda: FD iterativo com 2 termos da série de Padé complexa e $\theta = 45^\circ$. Embaixo na direita: RTM.

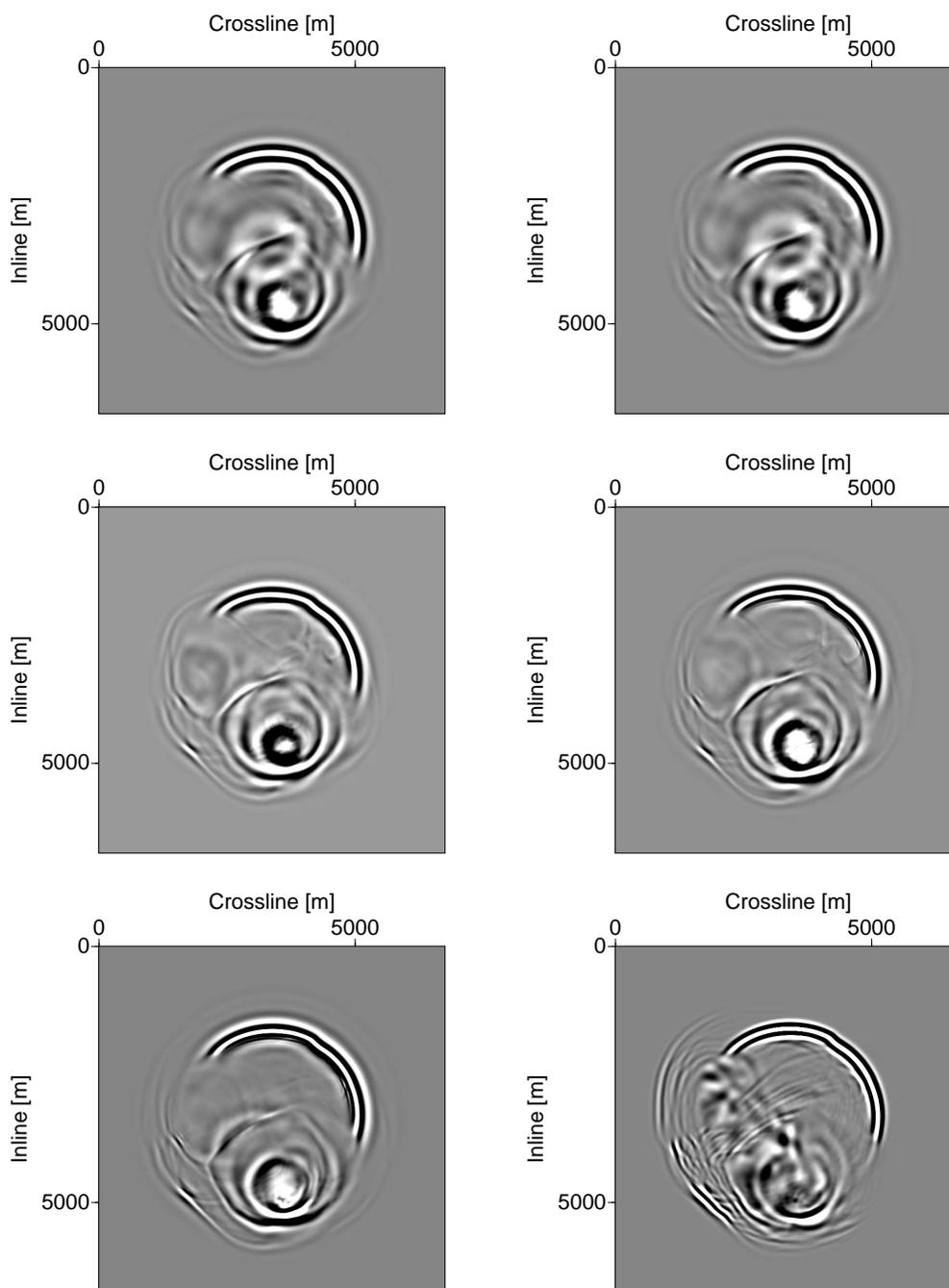


Figura 5.11: Seções da resposta ao impulso 3D no plano paralelo ao plano x_1x_2 em $x_3 = 1050$ m. Em cima na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 45^\circ$. Em cima na direita: FD-MUMPS com 1 termo da série de Padé complexa e $\theta = 45^\circ$. No meio na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 25^\circ$. No meio na direita: FD iterativo com 1 termo da série de Padé complexa otimizado e $\theta = 25^\circ$. Embaixo na esquerda: FD iterativo com 2 termos da série de Padé complexa e $\theta = 45^\circ$. Embaixo na direita: RTM.

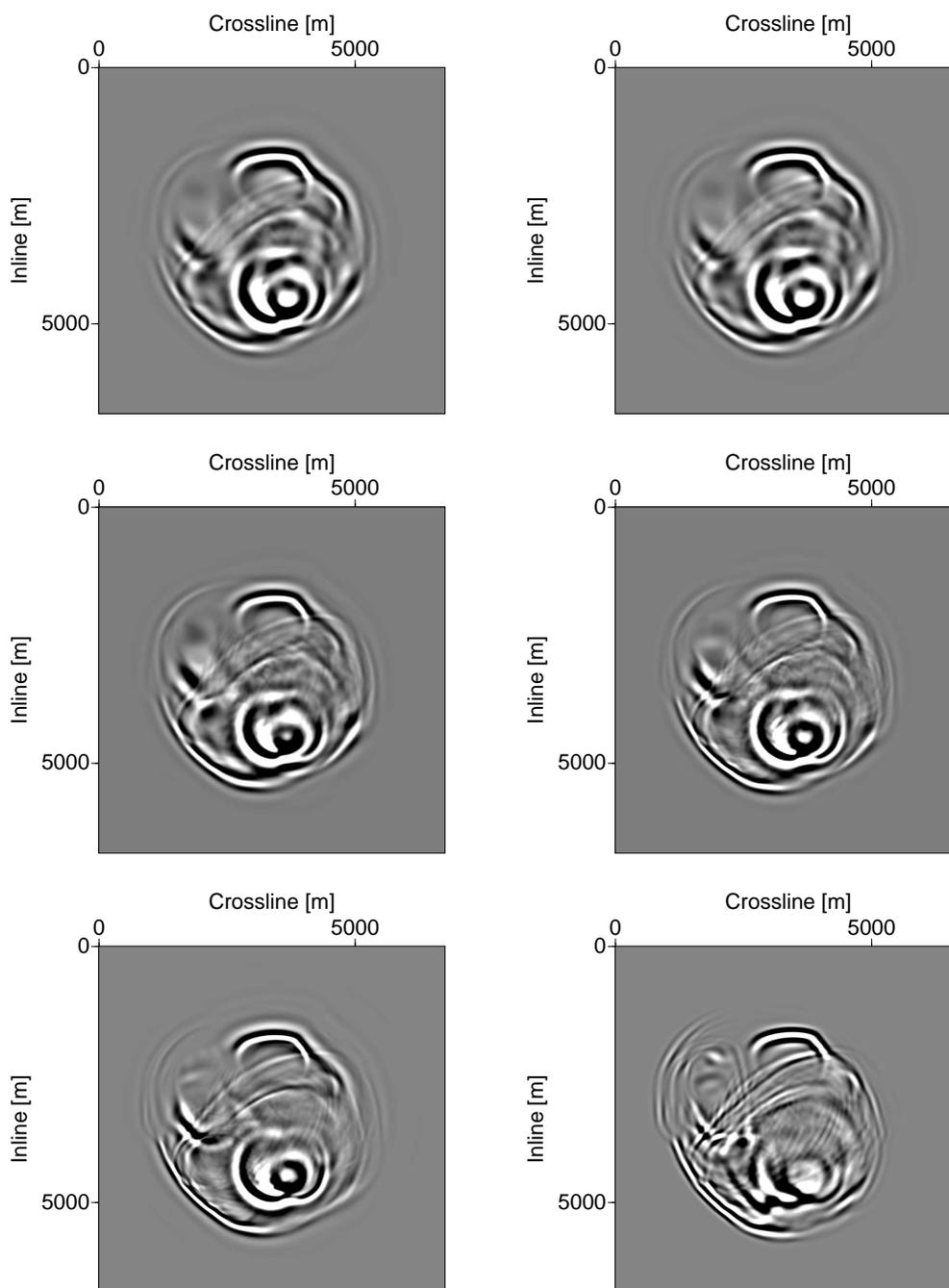


Figura 5.12: Seções da resposta ao impulso 3D no plano paralelo ao plano x_1x_2 em $x_3 = 1350$ m. Em cima na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 45^\circ$. Em cima na direita: FD-MUMPS com 1 termo da série de Padé complexa e $\theta = 45^\circ$. No meio na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 25^\circ$. No meio na direita: FD iterativo com 1 termo da série de Padé complexa otimizado e $\theta = 25^\circ$. Embaixo na esquerda: FD iterativo com 2 termos da série de Padé complexa e $\theta = 45^\circ$. Embaixo na direita: RTM.

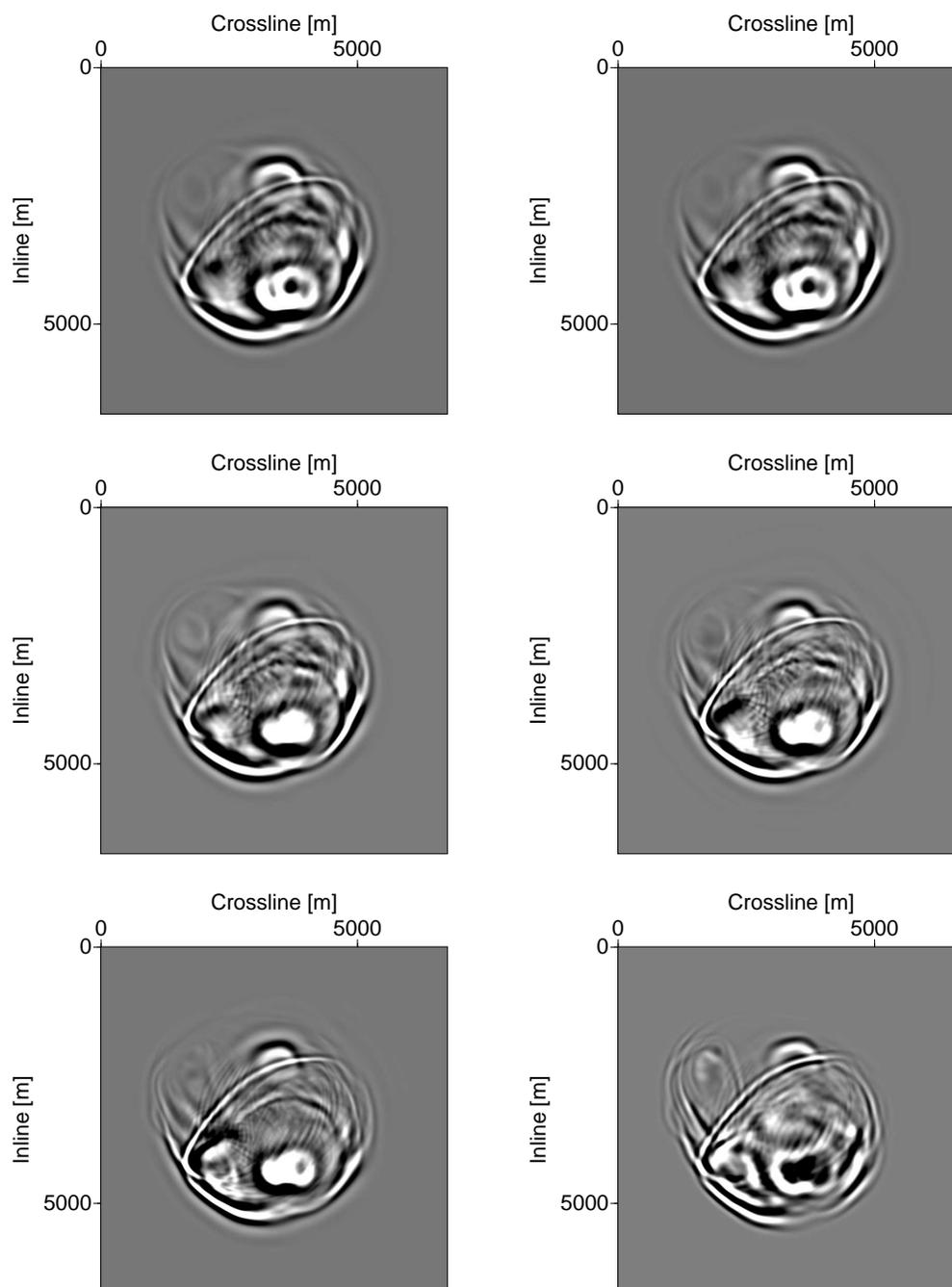


Figura 5.13: Seções da resposta ao impulso 3D no plano paralelo ao plano x_1x_2 em $x_3 = 1550$ m. Em cima na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 45^\circ$. Em cima na direita: FD-MUMPS com 1 termo da série de Padé complexa e $\theta = 45^\circ$. No meio na esquerda: FD iterativo com 1 termo da série de Padé complexa e $\theta = 25^\circ$. No meio na direita: FD iterativo com 1 termo da série de Padé complexa otimizado e $\theta = 25^\circ$. Embaixo na esquerda: FD iterativo com 2 termos da série de Padé complexa e $\theta = 45^\circ$. Embaixo na direita: RTM.

6 CONCLUSÕES

A performance do método iterativo utilizado na implementação do operador de extrapolação do campo de onda para baixo 3D sem *splitting* nas direções *inline* e *crossline* tem uma melhora expressiva usando expansão de Padé complexa. Para avaliar essa melhora na eficiência computacional do método iterativo, realizamos uma análise simples que indica que o uso dos coeficientes de Padé complexo deixa o operador de extrapolação do campo de onda para baixo melhor condicionado do que usando os coeficientes de Padé real, especialmente quando o campo de onda está sendo extrapolado em um meio de altas velocidades.

Nos testes numéricos usando modelo homogêneo, mostramos a eficácia do uso do PML no operador de extrapolação do campo de onda para baixo em evitar reflexões nas bordas do plano horizontal.

Os testes numéricos usando o modelo de sal SEG/EAGE mostram a viabilidade desse operador para meios onde o modelo de velocidade é complexo. Nesses testes numéricos comparamos a resposta ao impulso desse algoritmo de migração, com a resposta ao impulso do algoritmo de migração FD MUMPS. Nessa comparação, a resposta ao impulso de ambos os algoritmos de migração são praticamente iguais, mostrando assim que o Bi-CGSTAB é tão bom quanto o MUMPS para resolver o sistema linear. Porém, usando um único termo da série de Padé complexa, o Bi-CGSTAB é bem mais eficiente computacionalmente (em termos de memória alocada e tempo de CPU) que o MUMPS. A partir de dois termos da série de Padé complexa o MUMPS é mais eficiente computacionalmente (em termo de tempo de CPU) que o Bi-CGSTAB. Desses testes numéricos, vimos que usando apenas um termo da série de Padé complexa otimizado com ângulo de rotação da linha de corte de 25° no operador de extrapolação do campo de onda para baixo, temos uma resposta ao impulso quase igual à resposta ao impulso usando dois termos da série de Padé complexa, porém com um custo computacional menor que usando dois termos da série de Padé complexa.

A eficiência da solução do operador de extrapolação do campo de onda para baixo sem fazer *splittin* nas direções *inline* e *crossline* ainda continua ser um problema. Nos testes numéricos indicam que a combinação da aproximação de Padé complexa com um preconditionador pode reduzir ainda mais o custo computacional, deixando assim essa solução como uma opção mais sofisticada para corrigir o problema da anisotropia numérica.

Como perspectivas de trabalhos futuros, iremos estudar melhores preconditionadores para acelerar a velocidade de convergência do método iterativo, especialmente para baixas frequências, onde a performance do Bi-CGSTAB cai rapidamente. Iremos também investigar o desempenho de métodos iterativos mais modernos, como a família do método iterativo IDR, para resolver o sistema linear do operador de extrapolação do campo de onda para baixo.

REFERÊNCIAS

- AMAZONAS, D. R. et al. Wide-angle FD and FFD migration using complex Padé approximations. *Geophysics*, v. 72, n. 6, p. PS215–PS220, 2007.
- AMESTOY, P. R. et al. A full asynchronous multifrontal solver using distributed dynamic scheduling. *SIAM Journal fo matrix analysis and applications*, v. 23, p. 15–41, 2001.
- AMESTOY, P. R. et al. Hybrid scheduling for the parallel solution of linear systems. *Parallel computing*, v. 32, p. 136–141, 2006.
- BÉRENGER, J. A perfectly matched layer for the absorption of electromagnetic waves. *J. Comput. Phys.*, v. 114, p. 185–200, 1994.
- BIONDI, B. *3D seismic imaging*. [S.l.]: Society of Exploration Geophysicists, 2006.
- BROWN, D. L. Applications of operator separation in reflection seismology. *Geophysics*, v. 48, n. 3, p. 288–294, 1983.
- CLAERBOUT, J. *Imaging the Earth's Interior*. [S.l.]: Blackwell Scientific Publications, 1985.
- COLE, S. Iterative methods for 3-d finite-difference migration and modeling. Stanford University, 1989. (Technical report).
- COLLINO, F.; TSOGKA, C. Application of the perfectly matched absorbing layer model to the linear elastodynamic problem in anisotropic heterogeneous media. *Geophysics*, v. 66, n. 1, p. 294–307, 2001.
- ERLANGGA, Y. A. *A robust and efficient iterative method for the numerical solution of the Helmholtz equation*. Delft University: fl.126, 2005. Tese (Doutorado).
- GAZDAG, J. Wave equation migration with the phase-shift method. *Geophysics*, v. 43, n. 7, p. 1342–1351, 1978.
- GIJZEN, M. B. V.; SONNEVELD, P. An elegant idr(s) variant that efficiently exploits bi-orthogonality properties. Reports of department of applied mathematical analysis, Technical report, 2010. (Technical report).
- HESTENES, M. R.; STIEFEL, E. Method of conjugate gradients for solving linear systems. *J. Res. Nat. Bur. Standards*, v. 49, p. 409–436, 1952.
- ISERLES, A. *A first course in the numerical analysis of differential equations*. [S.l.]: Cambridge University Press, 1996.

- LANCZOS, C. An iteration method for the solution of the eigenvalue problem of linear differential and integral operator. *J. Res. Nat. Bur. Standards*, v. 45, p. 255–282, 1950.
- LI, Z. Compensating finite-difference errors in 3-d migration and modeling. *Geophysics*, v. 56, n. 10, p. 1650–1660, 1991.
- MILINAZZO, F. A.; ZALA, C. A.; BROOKE, G. H. Rational square-root approximations for parabolic equation algorithms. *J. Acoust. Soc. Am.*, v. 101, p. 760–766, 1997.
- NEKLYUDOV, D.; SILVESTROV, I.; TCHEVERDA, V. A helmholtz iterative solver with semianalytical preconditioner for the frequency-domain full-waveform inversion. In: 68TH ANN. INTERNAT. MEETING, SOCIETY OF EXPLORATION GEOPHYSICISTS. Denver, 2010. p. 1070–1074.
- NICHOLS, D. 3-d depth migration by a predictor-corrector method. Stanford University, 1991. (Technical report).
- ONOUE, Y.; FUJINO, S.; NAKASHIMA, N. An overview of a family of new iterative methods based on idr theorem and its estimation. In: IMECS. INTERNATIONAL MULTI-CONFERENCE OF ENGINEERS AND COMPUTER SCIENTISTS, 2009, Hong Kong. **Proceedings**. Hong Kong, 2009. p. 129–134.
- OP 'T ROOT, T. J. P. M.; STOLK, C. C. One-way wave propagation with amplitude based on pseudo-differential operators. *Wave Motion*, v. 47, p. 67–84, 2010.
- PLESSIX, R. E. A helmholtz iterative solver for 3d seismic-imaging problems. *Geophysics*, v. 72, n. 5, p. SM185–SM194, 2007.
- PLESSIX, R. E. Tree-dimensional frequency-domain full-waveform inversion with an iterative solver. *Geophysics*, v. 74, n. 6, p. WCC149–WCC157, 2009.
- RISTOW, D.; RÜHL, T. 3-d implicit finite-difference migration by multiway splitting. *Geophysics*, v. 62, n. 2, p. 554–567, 1997.
- RIYANTI, C. D. et al. A new iterative solver for the time-harmonic wave equation. *Geophysics*, v. 71, n. 5, p. E57–E63, 2006.
- SONNEVELD, P. Cgs: A fast lanczos-type solver for nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, v. 10, p. 36–52, 1987.
- SONNEVELD, P. On the convergence behaviour of idr(s). Reports of department of applied mathematical analysis, Delft University, 2010. (Technical report).

SONNEVELD, P.; GIJZEN, M. B. V. Idr(s): a family of simple and fast algorithms for solving large nonsymmetric linear systems. *SIAM J. Sci. Comp.*, v. 31, n. 2, p. 1035–1062, 2008.

VAN DE VORST, H. A. Bi-cgstab: A fast and smoothly converging variant of bi-cg for the solution of nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput*, v. 13, n. 2, p. 631–644, 1992.

VIVAS, F. A.; PESTANA, R. C. True-amplitude one-way wave equation migration in the mixed domain. *Geophysics*, v. 75, n. 5, p. 199–209, 2010.

ZHANG, Y.; ZHANG, G.; BLEISTEIN, N. Theory of true-amplitude one-way wave equation and true-amplitude common-shot migration. *Geophysics*, v. 70, p. E1–E10, 2005.

APÊNDICE

APÊNDICE A – MÉTODOS ITERATIVOS

Neste apêndice iremos discutir alguns métodos iterativos para resolver sistemas lineares da forma

$$\mathbf{A}\mathbf{u} = \mathbf{y}, \quad (\text{A.1})$$

onde $\mathbf{A} \in \mathbb{R}^{N \times N}$ é uma matriz não singular, simétrica e positiva-definida, $\mathbf{u} \in \mathbb{R}^N$ e $\mathbf{y} \in \mathbb{R}^N$ são vetores colunas.

A.1 MÉTODOS ITERATIVOS BÁSICOS

Métodos iterativos básicos são métodos derivados a partir de uma decomposição apropriada da matriz original do sistema linear, no nosso caso a matriz \mathbf{A} , em outras matrizes da mesma ordem e mesmo domínio, ou seja, se $\mathbf{A} = \mathbf{H} - \mathbf{G}$, logo $\mathbf{H}, \mathbf{G} \in \mathbb{R}^{N \times N}$, pois $\mathbf{A} \in \mathbb{R}^{N \times N}$, nessa decomposição admitimos que \mathbf{H} tem inversa e será denotada por \mathbf{H}^{-1} .

Sendo assim, vamos começar considerando a decomposição da matriz original \mathbf{A} como foi feito anteriormente para reescrever a equação (A.1) da seguinte maneira:

$$\begin{aligned} \mathbf{H}\mathbf{u} &= \mathbf{y} + \mathbf{G}\mathbf{u}, \\ \mathbf{u} &= \mathbf{H}^{-1}(\mathbf{y} + \mathbf{G}\mathbf{u}), \\ \mathbf{u} &= \mathbf{H}^{-1}\mathbf{y} + (\mathbf{I} - \mathbf{H}^{-1}\mathbf{A})\mathbf{u}, \\ \mathbf{u} &= \mathbf{u} + \mathbf{H}^{-1}(\mathbf{y} - \mathbf{A}\mathbf{u}), \end{aligned} \quad (\text{A.2})$$

onde \mathbf{I} é a matriz identidade.

A solução exata da equação (A.1) é \mathbf{u} . Com o objetivo de encontrar essa solução, podemos usar a equação acima como uma fórmula de recorrência da seguinte maneira:

$$\mathbf{u}_{(i+1)} = \mathbf{u}_i + \mathbf{H}^{-1}\mathbf{r}_i, \quad (\text{A.3})$$

onde $\mathbf{r}_i \equiv \mathbf{y} - \mathbf{A}\mathbf{u}_i$ é o o vetor resíduo depois da i -ésima iteração.

Uma vez que temos uma aproximação inicial \mathbf{u}_0 podemos, a partir da equação acima gerar uma nova aproximação \mathbf{u}_1 , que esperamos que deve aproximar melhor a solução exata. Em seguida, \mathbf{u}_1 gera uma nova aproximação \mathbf{u}_2 que deve aproximar melhor a solução exata do que \mathbf{u}_1 e assim por diante. Desta forma, gera-se uma sequência de vetores \mathbf{u}_n que espera-se convergir para a solução exata \mathbf{u} . Na prática, como temos que

calcular um número finito de \mathbf{u}_n , temos que estabelecer um método de parada, onde a solução encontrada esteja suficientemente próxima da solução exata \mathbf{u} .

A equação de recorrência (A.3), é a equação do método iterativo básico. Esta equação é particularizada pela maneira que decompomos a matriz \mathbf{A} . Se $\mathbf{A} = \mathbf{D} - \mathbf{E}$, onde $\mathbf{D} = \text{diag}(\mathbf{A})$, temos o método iterativo de **Jacobi** dado por

$$\mathbf{u}_{(i+1)} = \mathbf{u}_i + \mathbf{D}^{-1}\mathbf{r}_i. \quad (\text{A.4})$$

Se $\mathbf{A} = \mathbf{L} - \mathbf{U}$, onde \mathbf{L} é uma matriz estritamente triangular inferior e \mathbf{U} é uma matriz estritamente triangular superior, obtemos o método iterativo de **Gauss-Seidel** dado por

$$\mathbf{u}_{(i+1)} = \mathbf{u}_i + \mathbf{L}^{-1}\mathbf{r}_i. \quad (\text{A.5})$$

A.2 MÉTODOS DE SUBESPAÇO DE KRYLOV

Métodos de subespaços de Krylov, são métodos de projeção para encontrar a solução aproximada de sistemas lineares como o da equação (A.1) em um subespaço específico, o subespaço de Krylov. Sendo assim, vamos de uma maneira intuitiva introduzir o subespaço de Krylov a partir da equação do método iterativo básico.

A.2.1 SUBESPAÇO DE KRYLOV

Considerando a matriz $\mathbf{H} = \mathbf{I}$ na equação (A.3), obtemos a chamada iteração de Richardson:

$$\mathbf{u}_{(i+1)} = \mathbf{u}_i + \mathbf{r}_i. \quad (\text{A.6})$$

Aplicando \mathbf{A} na equação acima e usando a definição de resíduo, obtemos uma outra relação de recorrência para o resíduo dada por:

$$\mathbf{r}_{(i+1)} = (\mathbf{I} - \mathbf{A})\mathbf{r}_i. \quad (\text{A.7})$$

Da equação (A.7) segue por indução que

$$\mathbf{r}_n = p_n(\mathbf{A})\mathbf{r}_0 \in \text{span} \{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \dots, \mathbf{A}^n\mathbf{r}_0\}, \quad (\text{A.8})$$

onde $p_n(\vartheta) = (1 - \vartheta)^n$ é um polinômio de grau exato n .

Da equação (A.6) segue por indução que

$$\mathbf{u}_n = \mathbf{u}_0 + \mathbf{r}_0 + \mathbf{r}_1 + \dots + \mathbf{r}_{n-1} = \mathbf{u}_0 + q_{n-1}(\mathbf{A}) \mathbf{r}_0, \quad (\text{A.9})$$

com q_{n-1} um polinômio de grau exato $n - 1$. Desta maneira, observamos que \mathbf{u}_n pertence ao espaço afim $\mathbf{u}_0 + \text{span} \{ \mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \dots, \mathbf{A}^{n-1}\mathbf{r}_0 \}$.

Os subespaços que obtemos na equação (A.8) e (A.9) são chamados subespaço de krylov.

A.2.2 GRADIENTE CONJUGADO

O método iterativo CG devido a Hestenes e Stiefel (1952), é uma das mais conhecidas e usadas técnicas iterativas do tipo Krylov para resolver sistemas lineares, cuja a matriz envolvida é simétrica positiva-definida, $\langle \mathbf{u}, \mathbf{A}\mathbf{u} \rangle > 0$. Estamos usando a notação $\langle \mathbf{a}, \mathbf{b} \rangle$ para denotar um produto interno hermitiano.

Uma maneira de resolver o sistema linear do tipo $\mathbf{A}\mathbf{u} = \mathbf{y}$ é encontrando uma solução aproximada \mathbf{u}_n que minimize o funcional quadrático de energia $f(\mathbf{s}) = \frac{1}{2} (\mathbf{A}\mathbf{s}, \mathbf{s}) - (\mathbf{y}, \mathbf{s})$. Uma maneira de fazer essa minimização do funcional quadrático de energia é minimizando a \mathbf{A} - norma do erro algébrico dado por: $\| \mathbf{e} \|^2_{\mathbf{A}} = \| \mathbf{u} - \mathbf{u}_n \|^2_{\mathbf{A}}$, onde $\| \mathbf{u} \|^2_{\mathbf{A}} = (\mathbf{A}\mathbf{u}, \mathbf{u})^{\frac{1}{2}}$.

Com essa ideia de minimização, temos que construir a partir de um vetor inicial \mathbf{u}_0 uma sequência de iterados $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_{n-2}, \mathbf{u}_{n-1}, \mathbf{u}_n$ que satisfaça a seguinte condição:

Condição A.1 (Minimização)

$$\| \mathbf{u} - \mathbf{u}_{n-1} \|^2_{\mathbf{A}} < \| \mathbf{u} - \mathbf{u}_{n-2} \|^2_{\mathbf{A}}$$

de tal modo que $\| \mathbf{u} - \mathbf{u}_n \|^2_{\mathbf{A}}$ convirja para o zero.

Com o objetivo de construir essa sequência de vetores usaremos a seguinte equação:

$$\mathbf{u}_{i+1} = \mathbf{u}_i + \alpha_i \mathbf{v}_i, \quad (\text{A.10})$$

onde \mathbf{v}_i é um vetor, que chamaremos vetor de busca, que nós diz a direção que temos que tomar no avanço de \mathbf{u}_i para \mathbf{u}_{i+1} de tal maneira que satisfaça a condição de minimização e $\alpha_i \in \mathbb{R}$ determina o comprimento que tem que ser esse avanço. A partir dessa equação, resulta a seguinte equação para o resíduo:

$$\mathbf{r}_{i+1} = \mathbf{r}_i - \alpha_i \mathbf{A}\mathbf{v}_i. \quad (\text{A.11})$$

Para gerarmos \mathbf{u}_{n+1} vetores que satisfaça a condição de minimização, precisamos de um subespaço $\text{span}\{\mathbf{v}_0, \mathbf{v}_1, \dots, \mathbf{v}_n\}$ e uma coleção de α_i 's que garanta esta condição de minimização de tal maneira que $\mathbf{u}_n \in \mathbf{u}_0 + \text{span}\{\mathbf{v}_0, \mathbf{v}_1, \dots, \mathbf{v}_n\}$. Uma maneira de satisfazer essa condição é calcular α_i como se segue:

$$\alpha_i = \frac{\langle \mathbf{v}_i, \mathbf{r}_i \rangle}{\langle \mathbf{v}_i, \mathbf{A}\mathbf{v}_i \rangle}. \quad (\text{A.12})$$

Uma consequência da equação (A.12) é

$$\langle \mathbf{r}_{i+1}, \mathbf{v}_i \rangle = \mathbf{0}. \quad (\text{A.13})$$

Um subespaço em particular que satisfaz a equação (A.13) é o subespaço de Krylov $\mathcal{K}_n(\mathbf{A}, \mathbf{r}_0)$. Sendo assim, podemos dizer que $\text{span}\{\mathbf{v}_0, \mathbf{v}_1, \dots, \mathbf{v}_n\} = \mathcal{K}_n(\mathbf{A}, \mathbf{r}_0)$, onde $\langle \mathbf{r}_{i+1}, \mathbf{r}_i \rangle = \mathbf{0}$ para todos \mathbf{r}_i 's. Podemos agora reescrever a equação (A.12) como

$$\alpha_i = \frac{\langle \mathbf{r}_i, \mathbf{r}_i \rangle}{\langle \mathbf{v}_i, \mathbf{A}\mathbf{v}_i \rangle}. \quad (\text{A.14})$$

Para construir os vetores \mathbf{v}_i , usaremos a seguinte relação:

$$\mathbf{v}_{i+1} = \mathbf{r}_{i+1} + \beta_i \mathbf{v}_i, \quad (\text{A.15})$$

tal que $\langle \mathbf{A}\mathbf{v}_{i+1}, \mathbf{v}_i \rangle = \mathbf{0}$ para todos \mathbf{v}_i 's e $\beta_i = \frac{\langle \mathbf{r}_{i+1}, \mathbf{r}_{i+1} \rangle}{\langle \mathbf{r}_i, \mathbf{r}_i \rangle} \in \mathbb{R}$.

Este processo de extrair a solução aproximada \mathbf{u}_n do subespaço de Krylov $\mathcal{K}_n(\mathbf{A}, \mathbf{r}_0)$ como foi discutido acima, é o que chamamos de método CG.

Segue a baixo o algoritmo CG:

Algoritmo A.1 (CG)

1. Inicialize com o vetor inicial \mathbf{u}_0 e o vetor conhecido \mathbf{y}
2. Calcule o resíduo inicial $\mathbf{r}_0 = \mathbf{y} - \mathbf{A}\mathbf{u}_0$
3. $\mathbf{v}_0 = \mathbf{r}_0$
4. Calcule a tolerância da aproximação e atribua o número máximo de iterações permitida
5. Faça até que o limite de tolerância seja atingido ou até o número máximo de iterações:
6. $\mathbf{z}_i = \mathbf{A}\mathbf{v}_i$

7. $\alpha_i = \frac{\langle \mathbf{r}_i, \mathbf{r}_i \rangle}{\langle \mathbf{v}_i, \mathbf{z}_i \rangle}$
8. $\mathbf{u}_{i+1} = \mathbf{u}_i + \alpha_i \mathbf{v}_i$
9. $\mathbf{r}_{i+1} = \mathbf{r}_i - \alpha_i \mathbf{z}_i$
10. $\beta_i = \frac{\langle \mathbf{r}_{i+1}, \mathbf{r}_{i+1} \rangle}{\langle \mathbf{r}_i, \mathbf{r}_i \rangle}$
11. $\mathbf{v}_{i+1} = \mathbf{r}_{i+1} + \beta_i \mathbf{v}_i$
12. Fim do Laço

Em um método iterativo, a eficiência computacional está relacionada com o número de produto matriz/vetor (**MV**) que o método tem que fazer para encontrar uma solução aproximada. Observamos que no algoritmo do CG tem apenas um produto **MV** o que deixa este algoritmo muito eficiente e além disso é de fácil implementação. Por outro lado, a garantia de sua convergência é apenas para uma matriz simétrica positiva definida, ou seja, se formos usar uma matriz geral, pode acontecer que este método seja divergente.

A velocidade de convergência do CG para a matriz **A** pode ser analisada pela seguinte equação:

$$\| \mathbf{u} - \mathbf{u}_n \|_A \leq 2 \left[\frac{\sqrt{\kappa(\mathbf{A})} - 1}{\sqrt{\kappa(\mathbf{A})} + 1} \right]^n \| \mathbf{u} - \mathbf{u}_0 \|_A, \quad (\text{A.16})$$

onde $\kappa(\mathbf{A}) = \lambda_{max}/\lambda_{min}$ é o número de condição da matriz **A** com λ_{max} e λ_{min} o seu maior e menor autovalor, respectivamente.

A partir da equação (A.16), vemos que quanto menor o número de condição $\mathcal{K}(\mathbf{A})$ mais rápida é a velocidade de convergência do método. Esta variação do número de condição dependerá fortemente da boa ou má condicionalidade da matriz **A**.