ANDERSON GABRIEL SANTIAGO CRAVO

ELASTOGRAFIA EM IMAGENS DE ULTRASSOM UTILIZANDO ELEMENTOS DE
CONTORNO

SÃO PAULO
2015

ANDERSON GABRIEL SANTIAGO CRAVO

ELASTOGRAFIA EM IMAGENS DE ULTRASSOM UTILIZANDO ELEMENTOS DE
CONTORNO

Tese apresentada à Escola Politécnica
da Universidade de São Paulo para
obtenção do título de Doutor em
Engenharia

SÃO PAULO
2015

ANDERSON GABRIEL SANTIAGO CRAVO

ELASTOGRAFIA EM IMAGENS DE ULTRASSOM UTILIZANDO ELEMENTOS DE
CONTORNO

Tese apresentada à Escola Politécnica
da Universidade de São Paulo para
obtenção do título de Doutor em
Engenharia

Área de Concentração:
Engenharia de Sistemas Eletrônicos

Orientador: Professor Doutor Luiz
Cezar Trintinalia

SÃO PAULO
2015

Catálogo-na-publicação

Cravo, Anderson Gabriel Santiago

Elastografia em imagens de ultrassom utilizando o método dos elementos de contorno / A.G. Santiago. -- São Paulo, 2015. 110 p.

Tese (Doutorado) - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia de Telecomunicações e Controle.

1.Elastografia virtual 2.Elementos de contorno 3.Ultrassom 4.Métodos numéricos 5.Fluxo óptico 6.Otimização material I.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia de Telecomunicações e Controle II.t.

Aos meus avós,

Hamilton Santiago e Diva Maria de Oliveira Santiago.

*“These two people are hard to find in the world. Which two?
The one who is first to do a kindness, and the one who is grateful and thankful for a kindness
done.” ”*

- The Buddha - AN 2.119

“As long as I live, so long do I learn.”

- Ramakrishna

“Some part of our being knows this - the Universe - is where we came from.

We long to return. And we can. Because the cosmos is also within us.

We’re made of star-stuff.

We are a way for the cosmos to know itself.”

- Carl Sagan

Agradecimentos

Agradeço imensamente à minha família, em especial aos meus avós Hamilton Santiago (*in memorian*) e Diva Maria de Oliveira Santiago (*in memorian*), à minha mãe Sandra, minha “irmã” Lia e meus tios, Renato e Otávio Santiago.

Agradeço ao meu orientador, professor Luiz Cezar Trintinália, pela orientação e incrível paciência comigo.

Ao meu chefe e co-orientador, professor Marco Antonio Gutierrez, pela orientação, paciência, amizade e por me apoiar neste trabalho.

Ao meu querido amigo, Fernando Antônio Tenório Fireman pelas longas conversas e apoio.

Agradeço ao pesquisador Fernando Mistuyama Cardoso, pela inestimável ajuda, sem a qual parte deste trabalho não seria possível.

Também agradeço imensamente à pesquisadora Maysa Malfiza Macedo, pela inestimável ajuda e amizade e por todos os dias de paciência e longas conversas.

Ao serviço de informática do Instituto do Coração, em especial ao “*Grupo do Insuportável*”, Adriana Palhares, Adílio Bernardes, Kátia Flávia Cândida Matheus, Mary Rose Kenchian e Priscila Matos.

Aos amigos e colegas de laboratório de P&D do Instituto do Coração, Alice Shimada Bacic, John Andrew Sims e Maria Tereza Abrahão pelos divertidos cafés da tarde, Admar Longo, Marina Rebelo e Ramon Moreno.

Às minhas queridas amigas que mesmo longe, são parte importante da minha vida, Carolina de Oliveira Gonçalves e Sandra Cartaxo.

Ao Kiko, Mi, Fumaça e Manu, pelas horas de divertimento e carinho que me proporcionam.

Agradeço ao CNPq e ao projeto Ultrassom/Finep pelo apoio financeiro.

Enfim, agradeço à todos aqueles que participaram da minha jornada. Certa vez li que mesmo um mendigo a quem negamos esmola é parte indissolúvel de nossa formação... Não há verdade maior. Algumas pessoas influenciaram minha vida de forma que sequer imaginam, com comentários ou atitudes nem sempre intencionais.

Agradeço também a todos que se detiveram alguns minutos para ler este texto...

I guess the word “thanks” is not big enough!

Lista de Figuras

1.1.1 Diagrama para elastografia virtual em imagens de ultrassom, rastreamento de ruído speckle e análise inversa.	17
2.0.1 Diagrama para cálculo de campo de deslocamento e deformações usando Lucas-Kanade, segundo Zakaria et al. [71].	21
3.2.1 Corpo sujeito à restrições de deslocamento e solicitações externas.	24
3.2.2 Corte em um plano Π e as forças internas resultantes.	24
3.2.3 Definição de tensão normal e de cisalhamento.	25
3.2.4 Campo de tensões em um elemento infinitesimal.	25
3.2.5 Vetor de força de superfície.	26
3.2.6 Modelo de uma fibra deformada.	27
3.2.7 Variação angular.	28
3.2.8 Corpo de prova e deformações associadas.	30
3.2.9 Estado plano de tensão.	31
3.2.10 Estado plano de deformação.	32
4.1.1 Malha para elementos de contorno.	34
4.1.2 Malha para elementos finitos.	35
4.2.1 Funções de interpolação para triângulos de primeira ordem.	38
4.2.2 Exemplo de malha de elementos finitos com dois elementos triangulares de primeira ordem.	40
4.2.3 Programa para montagem da matriz de rigidez global.	40
4.3.1 Domínio da Equação diferencial e seu contorno.	41
4.3.2 Normal ao contorno Γ	42
4.3.3 Ângulo interno.	45
4.3.4 Contorno discretizado.	45
4.3.5 Descontinuidade entre dois elementos adjacentes causada pelo uso de funções constantes.	46
4.3.6 Parametrização da geometria.	47
4.3.7 Condições de contorno para um problema bem posicionado.	49
4.3.8 Modelo com três sub-regiões.	50
4.3.9 Malha de elementos de contorno para o exemplo da Figura 4.3.8.	51
4.3.10 Equilíbrio de forças e compatibilidade de deslocamentos na interface de duas sub-regiões.	52

4.4.1 Malha para elementos de contorno considerando background e três lesões: (a) Malha para o background; (b) Malha para as lesões.	54
4.4.2 Malha de elementos finitos para background e três lesões.	55
4.4.3 Detalhe da malha de elementos finitos gerada a partir da malha de elementos de contorno.	55
4.4.4 Comparação entre arquiteturas de uma CPU e de uma GPU [cud].	56
4.4.5 Distribuição bidimensional de <i>threads</i> em blocos. [66]	57
4.4.6 Comparação entre CPUs e GPUs em termos de Giga FLOPS [cud].	58
4.4.7 Exemplo de código em CUDA.	59
4.4.8 Comparação entre CPU e GPU para implementação de elementos de contorno constantes [45].	60
5.1.1 Diagrama para análise inversa.	63
5.4.1 Modelagem do problema com phantom.	68
5.4.2 Gráfico de convergência para análise inversa utilizando uma simulação ideal, sem ruído.	69
5.4.3 Diagrama validação da análise inversa utilizando entrada ideal (sem ruído).	69
5.4.4 Diagrama validação da análise inversa utilizando entrada com ruído: (a) Ruído adicionado ao vetor de pressão; (b) Ruído adicionado ao vetor de deslocamentos;	70
5.4.5 Curva de convergência para simulações com adição de ruído branco gaussiano ao vetor de pressão ($SNR_{dB} = 30dB$).	71
5.4.6 Curva de convergência para simulações com adição de ruído branco gaussiano ao vetor de pressão ($SNR_{dB} = 50dB$).	71
5.4.7 Curva de convergência para simulações com adição de ruído branco gaussiano ao vetor de pressão ($SNR_{dB} = 100dB$).	72
5.4.8 Curva de convergência para simulações com adição de ruído branco gaussiano ao vetor de deslocamento ($SNR_{dB} = 30dB$).	72
5.4.9 Curva de convergência para simulações com adição de ruído branco gaussiano ao vetor de deslocamento ($SNR_{dB} = 50dB$).	73
5.4.10 Curva de convergência para simulações com adição de ruído branco gaussiano ao vetor de deslocamento ($SNR_{dB} = 100dB$).	73
6.1.1 Imagem de ultrassom e um conjunto χ_1 de espalhadores.	75
6.1.2 Malha de elementos finitos deformada e deslocamento dos espalhadores.	75
6.1.3 Elemento finito deformado e distâncias entre o espalhador e os vértices do elemento.	76
6.1.4 phantom matemático $r = 5.5mm$: (a) Imagem pré-deformação; (b) Imagem pós-deformação.	80
7.2.1 Phantoms transcutâneos com lesão, sendo a figura à esquerda a imagem pré-deformação e a imagem a direita pós-deformação.	83
7.2.2 Simulações com Lucas-Kanade variando o tamanho da janela de amostragem: (a) Janela $20 \times 20pixels$; (b) Janela $40 \times 40pixels$; (c) Janela $60 \times 60pixels$;	84
7.4.1 phantom com malha de elementos finitos e deslocamentos calculados por elementos finitos (marrom) e fluxo óptico (vermelho) utilizando o algoritmo de Lucas-Kanade.	87

7.4.2 Detalhe para comparação qualitativa do algoritmo de fluxo óptico por Lucas-Kanade e elementos finitos.	88
7.4.3 phantom com malha de elementos finitos e deslocamentos calculados por elementos finitos (marrom) e fluxo óptico (vermelho) utilizando o algoritmo de Brox.	89
7.4.4 Detalhe para comparação qualitativa do algoritmo de fluxo óptico por Brox e elementos finitos.	90
8.1.1 Erro ϵ (%) para o background em função da pressão aplicada.	93
8.1.2 Erro ϵ (%) para a lesão em função da pressão aplicada.	94
8.2.1 Erro ϵ (%) para o background em função da pressão aplicada.	96
8.2.2 Erro ϵ (%) para a lesão em função da pressão aplicada.	97
8.2.3 Convergência do método em função da quantidade de elementos de contorno para a região da lesão.	97
8.2.4 Convergência do método em função da quantidade de elementos de finitos para a região da lesão.	98

Lista de Tabelas

5.1	Comparação entre MVC e diferenças finitas.	66
5.2	Parâmetros utilizados para simulação do phantom com duas lesões.	68
7.1	Medidas de erro para o phantom 1.	88
7.2	Medidas de erro para o phantom 1.	90
8.1	Propriedades físicas e materiais para o phantom 1.	91
8.2	Resultados para os módulos de elasticidade estimados utilizando MEC e Lucas-Kanade para o phantom 1.	92
8.3	Resultados para os módulos de elasticidade estimados utilizando MEC e Brox para o phantom 1.	92
8.4	Resultados para os módulos de elasticidade estimados utilizando MEF e Lucas-Kanade para o phantom 1.	92
8.5	Resultados para os módulos de elasticidade estimados utilizando MEF e Brox para o phantom 1.	93
8.6	Erro médio e desvio padrão para cada associação de métodos para o phantom 1.	93
8.7	Material and geometric properties for phantom 2.	94
8.8	Resultados para os módulos de elasticidade estimados utilizando MEC e Lucas-Kanade para o phantom2.	95
8.9	Resultados para os módulos de elasticidade estimados utilizando MEC e Brox para o phantom2.	95
8.10	Resultados para os módulos de elasticidade estimados utilizando MEF e Lucas-Kanade para o phantom2.	95
8.11	Resultados para os módulos de elasticidade estimados utilizando MEF e Brox para o phantom2.	96
8.12	Erro médio e desvio padrão para cada associação de métodos para o phantom2.	96
8.13	Quantidade de iterações para a convergência e o tempo máximo para cada combinação de método.	98

Notações e Definições

- MEC - Método dos Elementos de Contorno
- MEF - Método dos Elementos Finitos
- MVC - Método de Variável Complexa
- EIC - Equação Integral de Contorno
- OF - Fluxo Óptico
- CAD - Computer Aided Detection
- $h_i(\eta)$ - Função de interpolação polinomial para elementos de contorno
- $[H]$ - Matriz de influência para forças de superfície
- $[G]$ - Matriz de influência para deslocamentos
- ω_i - Função de interpolação polinomial para elementos finitos
- $[K]$ - Matriz de rigidez de elementos finitos
- $[K]^e$ - Matriz de rigidez elementar de elementos finitos
- η - Variável paramétrica
- Ω - Domínio do problema
- $\partial\Omega$ - Contorno do domínio Ω
- Γ - Contorno do domínio, $\Gamma = \partial\Omega$
- N_e - Número de elementos de contorno
- N_R - Quantidade de sub-regiões no problema
- N_E - Quantidade de materiais com módulos de elasticidade distintos
- σ - Tensão $[Pa] = [N/m]$

- ε - Deformação $[m/m]$
- E_i - Módulo de Young do i-ésimo material
- ν - Coeficiente de Poisson
- C_{ijkl} - Tensor de propriedades materiais
- $[C]$ - Matriz de propriedades materiais Como uma das limitações do MEC e MEF
- ω_i - i-ésima função de ponderação
- u_{ij}^* - Função de Green para deslocamentos
- t_{ij}^* - Função de Green para forças de superfície
- $\vec{\xi}$ - Ponto fonte
- j - Unidade complexa $j = \sqrt{-1}$
- $\bar{\sigma}$ - Desvio padrão

Sumário

1	Introdução	14
1.1	Objetivos	16
1.2	Estrutura da Tese	17
2	Revisão Bibliográfica	19
3	Elementos da Mecânica dos Sólidos Deformáveis	23
3.1	Introdução	23
3.2	Relações de Tensão, Deformação e Lei de Hooke para Elasticidade Isotrópica Linear	23
4	O Método dos Elementos Finitos e o Método dos Elementos de Contorno	33
4.1	Introdução	33
4.2	O Método dos Elementos Finitos	35
4.2.1	Formulação das Equações de Elementos Finitos	35
4.3	O Método dos Elementos de Contorno	41
4.3.1	Formulação da Equação Integral de Contorno para Elastostática	41
4.3.2	Discretização em Elementos de Contorno	45
4.3.3	Método de Sub-regiões	50
4.4	Malhas para Elementos de Contorno e Elementos Finitos	53
5	Análise Inversa	62
5.1	Introdução	62
5.2	Formulação do Problema	63
5.3	Método de Variáveis Complexas	65
5.4	Validação da Análise Inversa	67
6	Criação de Phantoms Matemáticos	74
6.1	Introdução	74
6.1.1	Elementos Finitos e Morfismo	75
6.1.2	O <i>software</i> USSim	76

7	Fluxo Óptico para Rastreamento de Ruído Speckle	81
7.1	Introdução	81
7.2	Fluxo Óptico por Lucas-Kanade	82
7.3	Fluxo Óptico por Brox	85
7.4	Validação dos Métodos para rastreamento de ruído speckle	86
7.4.1	Análise de Erro para o Algoritmo de Lucas-Kanade	87
7.4.2	Análise de Erro para o Algoritmo de Brox	89
8	Resultados	91
8.1	Phantom 1	91
8.2	Phantom 2	94
9	Discussão dos Resultados, Conclusões e Trabalhos Futuros	99
9.1	Discussão dos Resultados e Conclusão	99
9.2	Sugestões para Trabalhos Futuros	100
	Referências Bibliográficas	100

Resumo

Este trabalho apresenta uma nova metodologia para elastografia virtual em imagens simuladas de ultrassom utilizando métodos numéricos e métodos de visão computacional. O objetivo é estimar o módulo de elasticidade de diferentes tecidos tendo como entrada duas imagens da mesma seção transversal obtidas em instantes de tempo e pressões aplicadas diferentes. Esta metodologia consiste em calcular um campo de deslocamento das imagens com um método de fluxo óptico e aplicar um método iterativo para estimar os módulos de elasticidade (análise inversa) utilizando métodos numéricos. Para o cálculo dos deslocamentos, duas formulações são utilizadas para fluxo óptico: Lucas-Kanade e Brox. A análise inversa é realizada utilizando duas técnicas numéricas distintas: o Método dos Elementos Finitos (MEF) e o Método dos Elementos de Contorno (MEC), sendo ambos implementados em Unidades de Processamento Gráfico de uso geral, GpGPUs (*“General Purpose Graphics Units”*). Considerando uma quantidade qualquer de materiais a serem determinados, para a implementação do Método dos Elementos de Contorno é empregada a técnica de sub-regiões para acoplar as matrizes de diferentes estruturas identificadas na imagem. O processo de otimização utilizado para determinar as constantes elásticas é realizado de forma semi-analítica utilizando cálculo por variáveis complexas. A metodologia é testada em três etapas distintas, com simulações sem ruído, simulações com adição de ruído branco gaussiano e phantoms matemáticos utilizando rastreamento de ruído speckle. Os resultados das simulações apontam o uso do MEF como mais preciso, porém computacionalmente mais caro, enquanto o MEC apresenta erros toleráveis e maior velocidade no tempo de processamento.

Palavras-chave: Elastografia virtual; Elementos de Contorno; Ultrassom; Métodos Numéricos; Fluxo Óptico; Otimização Material

Abstract

This thesis presents a new methodology for computational elastography applied to simulated ultrasound images, using numerical methods and computer vision methods. The aim is to estimate the elastic moduli of different tissues using two different images of the same cross section acquired in different times and pressure conditions. The proposed methodology consists in evaluate the displacement field using optical flow techniques and then apply an inverse analysis using a numerical method. In order to evaluate the displacement field, two distinct formulations for optical flow are used: Lucas-Kanade and Brox. For the inverse analysis problem, the Finite Element Method and the Boundary Element Method are used, both implemented in general purpose graphic units, GpGPUs. Considering a number of materials that may be present in the images, the multiresgions boundary element method is used in order to couple different matrices for different materials. The optimization process is evaluated using complex variable method. The methodology is validated in three different steps: noiseless simulations; additive white gaussian noise simulations; and ultrasound mathematical phantom with speckle tracking. The results show that the Finite Element Method presents more accurate estimatives but a high computational cost, while the Boundary Element Method presents tolerable errors but a better processing time.

Keywords: Virtual Elastography; Boundary Elements; Ultrasound; Numerical Methods; Optical Flow; Material Optimization;

Capítulo 1

Introdução

Nas últimas décadas, ocorreu um aumento significativo da expectativa de vida graças ao desenvolvimento de novas tecnologias em diversas áreas, principalmente em Ciências da Saúde. Segundo a Organização Mundial de Saúde¹ (OMS - “*World Health Organization*”), apenas no Brasil, a expectativa de vida média em 1990 era de até 66 anos, já em 2014 era de 74 anos. Grande parte desta evolução, se deve ao investimento em novas tecnologias de prevenção de doenças. Entre as técnicas preventivas, uma das principais que podem ser citadas é a aquisição de imagens e entre as mais utilizadas, o ultrassom.

Técnicas de ultrassom são utilizadas no diagnóstico de uma grande variedade de doenças, desde detecção de tumores em mama ou tireóide à aplicações intravasculares (“*Intravascular Ultrasound*” - IVUS), para verificação de lesões e possíveis obstruções vasculares.

Tomando como exemplo exames de mamografia, a detecção de nódulos é realizada de forma subjetiva pelo técnico responsável, que verifica a uniformidade da imagem em relação a padrões previamente estabelecidos, isto é, busca por possíveis inclusões presentes na imagem. Assim, o laudo depende, entre outros fatores, da experiência do operador e torna o exame altamente operador-dependente. Julia et al. [36] aponta que cerca de 10 a 30% das lesões não são consideradas, devido a limitações dos técnicos responsáveis e no mesmo estudo, aponta que o uso de CAD (“*Computer Aided Detection*” - Detecção por Auxílio de Computador) aumenta em 21% a sensibilidade na detecção de um câncer de mama pelo especialista.

Neste sentido, entre os vários métodos para auxiliar o especialista no processo de avaliação de imagens, está a Elastografia. Como descreve Doyley and Parker [23], a elastografia consiste em (i) aplicar uma perturbação sobre o tecido, usando uma fonte de excitação mecânica quasi-estática, harmônica ou transiente. Tais fontes podem ser de origens externas ou internas, por exemplo, a aplicação de pressão sobre a superfície imageada pelo transdutor de ultrassom ou por algum fenômeno biológico, como respiração ou a própria pressão arterial; (ii) medir a resposta em termos de deslocamentos, deformações ou fase e amplitude das vibrações; (iii) estimar as

¹http://gamapserver.who.int/gho/interactive_charts/mbd/life_expectancy/atlas.html - Acessado em 14/08/2014.

propriedades mecânicas dos tecidos imageados. Embora os trabalhos pioneiros em Elastografia tenham sido desenvolvidos para aplicações em imagens de ultrassom, estudos equivalentes podem ser encontrados em várias modalidades de imagens médicas, como tomografia por coerência óptica (*“optical coherence tomography”* - OCT) e ressonância magnética. Deve-se ressaltar que, salvo aplicações em imagens intravasculares, onde é necessário a utilização de catéteres para aquisição das imagens dos vasos sanguíneos, métodos de Elastografia garantem uma abordagem não invasiva, preservando a integridade do tecido imageado e minimizando o desconforto do paciente.

A análise dos tecidos imageados é baseada na premissa que tecidos saudáveis possuem comportamento mecânico diferente de tecidos lesionados, ou que apresentem algum tipo de anormalidade. Como exemplo, Krouskop and Wheeler [43] mostra que tumores benignos apresentam comportamentos próximos da linearidade material, enquanto tumores malignos apresentam não-linearidade material, ou seja, a relação entre tensão e deformação possui variação diferente de um modelo linear.

A literatura apresenta modelos complexos em relação ao comportamento elástico do tecido, considerando características anisotrópicas [30, 68], viscoelásticas [69, 63] e até mesmo poroelásticas [61, 60]. No entanto, a modelagem do problema físico é bem simplificada, segundo Doyley [22], alguns modelos assumem o comportamento mecânico do phantom físico como uma associação de molas em paralelo, limitando a aplicação das técnicas desenvolvidas em modelos mais realistas do ponto de vista clínico.

Como será discutido no próximo capítulo, alguns trabalhos possuem modelos mais simplificados em relação ao comportamento elástico dos tecidos, considerando-os como isotrópicos, porém com uma modelagem mais realística do problema físico. Estes trabalhos consideram mapas de deformação como parâmetros para identificar os tecidos lesionados e dois pontos importantes devem ser observados em relação às metodologias apresentadas: (i) em sua maioria, são utilizados sinais de rádio frequência (RF) para estimar os campos de deslocamento e os mapas de deformações dos tecidos imageados. Embora sinais de RF apresentem maior quantidade de informação, o que permite cálculos mais precisos dos campos de deslocamentos, o uso deste tipo de informação é um limitante para a aplicação da técnica, pois esta informação não é disponibilizada pelo fabricante do equipamento e, em muitos casos, é tratada como proprietária e confidencial. Assim, é desejável que o estudo seja baseado em imagens como dados de entrada para cálculo de deslocamentos e deformações, constituindo assim um método independente do fabricante do equipamento de imagens médicas; (ii) as informações de deformação, sem o conhecimento da distribuição de pressão aplicada, não possibilita uma estimativa precisa das características elásticas dos tecidos imageados, levando a uma análise subjetiva dos resultados da Elastografia. Como resalta Guo et al. [32], nenhuma das técnicas que utilizam apenas as informações de deslocamentos são capazes de estimar diretamente as propriedades materiais; ao invés disto, permitem que o próprio radiologista analise o campo de deformação, tornando o resultado do estudo operador-dependente.

Neste contexto, esta tese apresenta uma nova metodologia, que integra métodos de visão computacional para cálculo de deslocamentos em imagens de ultrassom e métodos de análise estrutural e

otimização material para estimar as propriedades elásticas e fornecer parâmetros consistentes para caracterização de tecidos baseados em seus módulos de elasticidade. São empregados métodos de Fluxo Óptico (FO) para o cálculo de deslocamentos relativos entre imagens de ultrassom, obtidas em instantes de tempo e pressões diferentes. O método de Brox [12] e o método de Lucas-Kanade [50] são utilizados nesta tese: enquanto o método de Brox mostra-se eficiente para cálculo de grandes deslocamentos, o método de Lucas-Kanade possui uma formulação mais simples e melhor desempenho computacional. Em seguida, para análise inversa são empregados métodos numéricos para elastostática baseados em formulações integrais, que possibilitam flexibilidade para descrição da geometria dos tecidos. Dois métodos com características distintas são comparados: o Método dos Elementos Finitos (MEF) [Zienkiewicz and Emeritus] e o Método dos Elementos de Contorno [7] (MEC). Para que estas análises sejam realizadas no menor tempo possível, os métodos de análise estrutural são implementados em Unidades de Processamento Gráfico (*“Graphical Process Units”* - GPUs).

1.1 Objetivos

O objetivo principal desta tese é desenvolver um método para realizar elastografia virtual utilizando uma sequência de imagens de ultrassom, algoritmos de fluxo óptico, método dos elementos de contorno e técnicas de análise inversa para otimização material. Este método é então validado utilizando simulações numéricas e por fim, aplicada a phantoms matemáticos.

Para que o objetivo desta tese fosse atingido, foram necessárias a implementação das seguintes etapas:

- Desenvolvimento e algoritmo de um programa de elementos de contorno para imagens 2D que seja capaz de tratar múltiplos materiais;
- Desenvolvimento e validação de um algoritmo de elementos finitos para imagens 2D;
- Desenvolvimento e validação dos métodos numéricos citados acima em Unidades de Processamento Gráfico (GPUs);
- Desenvolvimento e validação de um algoritmo para fluxo óptico capaz de realizar o rastreamento de ruído *speckle* em imagens de ultrassom, a saber, Lucas-Kanade e Brox;
- Desenvolvimento e validação de um método de análise inversa utilizando MEC e MEF;
- Desenvolvimento dos módulos de fluxo óptico e análise inversa;
- Validação da metodologia proposta utilizando phantoms matemáticos.

A Figura 1.1.1 mostra o diagrama da metodologia. No diagrama, t_0 e t_1 representam os instantes pré e pós-deformação e u e v os campos de deslocamentos horizontal e vertical respectivamente,

pressão aplicada é o campo de pressões aplicado para deformar o tecido imageado, E_0 é uma estimativa inicial para o módulo de elasticidade e $E_{estimado}$ é o módulo de elasticidade estimado.

Figura 1.1.1: Diagrama para elastografia virtual em imagens de ultrassom, rastreamento de ruído speckle e análise inversa.

1.2 Estrutura da Tese

Esta tese está estruturada como segue:

Capítulo 3 - O objetivo deste capítulo é fornecer os conceitos básicos para o entendimento da mecânica de sólidos isotrópicos em regime elástico linear. Apresenta-se os conceitos de tensor de tensões e suas relações com solicitações externas, ou seja, dado um conjunto de forças externas aplicadas sobre um corpo, deduz as condições de equilíbrio interno em função de tensões definidas em elementos de volume ou área infinitesimais; também apresenta os conceitos de deslocamentos e deformações, estabelecendo as relações diferenciais de compatibilidade cinemática e definindo linearidade geométrica; por fim, une os conceitos apresentados anteriormente por uma relação linear, a Lei de Hooke para materiais isotrópicos, definindo módulo de Elasticidade, ou módulo de Young e coeficiente de Poisson.

Capítulo 4 - Este capítulo visa fornecer uma visão geral da formulação do Método dos Elementos Finitos e o Método dos Elementos de Contorno para problemas bidimensionais. Apresenta o desenvolvimento das Equações Integrais de Contorno (EIC), as funções de Green para elasticidade linear isotrópica, as chamadas soluções fundamentais para forças de superfície e deslocamentos. Além disto, como este trabalho envolve a identificação de vários materiais dentro de um mesmo meio, é necessário o uso da técnica de sub-regiões, também abordada neste capítulo. É discutida a geração de malhas para os métodos de elementos de contorno e a utilização do *software* Gmsh para a criação de malhas de elementos finitos. Também é abordada a tecnologia de placas gráficas e sua aplicação a problemas de modelagem computacional. São discutidos os aspectos gerais, tais como arquitetura básica, limitações e algumas comparações do desempenho de GPUs em relação

a CPUs, além de tratar da implementação dos métodos numéricos utilizados nesta tese. Por fim, são apresentados os resultados da validação do programa implementado, comparando os resultados com modelos simples, encontrados na literatura e modelos analíticos.

Capítulo 5 - Apresenta a técnica de derivadas por variáveis complexas e o problema de otimização relacionado à análise inversa, necessária para estimar os valores de módulo de elasticidade dos materiais imageados. Além da formulação, apresenta um exemplo e comparação com diferenças finitas.

Capítulo 6 - Apresenta-se os phantoms matemáticos utilizados para validação da metodologia proposta nesta tese. São utilizados phantoms transcutâneos com lesão de diâmetro variável localizadas no centro da imagem. Também são considerados phantoms com diferentes módulos de elasticidade;

Capítulo 7 - Discute dois algoritmos para fluxo óptico, de Brox e de Lucas-Kanade. Apresenta as formulações e comparações entre os dois métodos baseados em resultados de imagens de ultrassom com simulações por elementos finitos;

Capítulo 8 - Apresenta os resultados da integração dos módulos de fluxo óptico com os métodos numéricos. O método é validado em três etapas: simulações sem ruído; simulações com ruído gaussiano aditivo; e a integração de fluxo óptico com elementos finitos e elementos de contorno.

Capítulo 9 - Conclusões da tese e sugestões para trabalhos futuros.

Capítulo 2

Revisão Bibliográfica

As técnicas de elastografia podem ser divididas em duas categorias: (i) as que estimam o valor do módulo de elasticidade e (ii) aquelas que fornecem um mapa de deformações. Em relação ao item (i), a técnica numérica utilizada para a análise inversa é o MEF. Como será visto mais adiante durante a revisão da literatura, os trabalhos que propoem estimar os módulos de elasticidade utilizando MEF, fazem uso, em sua grande maioria, de simulações numéricas adicionadas de ruído para a validação da metodologia. Já as técnicas do item (ii) podem ser classificadas entre aquelas que utilizam correlação de imagens (*“block matching”*) e aquelas que utilizam Fluxo Óptico.

Os estudos pioneiros utilizando técnicas de Elastografia datam do final da década de 80 ao início da década de 90, com os trabalhos de Ophir et al. [57] e O'Donnell et al. [56]. Graças ao sucesso na caracterização de tecidos em imagens de ultrassom, estes trabalhos serviram como base para outros trabalhos sobre Elastografia utilizando imagens de outras modalidades, como ressonância magnética (RM) e mais tarde, em tomografia por coerência óptica (*“Optical Coherence Tomography”* - OCT), onde podem ser citados Muthupillai R, Lomas DJ, Rossman PJ, Greenleaf JF, Manduca A [55], Bishop et al. [10], Weaver JB, Van Houten EE, Miga MI, Kennedy FE [70] para RM e Khalil et al. [40] e Kirkpatrick SJ, Wang RK [41] para OCT.

Raghavan et al. [64] apresenta um estudo utilizando diferenças finitas para simulação do comportamento mecânico e posterior análise inversa em elastografia. Utiliza um modelo isotrópico e linear. No entanto, o uso de diferenças finitas limita sua aplicação em modelos com geometrias mais complexas.

Kallel and Ophir [37] propoem uma técnica chamada Estimador de Deformações por Mínimos Quadrados (*“Least-Square Strain Estimator”* - LSSE). O trabalho consiste, basicamente, em assumir que as deformações possuem natureza afim, isto é, são descritas por equações de reta e o trabalho consiste em aplicar a técnica de mínimos quadrados para determinar os coeficientes destas equações. Utilizando simulações, os autores mostram que sua metodologia minimiza a influência do ruído ao calcular os valores de deformação, melhorando a relação sinal-ruído (*“Signal to Noise Ratio”* - SNR) para o elastograma (distribuição de deformações do tecido).

Romano et al. [65] apresentam um modelo de elementos finitos com considerações de amortecimento. Neste estudo, os autores consideram o coeficiente de Poisson variável e aplicam sua metodologia em simulações numéricas.

Maurice et al. [53] apresentam um estudo utilizando imagens de ultrassom intravascular (*“Intravascular Ultrasound”* - IVUS), e utilizam um modelo para estimar o movimento do ruído speckle, chamado de Modelo Lagrangeano para Estimação de Ruído Speckle (*“Lagrangian Speckle Model Estimator”* - LSME). Neste artigo, os autores integram o LSME com algoritmos de minimização e fluxo óptico para cálculo das matrizes hessianas. Seus resultados são apresentados em forma de diagramas com a distribuição do mapa de deformações sobre os vasos analisados.

Baldewsing et al. [3] apontam o uso de elementos finitos como uma possível ferramenta para estimar propriedades elásticas de tecidos imageados por ultrassom, em particular, os autores comparam simulações por MEF com imagens de IVUS. Os autores também apresentam outro resultado importante quanto ao comportamento elástico dos materiais imageados: eles afirmam ser possível modelar tecidos biológicos como materiais isotrópicos e desta forma, é necessário o cálculo apenas dos módulos de elasticidade de cada tecido, considerando o coeficiente de Poisson como valor constante e único para qualquer tecido imageado. Posteriormente, Baldewsing et al. [4] apresentam uma metodologia para estimar os valores de módulo de elasticidade em imagens de IVUS utilizando elementos finitos parametrizados. Neste trabalho, os autores buscam, através de técnicas de otimização morfológica e atualização dos valores de módulos de elasticidade das regiões de interesse (*“Regions of Interest”* - ROI) selecionadas, casar os contornos das malhas de elementos finitos com a geometria da imagem adquirida de IVUS. Esta abordagem utiliza apenas a imagem da seção transversal submetida à pressão como referência.

Em um trabalho mais recente, DiBattista and Noble [20] apresentam um algoritmo para elastografia utilizando *“block matching”* e, em um primeiro momento, utilizam sinais de RF de ultrassom para, posteriormente, aplicar o algoritmo em imagens simuladas de ultrassom. O trabalho consiste em determinar um índice de semelhança ótimo, utilizando coeficiente de correlação cruzada normalizado. Neste trabalho, a imagem onde é assumida uma deformação é percorrida por uma janela de tamanho fixo e uma função custo é minimizada para os valores de deslocamento e deformação. Toda a imagem é percorrida antes que novos valores estimados para os parâmetros desejados sejam testados. É importante observar que esta metodologia resulta em um mapa de deformações e não calcula os valores de módulo de elasticidade para os tecidos.

Zakaria et al. [71] utilizam um método de FO para obter os valores de deslocamento e deformação em imagens de ultrassom em carótida de ratos. Neste trabalho, os autores utilizam FO por Lucas-Kanade para o cálculo do campo de deslocamentos, em seguida, utilizam um *“image warp”*, ou seja, com os valores de deslocamentos conhecidos, a imagem pré-deformação é compensada pelos deslocamentos nos dois eixos coordenados e em seguida, os valores de deformações são calculados. As equações pertinentes ao método de Lucas-Kanade são apresentadas no Capítulo 7 e a Figura 2.0.1 ilustra a técnica, sendo I_1 e I_2 as imagens pré e pós-deformação, (x_1, x_2) as coorde-

nadas cartesianas na imagem, (u_1, u_2) o campo de deslocamento nas direções 1 e 2 respectivamente e $[\varepsilon]$ o tensor de deformações, tratado no Capítulo 3. Apostolakis et al. [2] fazem um estudo semelhante, utilizando Lucas-Kanade ponderado, também conhecido como fluxo óptico por mínimos quadrados ponderados (*“Weighted Least Squares Optical Flow”* - WLSOF), no qual a equação de fluxo óptico é ponderada por uma função de distribuição gaussiana. Esta abordagem equivale a aplicar um filtro gaussiano nas imagens pré e pós deformação e em seguida aplicar o método de Lucas-Kanade convencional.

Figura 2.0.1: Diagrama para cálculo de campo de deslocamento e deformações usando Lucas-Kanade, segundo Zakaria et al. [71].

Bertrand [8] apresenta uma metodologia para estimar o módulo de elasticidade com o MEF para análise estrutural, que inclui condições de contorno e de deslocamentos. O método de Newton-Rapson é usado para análise inversa. O autor utiliza simulações por elementos finitos e adição de ruído para validar sua metodologia.

Zhu et al. [73] utilizam MEF para estimar o módulo de elasticidade, em phantoms matemáticos. Consideram cada elemento como uma região de interesse, e assim, determinam o módulo de elasticidade para cada elemento de forma independente. Para isto, também apresentam uma formulação para estimar os valores de módulo de elasticidade fazendo uso da natureza esparsa das matrizes oriundas do MEF. Os autores consideram duas situações: uma sem ruído, sendo os dados de entrada os valores obtidos por simulações de elementos finitos, e uma segunda hipótese, em que consideram vetores de carregamento e de deslocamento adicionados de ruído gaussiano branco. Em Guo et al. [32] os autores apresentam um trabalho similar, considerando cada elemento como uma região de interesse, sendo que o módulo de elasticidade é estimado pela minimização de uma função custo, resultante da soma da norma L2 do resíduo (diferença entre o valores estimados por

MEF e os valores de referência, obtidos por simulação por simulações numéricas) em todos os nós do domínio.

Khalil et al. [40] utilizam um phantom transcutâneo de OCT e MEF como ferramenta para análise inversa. Os deslocamentos entre imagens são obtidos por minimização de um funcional de energia, descrito em Chan et al. [19]. Graças à resolução espacial da técnica de OCT, os estudos apresentam bons resultados na estimação dos módulos de elasticidade.

Liu et al. [47] utilizam uma técnica de *“block matching”* associada ao algoritmo de otimização de Powell (*“Numerical Optimization Method with Powell Algorithm”* - NOMPA), que é um algoritmo de otimização sem uso de derivadas [62]. Os autores utilizam sinais de RF e determinam diretamente os valores de deslocamento e deformação. A função objetivo a ser minimizada é a norma L2 da diferença entre as imagens pré e pós-deformação.

Recentemente, Pan et al. [58] apresentam uma metodologia utilizando FO e sinais de RF simulados. A técnica consiste em aplicar correlação cruzada e fluxo óptico com compensação dos deslocamentos e deformações, chamdo de Fluxo Óptico em dois passos (*“Two Step Optical Flow”* - TSOF) e novamente aplicar TSOF.

Como pode ser observado, os métodos disponíveis para elastografia virtual com aplicações em problemas práticos, se resumem ao cálculo do campo de deformações utilizando métodos de visão computacional. Por outro lado, métodos que quantificam módulos de elasticidade não são validados em imagens simuladas ou reais. Desta forma, a contribuição desta tese está em integrar métodos de fluxo óptico e modelos numéricos e a utilização do MEC como ferramenta de análise estrutural ao invés do MEF.

Capítulo 3

Elementos da Mecânica dos Sólidos Deformáveis

3.1 Introdução

Nos próximos capítulos serão apresentadas as formulações para o MEC e MEF para elasticidade linear e isotrópica. Antes, no entanto, é interessante discutir algumas Equações fundamentais da Mecânica dos Sólidos para mais adiante no texto, o desenvolvimento das Equações Integrais de Contorno (EIC) possa ficar mais claro. O objetivo é apresentar três conceitos fundamentais para o estudo de corpos deformáveis:

1. Tensor de tensões, equilíbrio de forças e forças de superfície;
2. Tensor de deformações e compatibilidade cinemática;
3. Lei de Hooke para materiais isotrópicos e lineares.

O primeiro item considera o equilíbrio interno do corpo em resposta a aplicação de solicitações externas; já o segundo item, relaciona os deslocamentos e deformações de um corpo elástico linear considerando pequenas deformações. Nesta tese, são assumidos deslocamentos de até 2% da maior dimensão do corpo; o terceiro item relaciona tensões e deformações através das propriedades materiais do sólido.

3.2 Relações de Tensão, Deformação e Lei de Hooke para Elasticidade Isotrópica Linear

Considerando um corpo composto de um material linear e em repouso, sujeito a restrições de deslocamento e submetido a solicitações externas, isto é, aplicação de forças de superfícies como

mostra a Figura 3.2.1, é possível descrever o equilíbrio em termos de somatório de forças que atuam sobre este corpo (Equação 3.2.1).

Figura 3.2.1: Corpo sujeito à restrições de deslocamento e solicitações externas.

$$\sum_i \vec{F}_i = 0 \quad (3.2.1)$$

Também é possível descrever o equilíbrio em partes do corpo por meio de forças internas resultantes da aplicação das forças de superfície. Considerando um plano Π , as forças internas se dividem entre forças normais, $\vec{F}_n$, e de cisalhamento, $\vec{F}_c$, como mostra a Figura 3.2.2.

Figura 3.2.2: Corte em um plano Π e as forças internas resultantes.

Estas forças, por sua vez, descrevem um estado de tensão $[\sigma]$ em um ponto do corpo. Cada componente do tensor $[\sigma]$ é definido pela Equação 3.2.2, sendo o sub-índice i a direção da força e j o eixo normal ao plano que contém ΔA_j (Figura 3.2.3).

$$\sigma_{ij} \triangleq \lim_{\Delta A_j \rightarrow 0} \frac{F_i}{\Delta A_j} \quad (3.2.2)$$

Figura 3.2.3: Definição de tensão normal e de cisalhamento.

Para problemas de elastostática linear, o campo de tensões para um elemento infinitesimal em coordenadas Cartesianas é ilustrado na Figura 3.2.4 e definido pela Equação 3.2.3.

Figura 3.2.4: Campo de tensões em um elemento infinitesimal.

$$[\sigma] = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{bmatrix} \quad (3.2.3)$$

sendo que as componentes da diagonal principal representam as tensões normais, enquanto as outras componentes são as tensões de cisalhamento. Cabe notar que o tensor é simétrico, $\sigma_{ij} = \sigma_{ji}$.

Com a ajuda da Equação 3.2.3, também é possível calcular as forças de superfícies normais em qualquer plano no sólido utilizando o Tensor de Cauchy, Equação 3.2.4, como ilustrado na Figura 3.2.5.

$$\{t\} = [\sigma] \{n\} \quad (3.2.4)$$

Figura 3.2.5: Vetor de força de superfície.

O equilíbrio de forças para o elemento infinitesimal é descrito por um conjunto de três Equações diferenciais (Equação 3.2.5),

$$\frac{\partial \sigma_{i1}}{\partial x_1} + \frac{\partial \sigma_{i2}}{\partial x_2} + \frac{\partial \sigma_{i3}}{\partial x_3} + b_i = 0 \quad i = 1, 2, 3 \quad (3.2.5)$$

onde b_i é a i -ésima componente do vetor de forças de corpo, isto é, forças presentes no domínio do problema, como por exemplo, peso próprio.

Outro aspecto importante para o estudo da Mecânica dos Sólidos, é o estudo das deformações

causadas em corpos submetidos a solicitações externas. Considerando um elemento pré e pós-deformação, mostrado na Figura 3.2.6,

Figura 3.2.6: Modelo de uma fibra deformada.

o deslocamento u_Q sofrido pelo ponto $Q(x_1, x_2)$ é obtido pela expansão em série de Taylor,

$$u_Q = u_P + \frac{du_P}{dx_1} \Delta x_1 + \frac{1}{2} \frac{d^2 u_P}{dx_1^2} \Delta x_1^2 + \cdots + \frac{1}{n!} \frac{d^n u_P}{dx_1^n} \Delta x_1^n \quad (3.2.6)$$

Adotando a premissa que $\Delta x^2 \ll \Delta x$, e assim ignorando os termos de ordem superior a $\mathcal{O}(\Delta x^2)$, o deslocamento u_Q pode ser escrito como

$$u_Q = u_P + \frac{du_P}{dx_1} \Delta x_1 \quad (3.2.7)$$

Assim, a variação de comprimento do segmento $\bar{P}Q$ é dado pela Equação 3.2.8

$$\frac{\Delta u}{\Delta x_1} = \frac{du_P}{dx_1} \quad (3.2.8)$$

no limite $\Delta x \rightarrow 0$, a deformação é definida pela Equação 3.2.9

$$\varepsilon_1 = \frac{\partial u}{\partial x_1} \quad (3.2.9)$$

O mesmo raciocínio pode ser aplicado aos três eixos coordenados, e também é possível definir deformações angulares em função dos deslocamentos, como mostra a Figura 3.2.7

Figura 3.2.7: Variação angular.

Assim, para problemas tridimensionais, há três componentes de deformações ao longo dos eixos coordenados e seis variações angulares. Cada uma destas componetes faz parte do que se chama de tensor de deformações $[\varepsilon]$, sendo cada uma delas definida de forma geral pela Equação 3.2.10, onde as componentes da diagonal representam as deformações axiais enquanto as outra componentes representam as variações angulares. Assim como o tensor $[\sigma]$, $\varepsilon_{ij} = \varepsilon_{ji}$.

$$\varepsilon_{ij} = \frac{1}{2} \left(\frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j} \right) \quad i, j = 1, 2, 3 \quad (3.2.10)$$

Por fim, a Lei de Hooke relaciona o tensor de tensões e o tensor de deformações, Equação 3.2.11.

$$\sigma_{ij} = \sum_{k=1}^3 \sum_{l=1}^3 C_{ijkl} \varepsilon_{kl} \quad (3.2.11)$$

A Equação 3.2.11 relaciona dois tensores de segunda ordem através de um tensor de propriedades materiais de quarta ordem, C_{ijkl} . Este tensor também apresenta simetria, como mostrado no conjunto de Equações 3.2.12.

$$\begin{aligned} C_{ijkl} &= C_{ijlk} \\ C_{jilk} &= C_{ijkl} \\ C_{ikjl} &= C_{klij} \end{aligned} \quad (3.2.12)$$

Assim, de um total de 81 componentes, e usando também a simetria dos tensores $[\sigma]$ e $[\varepsilon]$, a quantidade de componentes independentes é reduzida para 21 [26]. Para um problema isotrópico, é possível reescrever a Equação 3.2.11 de forma matricial, como na Equação 3.2.13

$$\{\sigma\} = [C] \{\varepsilon\} \quad (3.2.13)$$

Sendo

$$\{\sigma\} = \begin{Bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{12} \\ \sigma_{13} \\ \sigma_{23} \end{Bmatrix} \quad (3.2.14)$$

$$\{\varepsilon\} = \begin{Bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ \varepsilon_{12} \\ \varepsilon_{13} \\ \varepsilon_{23} \end{Bmatrix} \quad (3.2.15)$$

$$[C] = \begin{bmatrix} C_{1111} & C_{1122} & C_{1133} & C_{1112} & C_{1113} & C_{1123} \\ & C_{2222} & C_{2233} & C_{2212} & C_{2213} & C_{2223} \\ & & C_{3333} & C_{3312} & C_{3313} & C_{3323} \\ & & & C_{1212} & C_{1213} & C_{1223} \\ & sym & & & C_{1313} & C_{1323} \\ & & & & & C_{2323} \end{bmatrix} \quad (3.2.16)$$

Para problemas isotrópicos, a matriz de propriedades materiais da Equação 3.2.16 é estabelecida como

$$[C] = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1-\nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1-\nu & 0 & 0 & 0 \\ 0 & 0 & 0 & (1-2\nu)/2 & 0 & 0 \\ 0 & 0 & 0 & 0 & (1-2\nu)/2 & 0 \\ 0 & 0 & 0 & 0 & 0 & (1-2\nu)/2 \end{bmatrix} \quad (3.2.17)$$

Sendo, na Equação 3.2.17, E o módulo de elasticidade ou módulo de Young e ν o coeficiente de Poisson. O módulo de elasticidade é em geral, obtido experimentalmente, relacionando a deformação axial à tensão aplicada, sendo definido por

$$E \triangleq \frac{\sigma_1}{\varepsilon_1} \quad (3.2.18)$$

A Figura 3.2.8 ilustra o conceito de um corpo de prova.

Figura 3.2.8: Corpo de prova e deformações associadas.

O coeficiente de Poisson por sua vez, relaciona as deformações laterais de um corpo de prova, isto é

$$\nu = -\frac{\varepsilon_1}{\varepsilon_2} = -\frac{\varepsilon_1}{\varepsilon_3} \quad (3.2.19)$$

Todas as Equações apresentadas são válidas para problemas tridimensionais.

Considerando o problema tratado nesta tese, as Equações devem ser adaptadas para problemas bidimensionais. Grande parte desta simplificação se dá eliminando a terceira componente de cada Equação. A Equação 3.2.5, por exemplo, resulta em

$$\frac{\partial \sigma_{i1}}{\partial x_1} + \frac{\partial \sigma_{i2}}{\partial x_2} + b_i = 0 \quad i = 1, 2 \quad (3.2.20)$$

A única observação importante que deve ser feita é em relação à matriz $[C]$, na Equação 3.2.16. Para problemas 2D, duas abordagens são possíveis: estado plano de tensão ou estado plano de deformação. O primeiro ocorre quando consideramos um corpo com uma de suas dimensões muito menor em relação às demais, como mostra a Figura 3.2.9 em que $\Delta x_3 \lll \Delta x_1$ e $\Delta x_3 \lll \Delta x_2$.

Figura 3.2.9: Estado plano de tensão.

Neste caso, as tensões ao longo do eixo x_3 são nulas ($\sigma_3 = 0$), enquanto as deformações devem ser levadas em consideração.

O segundo caso, estado plano de deformação, ocorre quando uma das dimensões é muito maior que as outras duas e a pressão aplicada pode ser considerada constante ao longo deste eixo, como na modelagem de paredes de contenção ou vasos de pressão, como ilustra a Figura 3.2.10 em que $\Delta x_3 \gg \Delta x_1$ e $\Delta x_3 \gg \Delta x_2$ e $P(x_1, x_2, x_3) = P(x_1, x_2)$.

Figura 3.2.10: Estado plano de deformação.

Nesta tese, o problema será tratado como estado plano de deformação, desta forma,

$$[C] = \frac{E(1-\nu)}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1 & \nu/(1-\nu) & 0 \\ \nu/(1-\nu) & 1 & 0 \\ 0 & 0 & (1-2\nu)/2(1-\nu) \end{bmatrix} \quad (3.2.21)$$

É interessante ressaltar que a vantagem de adotar grandezas de engenharia, como tensões e deformações ao invés de grandezas físicas como forças e deslocamentos, é definir valores absolutos independentes das dimensões dos sólidos estudados.

Capítulo 4

O Método dos Elementos Finitos e o Método dos Elementos de Contorno

4.1 Introdução

Para a solução de certos problemas é necessário a utilização de métodos numéricos, ou seja, soluções aproximadas. Isto devido à complexidade do problema, ou até mesmo a impossibilidade de uma solução analítica por domínios irregulares ou pela dificuldade em descrever o fenômeno por meio de funções contínuas. A maioria dos fenômenos físicos pode ser descrita por Equações diferenciais ordinárias, envolvendo apenas uma dimensão, ou parciais envolvendo duas ou mais dimensões. Para a solução numérica de Equações diferenciais podem ser citados como métodos integrais o Método dos Elementos Finitos (MEF) e Método dos Elementos de Contorno (MEC) . Algumas vantagens de cada método são listadas a seguir [39].

- Método dos Elementos Finitos:

- Vantagens:

1. Tecnologia bem desenvolvida;
2. Usado extensivamente em Mecânica dos Sólidos, Térmica e Fluidos e Eletromagnetismo;
3. Integração de funções simples;
4. Matrizes esparsas.

- Desvantagens:

1. Necessita de malhas de domínio;
2. Requer relações integrais para sua formulação;
3. Dificuldade em se tratar de domínios infinitos;
4. Resolução do gradiente de resposta bastante relacionado ao refinamento da malha.

- Método dos Elementos de Contorno:

- Vantagens:

1. Modela contorno e condições de contorno naturalmente;
2. Requer apenas malhas no contorno, o que diminui a quantidade de variáveis a serem solucionadas;
3. Utilizado em Mecânica de Estruturas, Térmica, Acústica, Mecânica dos Flúidos e Eletromagnetismo;
4. Ideal para domínios infinitos;
5. Resolução do gradiente de resposta não relacionado ao refinamento da malha de volume;
6. Matrizes esparsas em blocos, se utilizado sub-regiões;
7. Única técnica para tratamento de certos problemas, como por exemplo, propagação em meios infinitos.

- Desvantagens:

1. Requer relações integrais, além da transformação para Equação de Contorno utilizando soluções fundamentais;
2. Requer integração numérica de funções complexas;

As Figuras 4.1.1 e 4.1.2 ilustram exemplos de malhas de elementos de contorno e de elementos finitos.

Figura 4.1.1: Malha para elementos de contorno.

Figura 4.1.2: Malha para elementos finitos.

É possível perceber pelo exposto acima, que o MEC apresenta grande vantagem em relação ao MEF nos seguintes pontos:

- As criação de malhas de contorno utiliza algoritmos mais simples que para malhas de domínio utilizadas no MEF;
- As matrizes resultantes do MEC são menores comparadas aos do MEF, o que implica em menor tempo de solução dos sistemas e menor consumo de memória;
- Graças à formulação do MEC, as matrizes resultantes são melhores condicionadas que o MEF, como será visto mais adiante no texto;

A seguir, será apresentada brevemente a formulação para MEF e MEC para problemas de elastostática.

4.2 O Método dos Elementos Finitos

O MEF é uma das ferramentas numéricas mais populares e amplamente empregadas atualmente, sendo usadas em problemas de Mecânica dos Sólidos, Eletromagnetismo, Mecânica dos Flúidos [5] entre outras áreas da Ciência. Isto deve-se a formulação e programação relativamente simples. A seguir, será discutida formulação para MEF linear, isto é, considerando pequenos deslocamentos.

4.2.1 Formulação das Equações de Elementos Finitos

A formulação do MEF para problemas lineares toma a Equação de equilíbrio 3.2.20, que ponderada por um vetor de funções peso $\vec{\omega}_i$ com $i = 1, 2$, e integrada sobre todo o domínio de um elemento Ω_e como mostra a Equação 4.2.1 [5].

$$\int_{\Omega_e} \vec{\omega}_1 \left(\frac{\partial \sigma_{11}}{\partial x_1} + \frac{\partial \sigma_{12}}{\partial x_2} + b_1 \right) d\Omega_e = 0 \quad (4.2.1)$$

$$\int_{\Omega_e} \vec{\omega}_2 \left(\frac{\partial \sigma_{21}}{\partial x_1} + \frac{\partial \sigma_{22}}{\partial x_2} + b_1 \right) d\Omega_e = 0 \quad (4.2.2)$$

Considerando $b_1 = b_2 = 0$, a Equações 4.2.1 e 4.2.2 é integrada por partes, resultando nas Equações 4.2.3 e 4.2.4:

$$\int_{\Omega_e} \left(\frac{\partial \vec{\omega}_1}{\partial x_1} \sigma_{11} + \frac{\partial \vec{\omega}_1}{\partial x_2} \sigma_{12} \right) d\Omega_e - \int_{\Gamma_e} \vec{\omega}_1 t_1 d\Gamma_e = 0 \quad (4.2.3)$$

$$\int_{\Omega_e} \left(\frac{\partial \vec{\omega}_2}{\partial x_1} \sigma_{11} + \frac{\partial \vec{\omega}_2}{\partial x_2} \sigma_{12} \right) d\Omega_e - \int_{\Gamma_e} \vec{\omega}_2 t_1 d\Gamma_e = 0 \quad (4.2.4)$$

Reescrevendo as tensões em função das deformação, isto é, aplicando a Lei de Hooke da Equação 3.2.11, sua forma matricial resulta na Equação 4.2.5:

$$\int_{\Omega_e} \begin{bmatrix} \frac{\partial \vec{\omega}_1}{\partial x_1} & 0 & \frac{\partial \vec{\omega}_1}{\partial x_2} \\ 0 & \frac{\partial \vec{\omega}_2}{\partial x_1} & \frac{\partial \vec{\omega}_2}{\partial x_2} \end{bmatrix} [C] \{\varepsilon\} d\Omega_e - \int_{\Gamma_e} \begin{Bmatrix} \vec{\omega}_1 t_1 \\ \vec{\omega}_2 t_2 \end{Bmatrix} d\Gamma_e = 0 \quad (4.2.5)$$

Utilizando a Equação de compatibilidade cinemática 3.2.10, a Equação finalmente é reescrita em função dos deslocamentos,

$$\int_{\Omega_e} \begin{bmatrix} \frac{\partial \vec{\omega}_1}{\partial x_1} & 0 & \frac{\partial \vec{\omega}_1}{\partial x_2} \\ 0 & \frac{\partial \vec{\omega}_2}{\partial x_1} & \frac{\partial \vec{\omega}_2}{\partial x_2} \end{bmatrix} [C] \begin{Bmatrix} \frac{\partial u_1}{\partial x_1} \\ \frac{\partial u_2}{\partial x_2} \\ \frac{\partial u_2}{\partial x_1} + \frac{\partial u_1}{\partial x_2} \end{Bmatrix} d\Omega_e - \int_{\Gamma_e} \begin{Bmatrix} \vec{\omega}_1 t_1 \\ \vec{\omega}_2 t_2 \end{Bmatrix} d\Gamma_e = 0_{2 \times 1} \quad (4.2.6)$$

Os deslocamentos da Equação 4.2.6, podem então, ser interpolados em cada elemento finito, como mostra a Equação 4.2.7, sendo $\alpha_k(x_1, x_2)$ as funções de inetrepolação e u_i^k o deslocamento segundo a i -ésima direção do k -ésimo nó do elemento:

$$u_i = \sum_{k=1}^N \alpha_k(x_1, x_2) u_i^k \quad (4.2.7)$$

Na Equação 4.2.7, os valores u_i^k são constantes no domínio Ω e representam os deslocamentos nodais em cada vértice do elemento.

As funções ω_i podem ser determinadas utilizando vários métodos, por exemplo, o Método de Colocação, Mínimos Quadrados ou Galerkin [44]. Nesta tese, foi adotado o método de Galerkin, que assume que:

$$\omega_i^k = \frac{\partial u_i}{\partial u_i^k}, \quad 1 \leq k \leq N \quad (4.2.8)$$

Desta forma, assumindo uma discretização por elementos triangulares de primeira ordem, como mostram as Figuras 4.2.1a, 4.2.1b e 4.2.1c, as funções α_i escritas em função dos vértices (x_1^k, x_2^k) $k = 1, 2, 3$ de um determinado triângulo, são dadas pelas Equações 4.2.9, 4.2.10 e 4.2.11 em que A é a área do elemento triangular, ilustradas na Figura 4.2.1:

$$\alpha_1(x_1, x_2) = \frac{1}{2A} [(x_1^2 x_2^3 - x_1^3 x_2^2) + (x_2^2 - x_2^3) x_1 + (x_1^3 - x_1^2) x_2] \quad (4.2.9)$$

$$\alpha_2(x_1, x_2) = \frac{1}{2A} [(x_1^3 x_2^1 - x_1^1 x_2^3) + (x_2^3 - x_2^1) x_1 + (x_1^1 - x_1^3) x_2] \quad (4.2.10)$$

$$\alpha_3(x_1, x_2) = \frac{1}{2A} [(x_1^1 x_2^2 - x_1^2 x_2^1) + (x_2^1 - x_2^2) x_1 + (x_1^2 - x_1^1) x_2] \quad (4.2.11)$$

(a)

(b)

(c)

Figura 4.2.1: Funções de interpolação para triângulos de primeira ordem.

Desta forma, a Equação 4.2.8 resulta em:

$$\omega_i^k(x_1, x_2) = \alpha_k(x_1, x_2) \quad 1 \leq k \leq 3; i = 1, 2 \quad (4.2.12)$$

$$\begin{aligned}
& \underbrace{\int_{\Omega_e} \begin{bmatrix} \frac{\partial \alpha_1}{\partial x_1} & 0 & \frac{\partial \alpha_1}{\partial x_2} \\ 0 & \frac{\partial \alpha_1}{\partial x_2} & \frac{\partial \alpha_1}{\partial x_1} \\ \frac{\partial \alpha_2}{\partial x_1} & 0 & \frac{\partial \alpha_2}{\partial x_2} \\ 0 & \frac{\partial \alpha_2}{\partial x_2} & \frac{\partial \alpha_2}{\partial x_1} \\ \frac{\partial \alpha_3}{\partial x_1} & 0 & \frac{\partial \alpha_3}{\partial x_2} \\ 0 & \frac{\partial \alpha_3}{\partial x_2} & \frac{\partial \alpha_3}{\partial x_1} \end{bmatrix} [C] \begin{bmatrix} \frac{\partial \alpha_1}{\partial x_1} & 0 & \frac{\partial \alpha_2}{\partial x_1} & 0 & \frac{\partial \alpha_3}{\partial x_1} & 0 \\ 0 & \frac{\partial \alpha_1}{\partial x_2} & 0 & \frac{\partial \alpha_2}{\partial x_2} & 0 & \frac{\partial \alpha_3}{\partial x_2} \\ \frac{\partial \alpha_1}{\partial x_2} & \frac{\partial \alpha_1}{\partial x_1} & \frac{\partial \alpha_2}{\partial x_2} & \frac{\partial \alpha_2}{\partial x_1} & \frac{\partial \alpha_3}{\partial x_2} & \frac{\partial \alpha_3}{\partial x_1} \end{bmatrix} d\Omega_e}_{[K]^e} \underbrace{\begin{Bmatrix} u_1^1 \\ u_2^1 \\ u_1^2 \\ u_2^2 \\ u_1^3 \\ u_1^3 \end{Bmatrix}}_{\{u\}} - \underbrace{\int_{\Gamma_e} \begin{Bmatrix} z_1 t_1^1 \\ z_1 t_2^1 \\ z_2 t_1^2 \\ z_2 t_2^2 \\ z_3 t_1^3 \\ z_3 t_2^3 \end{Bmatrix}}_{\{f\}^e} d\Gamma_e = 0_{6 \times 1} \quad (4.2.13)
\end{aligned}$$

É importante notar que as derivadas das funções $\alpha_i(x_1, x_2)$ são constantes, desta forma, a Equação 4.2.13 resulta em

$$[K]^e = [B]^T [C] [B] A \quad (4.2.14)$$

$$[B] = \begin{bmatrix} (x_2^2 - x_2^3) & 0 & (x_2^3 - x_2^1) & 0 & (x_2^1 - x_2^2) & 0 \\ 0 & (x_1^3 - x_1^2) & 0 & (x_1^1 - x_1^3) & 0 & (x_1^2 - x_1^1) \\ (x_1^3 - x_1^2) & (x_2^2 - x_2^3) & (x_1^1 - x_1^3) & (x_2^3 - x_2^1) & (x_1^2 - x_1^1) & (x_2^1 - x_2^2) \end{bmatrix} \frac{1}{2A} \quad (4.2.15)$$

Para elementos de primeira ordem, não é necessário nenhuma integração numérica para a matriz de rigidez, enquanto os elementos vetor de forças de superfície (segundo termo da Equação 4.2.13) possuem solução analítica, facilmente obtida pela integração de polinômios de primeira ordem.

Resta determinar a matriz de rigidez e o vetor de força globais $[K]^G$ e $\{f\}^G$. Para isto, é necessário utilizar o conceito de matriz de incidência nodal, ou conectividade nodal. A Figura 4.2.2 ilustra uma malha de elementos triangulares com dois elementos. Na Figura 4.2.2, é mostrada a numeração global dos nós e a numeração local, para o elemento 1.

Figura 4.2.2: Exemplo de malha de elementos finitos com dois elementos triangulares de primeira ordem.

A matriz de conectividade nodal para a malha da Figura 4.2.2 é

$$[W] = \begin{bmatrix} 1 & 1 & 2 & 4 \\ 2 & 2 & 3 & 4 \end{bmatrix} \quad (4.2.16)$$

Com a ajuda da matriz da Equação 4.2.16, a matriz de rigidez global pode ser montada utilizando programação serial como mostra o algoritmo da Figura 4.2.3. O vetor $\{f\}^G$ é calculado utilizando o mesmo raciocínio.

```

for element = 1: numberOfElements
    nodes = W(element, 2: 4);
    for lin = 1: 3
        idl1 = (2*lin - 1): 2*lin;
        idg1 = (2*nodes(lin) - 1): 2*nodes(lin);
        for col = 1: 3
            idl2 = (2*col - 1): 2*col;
            idg2 = (2*nodes(col) - 1): 2*nodes(col);
            Kg(idg1, idg2) = Kg(idg1, idg2) + Ke(idl1, idl2);
        end
    end
end

```

Figura 4.2.3: Programa para montagem da matriz de rigidez global.

O sistema de equações resulta em:

$$[K]^G \{u\} = \{f\}^G \quad (4.2.17)$$

Resta incorporar as condições de contorno para que o sistema linear da Equação 4.2.17 possa ser resolvido. Para isto, é feita uma permutação entre as colunas de $[K]^G$ com valores de deslocamentos prescritos e as linhas de $\{f\}^G$ com as forças nodais desconhecidas, em que na maior parte dos casos, são as reações de apoio. Isto resulta num sistema do tipo $[A]\{x\} = \{y\}$, em que a matriz $[A]$ é invertível.

4.3 O Método dos Elementos de Contorno

4.3.1 Formulação da Equação Integral de Contorno para Elastostática

Nesta seção será apresentada a formulação de elementos de contorno para Elastostática [27, 6, 7, 29], para isto as Equações de equilíbrio 3.2.20, válidas em um domínio Ω cujo contorno $\partial\Omega = \Gamma$ como ilustra a Figura 4.3.1, são ponderadas por funções peso, que aqui será considerada como um campo de deslocamentos arbitrários, u_i^* e integradas em todo o domínio Ω , como mostra a Equação 4.3.1

Figura 4.3.1: Domínio da Equação diferencial e seu contorno.

$$\sum_{i=1}^2 \left[\int_{\Omega} u_i^* \left(\frac{\partial \sigma_{i1}}{\partial x_1} + \frac{\partial \sigma_{i2}}{\partial x_2} \right) d\Omega + \int_{\Omega} \omega_i b_i d\Omega \right] = 0 \quad (4.3.1)$$

Integrando por partes o primeiro termo da Equação 4.3.1:

$$\sum_{i=1}^2 \left[\int_{\Omega} u_i^* \frac{\partial \sigma_{il}}{\partial x_l} d\Omega \right] = \sum_{i=1}^2 \left\{ \int_{\Omega} \left[\frac{\partial (\sigma_{il} u_i^*)}{\partial x_l} - \sigma_{il} \frac{\partial u_i^*}{\partial x_l} \right] d\Omega \right\} \quad (4.3.2)$$

Utilizando o Teorema de Gauss [42], o primeiro termo do lado esquerdo da Equação 4.3.2 é escrito como uma integral sob o contorno Γ , como mostra a Equação 4.3.3

$$\int_{\Omega} \frac{\partial (\sigma_{il} u_i^*)}{\partial x_l} d\Omega = \int_{\Gamma} \sigma_{il} u_i^* n_l d\Gamma \quad (4.3.3)$$

sendo n_j a j -ésima componente do vetor normal unitário a Γ (Figura 4.3.2).

Figura 4.3.2: Normal ao contorno Γ .

A equação de equilíbrio pode ser reescrita da seguinte forma:

$$\sum_{i=1}^2 \left[\underbrace{\int_{\Gamma} (\sigma_{i1}n_1 + \sigma_{i2}n_2) u_i^* d\Gamma}_{T1} - \underbrace{\int_{\Omega} \left(\sigma_{i1} \frac{\partial u_i^*}{\partial x_1} + \sigma_{i2} \frac{\partial u_i^*}{\partial x_2} \right) d\Omega}_{T2} + \underbrace{\int_{\Omega} u_i^* b_i d\Omega}_{T3} \right] = 0 \quad (4.3.4)$$

De acordo com a Equação 3.2.4, o termo T1 da Equação 4.3.4 pode ser escrito como

$$\int_{\Gamma} (\sigma_{i1}n_1 + \sigma_{i2}n_2) u_i^* d\Gamma = \int_{\Gamma} t_i u_i^* d\Gamma \quad (4.3.5)$$

O termo T2 pode ser resolvido observando que:

$$\sum_{k=1}^2 \sum_{i=1}^2 \sigma_{ik} \frac{\partial u_i^*}{\partial x_k} = \sum_{k=1}^2 \sum_{i=1}^2 \sigma_{ik} \frac{\partial u_k^*}{\partial x_i} \quad (4.3.6)$$

E lembrando da Equação 3.2.10, a Equação 4.3.6 pode ser escrita em função das deformações associadas aos deslocamentos u_{ik}^* , como mostra a Equação 4.3.7:

$$u_{ik}^* = \sum_{k=1}^2 \sum_{i=1}^2 \sigma_{ik} \varepsilon_{ik}^* \quad (4.3.7)$$

Fazendo $b_i = 0$, isto é, as forças de corpo são nulas, o termo T3 é nulo, a Equação 4.3.4 resulta em:

$$\sum_{i=1}^2 \left[\underbrace{\int_{\Gamma} t_i u_i^* d\Gamma}_{T1} - \underbrace{\int_{\Omega} (\sigma_{i1} \varepsilon_{i1}^* + \sigma_{i2} \varepsilon_{i2}^*) d\Omega}_{T2} \right] = 0 \quad (4.3.8)$$

O termo T2 pode ser resolvido como proposto por Kane [39] que, utilizando identidades da

aplicação do método de trabalho virtual, mostra que:

$$\int_{\Omega} (\sigma_{i1}\varepsilon_{i1}^* + \sigma_{i2}\varepsilon_{i2}^*) d\Omega = \int_{\Gamma} t_i^* u_i d\Gamma + \int_{\Omega} b_i^* u_i d\Omega \quad (4.3.9)$$

Os termos u_i^* , t_i^* são chamados de estados auxiliares e são a resposta em termos de forças de superfície e deslocamentos à aplicação de um carregamento pontual b_i^* em um corpo elástico ilimitado.

Neste ponto, é importante observar que em consequência de uma força aplicada na i -ésima direção, implica em um campo de deslocamentos $\vec{u}$, assim, se as forças são aplicadas em duas direções simultaneamente, os deslocamentos são a superposição das componentes referentes a cada excitação [15], como mostra a Equação 4.3.10

$$u_i^* = u_{i1}^* n_1 + u_{i2}^* n_2 \quad (4.3.10)$$

De forma análoga, o tensor de tensões passa a ser um tensor de ordem 3, isto porque para cada direção existe um estado de tensão σ_{ij}^* diferente, e assim, o tensor de Cauchy é dado por

$$t_{il}^* = \sum_{k=1}^2 \sigma_{ilk}^* n_k \quad (4.3.11)$$

Substituindo t_{ij}^* e u_{ij}^* na Equação 4.3.9,

$$\sum_{i=1}^2 \left[\int_{\Gamma} (t_i u_{il}^* - t_{il}^* u_i) d\Gamma + \int_{\Omega} b_i^* u_i d\Omega \right] = 0 \quad (4.3.12)$$

O carregamento pontual da Equação 4.3.12 pode ser representado como mostra a Equação 4.3.13:

$$b_i^* = \delta(\vec{x} - \vec{\xi}) n_i \quad (4.3.13)$$

sendo $\delta(\vec{x} - \vec{\xi})$ a representação Delta de Dirac, a Equação 4.3.13 é a representação de uma carga pontual aplicada ao sólido no ponto $\vec{\xi}$ (também chamado de “ponto fonte”) na direção $\vec{n}$.

Resta determinar a forma das funções u_{il}^* e t_{il}^* . Para isto, a equação de equilíbrio 3.2.20 é resolvida para os deslocamentos num domínio ilimitado, aplicando a lei de Hooke (Equação 3.2.11) e as equações de compatibilidade cinemática (Equação 3.2.10), tendo como excitação carregamentos pontuais (Equação 4.3.13), como mostram por exemplo, Gao and Davies [27], Gaul et al. [29]. Para problemas isotrópicos, u_{il}^* é dada pela Equação 4.3.14, sendo δ_{ij} o Delta de Kronecker e r definido pela Equação 4.3.15, em que $\vec{\xi}$ representa o ponto de aplicação da carga pontual e $\vec{x}$ o ponto onde a resposta à aplicação desta carga é medida.

$$u_{ij}^* (\vec{x}, \vec{\xi}) = \frac{1}{8\mu\pi(1-\nu)} \left\{ \left[(3-4\nu) \log \left(\frac{1}{r} \right) \right] \delta_{ij} + \frac{\partial r}{\partial x_i} \frac{\partial r}{\partial x_j} \right\} \quad (4.3.14)$$

$$r = \sqrt{(x_1 - \xi_1)^2 + (x_2 - \xi_2)^2} \quad (4.3.15)$$

A solução para um problema de excitação pontual é conhecida como função de Green. A função de Green correspondente para t_{ij}^* é dada pela Equação 4.3.16.

$$t_{ij}^* (\vec{x}, \vec{\xi}) = -\frac{1}{4\pi(1-\nu)r} \left\{ (1-2\nu) \left[\frac{\partial r}{\partial x_i} n_j - \frac{\partial r}{\partial x_j} n_i \right] + \left(\frac{\partial r}{\partial x_1} n_1 + \frac{\partial r}{\partial x_2} n_2 \right) \left[(1-2\nu) \delta_{ij} + 2 \frac{\partial r}{\partial x_i} \frac{\partial r}{\partial x_j} \right] \right\} \quad (4.3.16)$$

As propriedades da representação Delta de Dirac são descritas pela Equações 4.3.17, 4.3.18 e 4.3.19,

$$\delta (\vec{x}, \vec{\xi}) = \begin{cases} 0 & \vec{x} \neq \vec{\xi} \\ \infty & \vec{x} = \vec{\xi} \end{cases} \quad (4.3.17)$$

$$\int_{-\infty}^{\infty} \delta (\vec{x}, \vec{\xi}) d\Omega = 1 \quad (4.3.18)$$

$$\int_{-\infty}^{\infty} f(x) \delta(x, a) d\Omega = f(a) \quad (4.3.19)$$

e a integral de domínio na Equação 4.3.12 se torna dependente da escolha da localização do ponto fonte $\vec{\xi}$. Desta forma, as seguintes situações são definidas pela Equação 4.3.17:

$$\begin{cases} \int_{\Omega} b_i^* u_j d\Omega = \delta_{ij} u_i & \vec{\xi} \in \Omega \\ \int_{\Omega} b_i^* u_j d\Omega = [0]_{2 \times 2} & \vec{\xi} \notin \Omega \end{cases} \quad (4.3.20)$$

O primeiro caso da Equação 4.3.20 pode ser utilizado para cálculo de deslocamentos no interior do domínio do problema. O segundo caso é utilizado na chamada formulação regular do MEC. No entanto, a literatura [39] aponta o uso da formulação singular, onde $\vec{\xi} \in \Gamma$, como a formulação que aponta melhores resultados em relação à formulação regular. O valor da integral é dado de forma analítica pela Equação 4.3.21

$$\int_{\Omega} b_i^* u_j d\Omega = \frac{\Delta\theta}{2\pi} \delta_{ij} u_i \quad \vec{\xi} \in \Gamma \quad (4.3.21)$$

sendo $\Delta\theta$ o ângulo interno, como mostra a Figura 4.3.3.

Figura 4.3.3: Ângulo interno.

Por fim, a Equação Integral de Contorno é dada por

$$\sum_{i=1}^2 \left[\int_{\Gamma} (t_i u_{ij}^* - t_{ij}^* u_i) d\Gamma + \frac{\Delta\theta}{2\pi} \delta_{ij} u_i \right] = 0 \quad (4.3.22)$$

4.3.2 Discretização em Elementos de Contorno

Tendo a EIC (Equação 4.3.22), é possível discretizar o contorno Γ em uma sequência de N_e elementos de contorno, como mostra a Figura 4.3.4.

Figura 4.3.4: Contorno discretizado.

Sendo que

$$\Gamma \cong \bigcup_{i=1}^{N_e} \Gamma_i \quad (4.3.23)$$

Isto permite interpolar as grandezas dentro de cada elemento como mostra a Equação 4.3.24:

$$\alpha \cong \sum_{i=1}^N h_i(\eta) \alpha_i \quad (4.3.24)$$

A aproximação é realizada tanto para a geometria quanto para as grandezas físicas. Na Equação 4.3.24, $h_i(\eta)$ é uma função de forma, que aqui, será considerada polinomial.

Para interpolar as grandezas físicas, isto é, forças de superfície e deslocamentos, as funções de interpolação serão consideradas constantes,

$$h(\eta) = 1 \quad (4.3.25)$$

O uso da Equação 4.3.25 como função de interpolação significa que não há variação do fenômeno ao longo do elemento e impõe uma descontinuidade entre elementos adjacentes (Figura 4.3.5).

Figura 4.3.5: Descontinuidade entre dois elementos adjacentes causada pelo uso de funções constantes.

Esta característica é inerente ao uso de funções de interpolação descontínuas mas, no entanto, ela permite uma implementação eficiente do método em computação paralela, como será discutido mais adiantes neste capítulo.

A EIC pode ser discretizada substituindo as Equações 4.3.24 e 4.3.23 em 4.3.22:

$$\sum_{e=1}^{N_e} \sum_{i=1}^2 \left[\int_{\Gamma_e} u_{ij}^* d\Gamma_e t_i^e - \int_{\Gamma_e} t_{ij}^* d\Gamma_e u_i^e + \frac{\Delta\theta}{2\pi} \delta_{ij} u_i \right] = 0 \quad (4.3.26)$$

Na Equação 4.3.26, t_i^e e u_i^e são as forças de superfície e deslocamentos associados ao elemento e . A geometria do problema, por sua vez, é discretizada utilizando uma variável paramétrica, isto é:

$$x_1(\eta) = h_1(\eta) x_1^1 + h_2(\eta) x_1^2 \quad (4.3.27)$$

$$x_2(\eta) = h_1(\eta) x_2^1 + h_2(\eta) x_2^2 \quad (4.3.28)$$

sendo

$$h_1(\eta) = \frac{1}{2}(1 - \eta) \quad (4.3.29)$$

$$h_2(\eta) = \frac{1}{2}(1 + \eta) \quad (4.3.30)$$

$$-1 \leq \eta \leq 1 \quad (4.3.31)$$

o conceito é ilustrado na Figura 4.3.6.

Figura 4.3.6: Parametrização da geometria.

As integrais da Equação 4.3.26 são realizadas num domínio normalizado

$$\int_{\Gamma_e} u_{ij}^* d\Gamma_e = \int_{-1}^1 u_{ij}^* J(\eta) d\Gamma_e(\eta) \quad (4.3.32)$$

$$\int_{\Gamma_e} t_{ij}^* d\Gamma_e = \int_{-1}^1 t_{ij}^* J(\eta) d\Gamma_e(\eta) \quad (4.3.33)$$

Nas Equações 4.3.32 e 4.3.33, $J(\eta)$ é o Jacobiano que relaciona Γ_i com o domínio normalizado da Equação 4.3.31. Para funções de primeira ordem, $J(\eta) = L_e/2$.

É possível reescrever a Equação 4.3.26 de forma matricial como segue:

$$[\bar{H}] \{u\} = [\bar{G}] \{t\} \quad (4.3.34)$$

sendo

$$[\bar{H}] = \begin{bmatrix} [\bar{H}]^1 & \dots & [\bar{H}]^{N_e} \end{bmatrix}^{2 \times 2N_e} \quad (4.3.35)$$

$$[\bar{G}] = \begin{bmatrix} [\bar{G}]^1 & \dots & [\bar{G}]^{N_e} \end{bmatrix}^{2 \times 2N_e} \quad (4.3.36)$$

As submatrizes $[\bar{H}]^e$ e $[\bar{G}]^e$ são dadas pelas Equações 4.3.37 e 4.3.38.

$$[\bar{H}]^e = \int_{-1}^1 t_{ij}^* J(\eta) d\Gamma_e + \frac{\Delta\theta}{2\pi} \delta_{ij} \in \mathbb{R}^{2 \times 2} \quad (4.3.37)$$

$$[\bar{G}]^e = \int_{-1}^1 u_{ij}^* J(\eta) d\Gamma_e \in \mathbb{R}^{2 \times 2} \quad (4.3.38)$$

$$\{u\} = \left\{ \left\{ \begin{matrix} u_1 & u_2 \end{matrix} \right\}^1, \dots, \left\{ \begin{matrix} u_1 & u_2 \end{matrix} \right\}^{N_e} \right\}^T \quad (4.3.39)$$

$$\{t\} = \left\{ \left\{ \begin{matrix} t_1 & t_2 \end{matrix} \right\}^1, \dots, \left\{ \begin{matrix} t_1 & t_2 \end{matrix} \right\}^{N_e} \right\}^T \quad (4.3.40)$$

$$[\bar{H}] \in \mathbb{R}^{2 \times 2N_e} \quad (4.3.41)$$

$$[\bar{G}] \in \mathbb{R}^{2 \times 2N_e} \quad (4.3.42)$$

e representam a resposta em termos de deslocamentos e forças de superfície de uma carga pontual aplicada sobre um ponto $\vec{\xi}$ em todo o contorno.

É possível escrever um sistema de Equações de ordem $2N_e \times 2N_e$ variando a localização do ponto fonte $\vec{\xi}$ ao redor da malha de elementos de contorno. Desta forma, para um elemento com função de interpolação constante, o ponto fonte é o ponto médio de cada elemento e as Equações 4.3.35 e 4.3.36 ficam

$$[H] = \begin{bmatrix} [\bar{H}]_1^1 & \dots & [\bar{H}]_1^{N_e} \\ \vdots & \ddots & \vdots \\ [\bar{H}]_{N_e}^1 & \dots & [\bar{H}]_{N_e}^{N_e} \end{bmatrix} \quad (4.3.43)$$

$$[G] = \begin{bmatrix} [\bar{G}]_1^1 & \dots & [\bar{G}]_1^{N_e} \\ \vdots & \ddots & \vdots \\ [\bar{G}]_{N_e}^1 & \dots & [\bar{G}]_{N_e}^{N_e} \end{bmatrix} \quad (4.3.44)$$

$$[H] \{u\} = [G] \{t\} \quad (4.3.45)$$

onde o sub-índice em cada submatriz indica o ponto fonte correspondente. As integrais para o cálculo dos blocos diagonais de 4.3.43 e 4.3.44 são integrais singulares, pois são realizadas sobre o elemento que contém o ponto fonte e isto implica que na Equação 4.3.15, $r \rightarrow 0$. A integral da

Equação 4.3.32 possui singularidade fraca, portanto, pode ser tratada no sentido do Valor Principal de Cauchy [7], enquanto a Equação 4.3.33 possui singularidade forte e pode ser tratada utilizando o que se chama de “Argumento de Corpo Rígido”, como descrito em Becker [6]. Este recurso permite que as diagonais de 4.3.44 sejam calculadas aplicando na Equação 4.3.45 condições de contorno de forças de superfície nulas e de deslocamentos unitários em cada direção e assim, cada elemento singular fica:

$$G_{ii} = - \sum_{\substack{j=1 \\ j \neq i}}^{N_e} G_{ij} \quad (4.3.46)$$

Para um problema bem posicionado, como mostrado na Figura 4.3.7, em um mesmo nó ou é conhecida a força de superfície aplicada, ou o deslocamento, nunca as duas grandezas. Isto faz com que o sistema de Equações seja sempre uma matriz quadrada, de ordem $2N_e \times 2N_e$, resultando num sistema como na Equação 4.3.47.

Figura 4.3.7: Condições de contorno para um problema bem posicionado.

$$[A] \{x\} = \{b\} \quad (4.3.47)$$

No sistema linear resultante, a matriz $[A]$ é a coleção de colunas das matrizes $[H]$ e $[G]$ correspondentes às incógnitas do problema, enquanto o vetor $\{b\}$ resulta da aplicação das condições de contorno. O sistema é resolvido para o vetor de incógnitas $\{x\}$.

Para elementos descontínuos de qualquer ordem, as matrizes $[H]$ e $[G]$ são sempre quadradas e graças a variação da posição do ponto fonte, suas linhas são sempre linearmente independentes. Sendo assim, uma forma alternativa de se resolver o problema via MEC, é como mostra a Equação 4.3.48

$$[G]^{-1} [H] \{u\} = \{t\} \quad (4.3.48)$$

Desta forma, é possível montar um sistema como da Equação 4.3.47 utilizando a mesma metodologia que em elementos finitos. Fazendo isto, a malha de elementos de contorno é considerada um “super elemento finito”.

4.3.3 Método de Sub-regiões

Na seção anterior, toda a formulação é referente a apenas um tipo de material, isto é, o método não trata não homogeneidades implicitamente. No entanto, várias técnicas são encontradas na literatura para tratar meios não homogêneos via elementos de contorno. Entre eles, podem ser citados alguns trabalhos relevantes, como por exemplo Lu and Wu [49], Santiago et al. [67], Lu [48], Kamiya and Koide [38], onde os autores manipulam as matrizes $[H]$ e $[G]$ para cada sub-região. Becker [6] apresenta uma formulação para a função de Green para sub-regiões.

Nesta tese, foi adotada a metodologia utilizada por dos Santos Hoeffl [21], compatibilizando os deslocamentos e impondo o equilíbrio de forças entre as interfaces de duas sub-regiões. Esta abordagem permite o cálculo das matrizes de cada sub-região separadamente e também a ordenação prévia das colunas de cada uma delas, como será mostrado a seguir e, assim, permite uma melhor paralelização em comparação a outras técnicas, que utilizam, por exemplo, inversão de matrizes. A Figura 4.3.8 mostra um exemplo de meio não homogêneo e a Figura 4.3.9 as malhas de elementos de contorno para cada uma das sub-regiões.

Figura 4.3.8: Modelo com três sub-regiões.

Figura 4.3.9: Malha de elementos de contorno para o exemplo da Figura 4.3.8.

Seja Ω_i o domínio da sub-região i , Γ_i o contorno deste domínio e I_{ij} a interface entre os domínios adjacentes Ω_i e Ω_j é necessário impor as seguintes condições:

$$\Omega_i \cap \Omega_j = \emptyset \quad i \neq j \quad (4.3.49)$$

$$\Gamma_i \cap \Gamma_j = I_{ij} \quad i \neq j \quad (4.3.50)$$

Considerando a interface entre duas sub-regiões, Figura 4.3.10

Figura 4.3.10: Equilíbrio de forças e compatibilidade de deslocamentos na interface de duas sub-regiões.

tem-se que, por equilíbrio de forças,

$$\vec{t}_1 + \vec{t}_2 = 0 \quad (4.3.51)$$

e a compatibilidade cinemática

$$\vec{u}_1 - \vec{u}_2 = 0 \quad (4.3.52)$$

desta forma, tendo dois conjuntos de matrizes, $[H]^1$, $[G]^1$ e $[H]^2$, $[G]^2$, onde o super-índice indica a sub-região a que se referem, é possível classificar as variáveis entre aquelas presentes nas interfaces, $\vec{t}_i^I$ e $\vec{u}_i^I$, e aquelas que pertencem apenas a uma sub-região, $\vec{t}_i^E$ e $\vec{u}_i^E$, as matrizes $[H]^i$ e $[G]^i$ podem ser particionadas como

$$\begin{bmatrix} H_I^i & H_E^i \end{bmatrix} \begin{Bmatrix} \vec{u}_i^I \\ \vec{u}_i^E \end{Bmatrix} = \begin{bmatrix} G_I^i & G_E^i \end{bmatrix} \begin{Bmatrix} \vec{t}_i^I \\ \vec{t}_i^E \end{Bmatrix} \quad (4.3.53)$$

Aplicando as Equações 4.3.51 e 4.3.52, tem-se para um problema com duas sub-regiões, o seguinte sistema matricial

$$\begin{bmatrix} [H]_I^1 & [H]_E^1 & [0] \\ -[H]_I^2 & [0] & [H]_E^2 \end{bmatrix} \begin{Bmatrix} \vec{u}^I \\ \vec{u}_1^E \\ \vec{u}_2^E \end{Bmatrix} = \begin{bmatrix} [G]_I^1 & [G]_E^1 & [0] \\ [G]_I^2 & [0] & [G]_E^2 \end{bmatrix} \begin{Bmatrix} \vec{t}^I \\ \vec{t}_1^E \\ \vec{t}_2^E \end{Bmatrix} \quad (4.3.54)$$

onde as matrizes de zeros na Equação 4.3.53 possuem dimensões compatíveis para o sistema. É possível aplicar o conceito em N_R regiões e assim, construir as matrizes $[H]$ e $[G]$.

4.4 Malhas para Elementos de Contorno e Elementos Finitos

A malha para elementos de contorno é gerada por uma segmentação semi-autimática, onde o operador informa a quantidade de possíveis lesões na imagem e em seguida, seleciona dois pontos, referentes ao diâmetro da lesão em uma imagem I_1 (imagem de referência, no instante t_1). Então, é aplicado um *chain algorithm* para determinar a conectividade dos nós. Cada material é considerado como uma sub-região diferente, isto é, a sub-região 1 Ω_1 é o background, enquanto as lesões são sub-domínios Ω_i da imagem I_1 . Considerando o domínio da imagem como Ω , tem-se que a união dos sub-domínios Ω_i é o próprio domínio Ω (Equação 4.4.1); a interseção dos sub-domínios Ω_i com $i \neq 1$ é nula (Equação 4.4.2) enquanto a interseção do domínio do background com os domínios das lesões, resulta no contorno de cada lesão (Equação 4.4.3).

$$\Omega = \bigcup_{i=1}^{N_{mat}} \Omega_i \quad (4.4.1)$$

$$\Omega_i \cap \Omega_j = \emptyset \quad i \neq j; i, j \neq 1 \quad (4.4.2)$$

$$\Omega_1 \cap \Omega_i = \Gamma_i \quad (4.4.3)$$

Assim, dado o domínio da imagem, o domínio do background é criado subtraindo os domínios de cada lesão. As Figuras 4.4.1a e 4.4.1b mostram as malhas para uma imagem com três lesões.

Figura 4.4.1: Malha para elementos de contorno considerando background e três lesões: (a) Malha para o background; (b) Malha para as lesões.

Desta forma, a malha de elementos de contorno é definida completamente por uma lista de coordenadas nodais e uma lista de conectividade nodal.

Para que os resultados obtidos por elementos de contorno possam ser comparados aos de elementos finitos, é necessário que a quantidade de elementos entre um método e outro seja coerente. Para isto, a malha de contorno é utilizada para criar a malha de elementos finitos, de forma que cada elemento de contorno seja a aresta de um elemento finito na fronteira entre as sub-regiões. Foi desenvolvido um programa em Matlab para, a partir das malhas de contorno e utilizando o *software* GMesh [31], gerar uma malha de elementos finitos triangulares de primeira ordem, de forma a garantir a compatibilidade entre as duas malhas e assegurar uma quantidade mínima de triângulos para preencher o domínio. A malha de elementos finitos para o modelo da Figura 4.4.1 é mostrada na Figura 4.4.2 e na Figura 4.4.3 é possível perceber em detalhe a compatibilidade entre a malha de contorno e a malha de elementos finitos.

Figura 4.4.2: Malha de elementos finitos para background e três lesões.

Figura 4.4.3: Detalhe da malha de elementos finitos gerada a partir da malha de elementos de contorno.

Um recurso que tem sido amplamente adotado visando à melhoria do desempenho de cálculo para o MEC, é a aplicação de programação em ambientes distribuídos, que exploram a arquitetura de múltiplos processadores possibilitando a paralelização dos cálculos tanto das matrizes de influência $[H]$ e $[G]$ quanto a solução de sistemas lineares. Isto vai de encontro com o modo usual de computação numérica, onde instruções são executadas repetidas vezes em um único núcleo de processamento (CPU). No contexto da programação tradicional, não há necessidade de se preocupar com conflitos ou trocas de informação, ou em alguns casos até mesmo com a arquitetura

utilizada para o processamento dos dados. De forma a explorar esta nova arquitetura de processadores disponíveis atualmente, onde um microcomputador conta com mais de um processador, são propostas técnicas como openMP, onde instruções são executadas em um único processador mas divididas em suas trilhas (“*threads*”).

Uma alternativa atraente à aplicação de CPU’s e a técnica de openMP para cálculos numéricos é a utilização de placas gráficas para uso geral (“*General Purpose Graphic Processor Unit*” - GPGPU). Historicamente, estas placas têm sua origem na indústria de jogos para microcomputadores e graças a grande demanda por jogos em tempo real, com alta definição em 3D e iteratividade, as GPUs evoluíram para processadores “*manycore*”, multithread e altamente paralelizáveis. A Figura 4.4.4 mostra uma comparação entre a arquitetura de uma CPU e de uma GPU¹ do fabricante *NVidia*, onde é possível perceber que a GPU possui uma quantidade muito maior de unidades lógico aritméticas - ULA (“*Arithmetic Logic Units*” - *ALU*).

Figura 4.4.4: Comparação entre arquiteturas de uma CPU e de uma GPU [cud].

É interessante introduzir o conceito de *threads*, blocos e *grids*. Um *thread* é um dos muitos componentes responsáveis pela execução dos núcleos, funções que são executadas na GPU também conhecidas como “*kernels*”, atuando sobre um único conjunto de informação. Blocos são conjuntos de *threads* que compartilham a mesma **memória compartilhada** - “*shared memory*”, que não deve ser confundida com a memória RAM da placa. Os dados executados pelos *threads* em um único bloco podem, e em alguns casos devem, ser sincronizados em um dado momento do programa. Por fim, blocos de *threads* são agrupados em *grids*. A Figura 4.4.5 mostra a distribuição bidimensional de *threads* e blocos.

¹<http://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html>

Figura 4.4.5: Distribuição bidimensional de *threads* em blocos. [66]

O gráfico da Figura 4.4.6 mostra uma comparação entre GPUs e CPUs em termos de operações de ponto flutuante (*floating point operations* - *flops*).

Figura 4.4.6: Comparação entre CPUs e GPUs em termos de Giga FLOPS [cud].

Outro ponto importante a favor do uso de GPUs em computação de alto desempenho, é seu baixo custo e fácil acesso quando comparados a grandes servidores.

Existem atualmente algumas API's ("Application Programming Interface") que possibilitam programação em placas gráficas. São elas: CUDA (<https://developer.nvidia.com/cuda-gpus>), a primeira a ser desenvolvida, e OpenCL (<https://www.khronos.org/opencl/>). Estas API's são extensões das linguagens C/C++ e possuem sintaxes muito próximas sendo que a primeira se limita às placas da marca NVidia, enquanto a segunda é utilizada em qualquer arquitetura de placas gráficas. No entanto, a primeira já possui uma grande biblioteca de funções como para tratamento de matrizes esparsas (CuSparse), álgebra linear (CuBlas) e análise de Fourier (CuFFT), além de pacotes em softwares como MatLab e Mathematica. Embora possuam características próprias de linguagem de programação, podem ser inseridas em códigos escritos, por exemplo, em Fortran 90 e Python (PyCuda).

A API CUDA permite manipular facilmente a alocação e desalocação de memória na placa e também especificar os núcleos, programas que serão executados por cada *thread*. O código mostrado na Figura 4.4.7 ilustra um programa simples de soma de dois vetores implementado em CUDA.

```

1 #include <iostream>
2 using std::cout;
3
4
5 __global__ void sumVector(const int N, const float* v1, const float* v2,
6                           float* v)
7 {
8     int idx = blockIdx.x + blockDim.x*threadIdx.x;
9     while (idx < N){
10         v[idx] = v1[idx] + v2[idx];
11         idx += blockDim.x * gridDim.x;
12     }
13 }
14
15 int main ()
16 {
17     float *v1;
18     float *v2;
19     float *v1d;
20     float *v2d;
21     float *vd;
22     float *vh;
23     const int N = 10000;
24
25     //---
26     // Preenche vetores no host:
27     v1 = new float[N];
28     v2 = new float[N];
29     for (int ii = 0; ii < N; ii++) {
30         v1[ii] = static_cast<float>(ii);
31         v2[ii] = 4.0*static_cast<float>(ii);
32     }
33
34     // Aloca memória no device:
35     cudaMalloc((void**)& vd, sizeof(float)*N);
36     cudaMalloc((void**)& v1d, sizeof(float)*N);
37     cudaMalloc((void**)& v2d, sizeof(float)*N);
38
39     // Transfere os valores do host para device:
40     cudaMemcpy(v1d, v1, sizeof(float)*N, cudaMemcpyHostToDevice);
41     cudaMemcpy(v2d, v2, sizeof(float)*N, cudaMemcpyHostToDevice);
42
43     // Executa a soma:
44     sumVector<<<256, 256>>>>(N, v1d, v2d, vd);
45
46     vh = new float[N];
47     cudaMemcpy(vh, vd, sizeof(float)*N, cudaMemcpyDeviceToHost);
48
49     for (int ii = 0; ii < 10; ii++)
50         cout << v1[ii] << " + " << v2[ii] << " = " << vh[ii] << "\n";
51     return 0;
52 }

```

Figura 4.4.7: Exemplo de código em CUDA.

Sendo:

- C1: Declaração do núcleo que será executado em cada *thread*;
- C2: Preenche os vetores no *host device* (CPU);
- C3: Aloca memória no *global device* (GPU) e transfere dados da CPU para a GPU;
- C4: Define a quantidade de blocos e *threads* que executarão a função e invoca a função;

Na trecho C1, a função que será executada na GPU é identificada pela palavra chave `__global__` e todos os parâmetros são declarados como ponteiros; aqueles que são apenas parâmetros de entrada, possuem o identificador `const`; cada *thread* é identificada pela variável `idx` e `blockIdx` (*“block index”*), `blockDim` (*“block dimension”*), `threadIdx` (*“thread index”*) e `gridDim` (*“grid dimension”*) são variáveis *struct* internas da API e fornecem informações sobre a placa gráfica sendo utilizada.

Labaki et al. [45] apresentaram uma implementação de elementos de contorno utilizando funções de interpolação constantes para problemas potenciais e analisaram o desempenho em relação à mesma técnica implementada em CPU. O gráfico da Figura 4.4.8 mostra os resultados da análise.

Figura 4.4.8: Comparação entre CPU e GPU para implementação de elementos de contorno constantes [45].

É importante notar que há uma quantidade mínima de elementos para que o uso de GPUs seja realmente eficiente. Isto se deve à transmissão de dados entre a memória RAM da CPU (*“host device”*) e a memória da placa gráfica (*“global device”*). Sendo assim, os dados devem ser tratados o máximo possível na GPU de forma a evitar a troca constante de mensagens entre CPU e GPU e é exatamente neste ponto que a programação em placas gráficas se torna um processo desafiador: embora o poder de processamento de uma placa gráfica seja muito superior ao de uma CPU, sua quantidade de memória é muito limitada. Se considerada por exemplo, uma placa modelo GeForce GTx 690, que possui 3072 núcleos de processamento (*“CUDA core”*) sendo considerado de alto desempenho, tem apenas 4Gb de memória RAM², enquanto um microcomputador convencional pode ter facilmente mais de 12Gb. Uma possível alternativa, é criar um *cluster* de GPUs de forma que cada uma delas seja responsável por um bloco dos dados a serem tratados.

Nesta tese, o cálculo dos coeficientes das matrizes $[H]$ e $[G]$ foi realizado em GPU, enquanto a solução do sistema em CPU. Foi utilizado o ambiente de programação Nsight CUDA 5.5 e MatLab R2013b com o pacote de computação paralela (*“Parallel Computing Toolbox”*), uma placa gráfica GeForce GTx 690 instalada em um microcomputador com 32Gb de RAM e processador Intel Core i7-3770 3.40 GHz. As informações de geometria e características materiais são passadas para a placa gráfica, onde as matrizes $[H]$ e $[G]$ são calculadas em sua forma vetorial. Em seguida, estas

²Dividida em dois módulos de 2Gb, ou seja, não é possível utilizar os 4Gb de forma contígua.

matrizes são devolvidas à CPU e rearranjadas em sua forma matricial, para que as condições de contorno possam ser aplicadas e o sistema de equações resolvido.

No problema em questão, cada *thread* é responsável por calcular um elemento de cada uma das matrizes $[H]$ e $[G]$. As matrizes são armazenadas de forma vetorial, por linha. São utilizados seis pontos de Gauss para a integração numérica (Equações 4.3.37 e 4.3.38). Como dito anteriormente, a memória da placa gráfica é extremamente limitada, sendo assim, os vetores contendo os pontos e pesos para a integração de Gauss e algumas variáveis auxiliares são armazenados na memória compartilhada.

De forma a validar a metodologia proposta, o MEC foi comparado ao MEF, tanto em relação à precisão dos resultados, número de iterações e velocidade de processamento. Assim, o MEF utilizando elementos triangulares de primeira ordem também foi implementado em CUDA, segundo uma das abordagens propostas por Cecka et al. [16], facilmente adaptando o algoritmo serial da Figura 4.2.3 para uma implementação em paralelo, e desta forma, calculando apenas os elementos não nulos da matriz e realizando a montagem na própria placa gráfica. É neste ponto que o uso de elementos descontínuos apresenta grande vantagem em relação aos elementos contínuos, já que eles não passam pelo processo de montagem.

Capítulo 5

Análise Inversa

5.1 Introdução

Os métodos abordados no Capítulo 4, são utilizados em problemas em que se conhecem *a priori*, as constantes elásticas, a geometria e as condições de contorno do problema. A este tipo de estudo, dá-se o nome de “*direct problem analysis*”, ou análise direta de problemas. Este tipo de problema possui solução única, isto é, é um problema bem posicionado. No entanto, em imagens médicas não se conhece *a priori* algum dos parâmetros citados anteriormente, ou seja, deseja-se conhecer alguma condição de contorno, constante material, geometria ou até mesmo a Equação diferencial que modela o problema, constituindo o que se chama de “*inverse problem analysis*”, ou problemas de análise inversa.

Para problemas de análise inversa, os métodos do Capítulo 4 podem ser aplicados a fim de determinar os parâmetros desejados. Dado um conjunto de dados iniciais, eles são responsáveis por calcular as respostas de deslocamentos e forças de superfície, que por sua vez são comparados a valores conhecidos *a priori* ou valores desejados, em uma sequência iterativa, de forma a minimizar os erros obtidos.

Quando se trata de problemas da Física-Matemática, isto é, modelagem matemática de problemas físicos, é necessário saber se a resposta encontrada por algum método proposto é única. Para isto, as seguintes perguntas devem ser respondidas [37]:

- A solução existe? (Critério de existência da solução)
- A solução é única? (Critério de unicidade)
- A solução depende da continuidade da informação dada? (Condição de continuidade)

Até o presente momento, não há registro de que a análise inversa seja um problema bem posicionado. Os trabalhos encontrados na literatura assumem que sim, desta forma, será adotada a mesma premissa nesta tese.

O problema de análise inversa em elastografia é realizado em três etapas:

1. Identificação dos deslocamentos utilizando imagens simuladas de ultrassom ($\vec{u}_{presc}$ = deslocamentos prescritos);
2. Determinação de deslocamentos utilizando métodos numéricos ($\vec{u}_{calc}$), dada uma geometria, pressão aplicada e um valor inicial para os módulos de elasticidade;
3. Minimização do erro entre os deslocamentos calculados e os deslocamentos prescritos;

O diagrama da Figura 5.1.1 ilustra um processo de análise inversa para elastografia.

Figura 5.1.1: Diagrama para análise inversa.

Os critérios de parada adotados nesta tese são a convergência do campo de deslocamentos, a estacionariedade do vetor de módulos de elasticidade ou uma quantidade máxima de iterações. O vetor $\vec{u}_{presc}$ é determinado utilizando técnicas de fluxo óptico, tratado no Capítulo 7. A formulação para o problema inverso será tratada a seguir.

5.2 Formulação do Problema

Nesta tese, o objetivo da análise inversa é, dado um campo de deslocamentos $\vec{u}_{presc}$, pressão aplicada $\vec{p}$ e uma geometria $[x]$, determinar um vetor de módulos de elasticidade $\vec{E} \in \mathbb{R}^{N_R+}$ de tal forma que minimize a Equação 5.2.1

$$\min_{\vec{E} \in \mathbb{R}^{N_R+}} \left\| \vec{u}_{presc} - \vec{u}(\vec{E}) \right\| \quad (5.2.1)$$

Nesta tese, o coeficiente de Poisson será considerado constante para todos os materiais, $\nu = 0,490$, como mostra os resultados experimentais em Freutel et al. [25], Gennisson et al. [30],

Chai et al. [17]. Assim, o problema limita-se a encontrar os valores do conjunto de módulos de elasticidade de cada material presente na análise.

Na Equação 5.2.1, o vetor $\vec{u}(\vec{E})$ é o vetor de deslocamentos calculado utilizando o MEC ou MEF.

Definindo o erro quadrático médio, $\Phi(\vec{E})$, na Equação 5.2.2

$$\Phi(\vec{E}) = \frac{1}{2} \left\| \vec{u}_{presc} - \vec{u}(\vec{E}) \right\|^2 \quad (5.2.2)$$

o gradiente de $\Phi(\vec{E})$ em relação às variáveis de projeto $\vec{E}$ é dado pela seguinte Equação:

$$\Phi'(\vec{E}) = \frac{\partial \Phi(\vec{E})}{\partial \vec{E}} = \left[\frac{\partial \vec{u}(\vec{E})}{\partial \vec{E}} \right]^T (\vec{u}_{presc} - \vec{u}(\vec{E})) \quad (5.2.3)$$

O termo $\left[\frac{\partial \vec{u}(\vec{E})}{\partial \vec{E}} \right]$ é chamado de matriz jacobiana, e definido na Equação 5.2.4:

$$J_{ij}(\vec{E}) = \left[\frac{\partial \vec{u}(\vec{E})}{\partial \vec{E}} \right]_{ij} = \frac{\partial u_i(\vec{E})}{\partial E_j} \quad (5.2.4)$$

Expandindo a Equação 5.2.3 em uma série de Taylor,

$$\Phi'(\vec{E}) = \Phi'(\vec{E}_i) + \Phi''(\vec{E}_i) \vec{\Delta} E_i \quad (5.2.5)$$

O termo $\Phi''(\vec{E})$ é chamado de matriz hessiana e pode ser aproximado pela Equação 5.2.6, com o termo $\vec{\Delta} E_i$ dado pela Equação 5.2.7.

$$\Phi''(\vec{E}) \approx [J]^T [J] \quad (5.2.6)$$

$$\vec{\Delta} E_i = \vec{E} - \vec{E}_i \quad (5.2.7)$$

Assumindo que o resíduo $(\vec{u}_{presc} - \vec{u}(\vec{E})) = 0$, portanto $\Phi'(\vec{E}) = 0$, a Equação 5.3.1 resulta na Equação 5.2.8

$$\Phi''(\vec{E}_i) \vec{\Delta} E_i = \Phi'(\vec{E}_i) \quad (5.2.8)$$

Resolvendo 5.2.8 para $\vec{\Delta} E_i$,

$$\vec{\Delta} E_i = \Phi''(\vec{E}_i)^{-1} \Phi'(\vec{E}_i) \quad (5.2.9)$$

ou de forma explícita:

$$\vec{\Delta}E_i = [J^T J]^{-1} J^T \left(\vec{u}_{presc} - \vec{u}(\vec{E}) \right) \quad (5.2.10)$$

Por fim, o vetor de módulo de elasticidade é atualizado, como mostra a Equação 5.2.11

$$\vec{E}_{i+1} = \vec{E}_i + \vec{\Delta}E_i \quad (5.2.11)$$

5.3 Método de Variáveis Complexas

Esta seção apresenta a formulação apresentada por Marczak [51], Mundstock and Marczak [54], que utiliza variáveis complexas para cálculo de derivadas de funções analíticas, ao invés do uso de diferenças finitas ou pela diferenciação analítica. Esta técnica é importante pois através dela é possível calcular as derivadas mostradas na seção anterior de forma mais precisa.

Considerando uma função contínua e suave, $f(x) \in \mathbb{R}$, temos que sua expansão em série de Taylor é dada pela Equação 5.3.1:

$$f(x) = f(x_0) + \left. \frac{df}{dx} \right|_{x=x_0} (x - x_0) + \mathcal{O}(\Delta x^2) \quad (5.3.1)$$

Considerando que a variável $x \in \mathbb{C}$ e um parâmetro $\gamma \in \mathbb{R}$ [52]:

$$x = x_0 + j\gamma \quad (5.3.2)$$

$$j = \sqrt{-1} \quad (5.3.3)$$

Na Equação 5.3.1, desprezando os termos de ordem $\mathcal{O}(x^2)$ e substituindo a Equação 5.3.2 resulta em:

$$f(x_0 + j\gamma) = f(x_0) + \left. \frac{df}{dx} \right|_{x=x_0} (j\gamma) \quad (5.3.4)$$

Por fim, a primeira derivada da função $f(x)$ em $x = x_0$ é dada pela parte complexa $\Im\{f(x_0 + j\gamma)\}$, como mostrado a seguir:

$$\begin{aligned} j \left. \frac{df}{dx} \right|_{x=x_0} &= \frac{1}{\gamma} [f(x_0 + j\gamma) - f(x_0)] \\ \left. \frac{df}{dx} \right|_{x=x_0} &= \frac{1}{\gamma} \Im[f(x_0 + j\gamma)] \end{aligned} \quad (5.3.5)$$

Esta técnica tem apresentado bons resultados, principalmente em aplicações em conjunto com outros métodos numéricos e em especial, em problemas de otimização, como mostram os trabalhos

de Gao and He [28], Mundstock and Marczak [54], Marczak [51].

Para o caso de problemas multidimensionais, a derivada de uma função $f(\vec{x})$, $\vec{x} \in \mathbb{R}^N$ é calculada da seguinte forma:

$$\frac{\partial f(\vec{x})}{\partial x_i} = \frac{\Im \{f(x_i + jh, \vec{y})\}}{h} \quad (5.3.6)$$

$$\vec{y} = \{x_1, x_2, \dots, x_{i-1}, x_{i+1}, \dots, x_N\}^T \quad (5.3.7)$$

Como exemplo, a derivada da função $f(x_1, x_2, x_3) = e^{x_1 x_3} + \sin(x_1 x_2^2)$ em relação à variável x_1 no ponto $(x_1, x_2, x_3) = (1, 1, 1)$ é dada por

$$\frac{\partial f}{\partial x_1} = x_3 e^{x_1 x_3} + x_2^2 \cos(x_1 x_2^2) \quad (5.3.8)$$

$$\left. \frac{\partial f}{\partial x_1} \right|_{(1,1,1)} = 3,258584134327185 \quad (5.3.9)$$

É possível verificar a eficiência do MVC comparando-o com diferenças finitas (DF) e alterando o valor do parâmetro γ . A Tabela 5.1 mostra esta comparação.

γ	Erro MVC (%)	Erro DF (%)
10^{-2}	$0,001114 \times 10^0$	$0,289098 \times 10^0$
10^{-6}	$1,113431 \times 10^{-11}$	$2,877537 \times 10^{-5}$
10^{-8}	0,000000	$1,997302 \times 10^{-6}$
10^{-30}	0,000000	100,000000

Tabela 5.1: Comparação entre MVC e diferenças finitas.

Oberservando os resulatdos do exemplo, dois pontos importantes devem ser observados:

1. O passo $\Delta x = \gamma$ utilizando em diferenças finitas é da mesma ordem ou maior que o passo utilizado para o cálculo da derivada, enquanto o uso de variáveis complexas fornece um erro de ordem muito menor que o parâmetro γ utilizado, isto mostra uma convergência mais rápida para a solução exata quando utilizado MVC;
2. Para um parâmetro γ muito pequeno, o erro para diferenças finitas chega a 100%, isto graças ao limite de precisão da máquina.

É possível verificar a precisão do método no exemplo de forma analítica, como segue:

$$\begin{aligned}
f(x_1 + j\gamma, x_2, x_3) &= e^{(x_1 + j\gamma)x_3} \sin[(x_1 + j\gamma)x_2^2] \\
&= e^{x_1 x_3} \cos(\gamma x_3) + \cosh(\gamma x_2^2) \sin(x_1 x_2^2) + \\
&\quad j \left\{ \gamma [x_3 e^{x_1 x_3} + x_2^2 \cos(x_1 x_2^2)] \right. \\
&\quad \left. - \frac{1}{6} \gamma^3 [x_3^3 e^{x_1 x_3} - x_2^6 \cos(x_1 x_2^2)] + \mathcal{O}(\gamma^5) \right\} \quad (5.3.10)
\end{aligned}$$

$$\frac{\Im[f(x_1 + j\gamma, x_2, x_3)]}{\gamma} = \underbrace{x_3 e^{x_1 x_3} + x_2^2 \cos(x_1 x_2^2)}_{T1} + \underbrace{\mathcal{O}(\gamma^2)}_{T2} \quad (5.3.11)$$

Observando que termo $T1$ da Equação 5.3.11 é a solução analítica da derivada da função $f(x_1, x_2, x_3)$ em relação a x_1 (Equação 5.3.8), se o valor de γ for suficientemente pequeno, é possível aproximar para zero os termos de ordem $\mathcal{O}(\gamma^2)$.

Além das vantagens numéricas, está também a simplicidade para programar o método. A restrição do método é a necessidade de uma função analítica.

Esta técnica para cálculo de derivadas com variável complexa foi utilizada para o cálculo das matrizes da Equação 5.2.4.

5.4 Validação da Análise Inversa

Para a validação da análise inversa, é proposto um estudo em duas etapas:

1. Simulações ideais (Figura 5.4.3);
2. Simulações com adição de ruído (Figuras 5.4.4a e 5.4.4b);

Foram criados modelos para elementos de contorno e elementos finitos com uma, duas e três lesões. A modelagem do problema é mostrada na Figura 5.4.1 para um modelo com uma lesão. As dimensões e condições de contorno são as mesmas sugeridas por Pan et al. [59].

Figura 5.4.1: Modelagem do problema com phantom.

A Tabela 5.2 mostra os parâmetros utilizados para a geração dos dados, onde p_0 é a pressão aplicada, E_{back} se refere ao módulo de elasticidade do background e $E_{lesão}$ ao módulo de elasticidade da lesão.

Tabela 5.2: Parâmetros utilizados para simulação do phantom com duas lesões.

Parâmtro	Valor
p_0	$10KPa$
E_{back}	$200KPa$
$E_{lesão}$	$1000KPa$

A Figura 5.4.2 mostra o resultado da convergência de cada método, elementos de contorno e elementos finitos, em função da quantidade de iterações.

Figura 5.4.2: Gráfico de convergência para análise inversa utilizando uma simulação ideal, sem ruído.

É possível observar que o MEC e o MEF apresentam a mesma taxa de convergência para modelos sem ruído, com até dezesseis iterações para convergência considerando como critério de parada $\|\Delta \vec{E}\| \leq 10^{-6}$ (Equação 5.2.10).

A modelagem para a etapa 2, com adição de ruído, é realizada tendo como referência o trabalho de Zhu et al. [73], que utiliza ruído branco gaussiano de duas formas distintas, uma somando-o ao vetor de pressão (Figura 5.4.4a) e numa segunda etapa, adicionando ruído diretamente ao vetor de deslocamentos (Figura 5.4.4b). A modelagem do ruído presente no vetor de deslocamentos entre duas imagens de ultrassom é apresentada por Bilgen and Insana [9]. Os valores obtidos para módulo de elasticidade foram calculados utilizando uma média de dez simulações.

Figura 5.4.3: Diagrama validação da análise inversa utilizando entrada ideal (sem ruído).

Figura 5.4.4: Diagrama validação da análise inversa utilizando entrada com ruído: (a) Ruído adicionado ao vetor de pressão; (b) Ruído adicionado ao vetor de deslocamentos;

Os modelos utilizados na primeira etapa também foram utilizados para as simulações com adição ruído gaussiano branco. Foram utilizados ruídos com relação sinal ruído (SNR_{dB}) de 30, 50 e 100dB somados ao vetor de pressão (Figuras 5.4.5, 5.4.6 e 5.4.7) e ao vetor de deslocamentos (Figuras 5.4.8, 5.4.9 e 5.4.10). Os módulos de elasticidade foram calculados pela média entre dez simulações.

Figura 5.4.5: Curva de convergência para simulações com adição de ruído branco gaussiano ao vetor de pressão ($SNR_{dB} = 30dB$).

Figura 5.4.6: Curva de convergência para simulações com adição de ruído branco gaussiano ao vetor de pressão ($SNR_{dB} = 50dB$).

Figura 5.4.7: Curva de convergência para simulações com adição de ruído branco gaussiano ao vetor de pressão ($SNR_{dB} = 100dB$).

Figura 5.4.8: Curva de convergência para simulações com adição de ruído branco gaussiano ao vetor de deslocamento ($SNR_{dB} = 30dB$).

Figura 5.4.9: Curva de convergência para simulações com adição de ruído branco gaussiano ao vetor de deslocamento ($SNR_{dB} = 50dB$).

Figura 5.4.10: Curva de convergência para simulações com adição de ruído branco gaussiano ao vetor de deslocamento ($SNR_{dB} = 100dB$).

Capítulo 6

Criação de Phantoms Matemáticos

6.1 Introdução

Para que qualquer método proposto para análise de fenômenos biológicos possa ser testado e validado, é necessário que haja corpos de prova, isto é, modelos físicos ou conceituais que possam simular o comportamento mecânico dos tecidos nos quais serão aplicadas as técnicas apresentadas.

Como o foco principal deste trabalho é a elastografia em imagens de ultrassom, é necessário um modelo matemático que seja capaz de simular imagens que contenham características intrínsecas à técnica de imageamento utilizado, como o ruído *speckle*, que surge graças a presença de espalhadores no meio no qual a onda acústica se propaga, fazendo com que ocorra o fenômeno de espalhamento e interferência da onda emitida pelo probe de ultrassom.

A imagem de ultrassom é formada pela captação de um sinal de rádio frequência, detectado por um cristal piezoelétrico (PZT). A envoltória do sinal é calculada e resulta em uma linha da imagem, também conhecida como modo A (amplitude). Desta forma, para que uma imagem seja criada, são necessárias várias linhas dispostas paralelamente.

A imagem é gerada graças à diferença das impedâncias acústicas entre os tecidos presentes no corpo. Isto causa efeitos de reflexão e atenuação gerando o contraste necessário para formação da imagem.

Para o estudo proposto nesta tese, é necessário criar phantoms matemáticos que simulem imagens com adição de ruído *speckle*. Na seção seguinte, será apresentado o software USSim desenvolvido por Cardoso et al. [14] em MatLab, que permite simular imagens de phantoms transcutâneos. Este software utiliza o Método dos Elementos Finitos, os *toolboxes* FieldII¹ [35] e iso2mesh [24] e uma técnica de morfismo para a criação de imagens em modo B pré e pós aplicação de pressão. É apresentada uma metodologia para criação de phantoms de imagens de ultrassom em modo B.

A seguir, serão detalhadas as técnicas de morfismo aliado ao MEF e o *software* USSim.

¹http://field-ii.dk/?./downloading_8_20.html

6.1.1 Elementos Finitos e Morfismo

Considerando uma imagem de ultrassom I_1 e um conjunto de coordenadas para os espalhadores $\chi_1 = \{\vec{p}_1, \dots, \vec{p}_{N_E}\}$ com cardinalidade $|\chi_1| = N_E$, aleatoriamente posicionados sobre a imagem, como mostrado na 5.4.1 Figura 6.1.1.

Figura 6.1.1: Imagem de ultrassom e um conjunto χ_1 de espalhadores.

Sobre esta imagem é criada uma malha de elementos finitos e, dadas as propriedades materiais e condições de contorno de deslocamento e força, uma simulação é realizada. O resultado é uma malha deformada, como mostra a Figura 6.1.2

Figura 6.1.2: Malha de elementos finitos deformada e deslocamento dos espalhadores.

É possível determinar os deslocamentos dos pontos do conjunto χ_1 utilizando “*matching*”. Em uma malha indeformada, dadas as distâncias entre cada espalhador e o elemento que o contém, como mostra a Figura 6.1.3

Figura 6.1.3: Elemento finito deformado e distâncias entre o espalhador e os vértices do elemento.

cada $\vec{d}_i$ é calculado como

$$\underbrace{\begin{Bmatrix} x_1^i \\ x_2^i \\ 1 \end{Bmatrix}}_{\vec{p}_i} = \underbrace{\begin{bmatrix} x_1^1 & x_1^2 & x_1^3 \\ x_2^1 & x_2^2 & x_2^3 \\ 1 & 1 & 1 \end{bmatrix}}_{[P_e]} \underbrace{\begin{Bmatrix} d_1^i \\ d_2^i \\ d_3^i \end{Bmatrix}}_{\vec{d}^i} \quad (6.1.1)$$

A última linha do sistema de Equações de 6.1.1 implica que a soma das distâncias dentro de um elemento deve ser sempre igual a 1, o que garante que o ponto esteja sempre no domínio do elemento. De posse do vetor $\vec{d}$ para cada ponto do conjunto χ_1 , é possível determinar um novo conjunto de pontos χ_2 como na Equação 6.1.2

$$\vec{p}_i' = [P_e'] \vec{d}^i \quad (6.1.2)$$

sendo $\vec{p}_i'$ as novas coordenadas de χ_2 , $[P_e']$ são as coordenadas dos vértices de cada elemento da malha já deformada e $\vec{d}$ as distâncias calculadas pela Equação 6.1.1.

6.1.2 O *software* USSim

O software consiste em uma interface gráfica, onde é possível escolher as características geométricas e materiais, parâmetros da simulação e da imagem de modo B. Cada uma destas etapas é descrita a seguir.

Características geométricas e materiais:

- Geometria da lesão;
- Módulos de elasticidade de cada tecido;
- Coeficiente de Poisson para os tecidos;

- Velocidade do som no tecido;
- Amplitude de cada espalhador;
- Quantidade de espalhadores;
- Atenuação em cada tecido.

Parâmetros da simulação:

- Quantidade de forças aplicadas;
- Valor das forças aplicadas (Kgf);
- Resolução do phantom ($pixel/\mu m$);
- Posição do probe;
- Dimensão do probe;
- Condições de contorno de deslocamento;
- Direção da aplicação das forças;

Parâmetros para a formação da imagem em modo B:

- Quantidade de elementos (transdutores);
- Largura de cada elemento;
- Altura de cada elemento;
- Distância entre elementos adjacentes;
- Distância entre os elementos e o tecido;
- Quantidade de elementos ativos na emissão;
- Quantidade de elementos ativos na recepção;
- Quantidade de feixes de ultrassom;
- Frequência central (MHz);
- Frequência de amostragem (MHz);
- Largura da imagem (mm);

O *software* realiza então, N sequências de simulações utilizando elementos finitos tetraédricos de primeira ordem para cada valor de força definido na etapa 2. Calculados os deslocamentos de cada nó na malha de elementos finitos, a metodologia apresentada na subseção anterior é aplicada ao conjunto de espalhadores, resultando em um conjunto χ , composto de N subconjuntos de espalhadores χ_2^i , $\chi = \{\chi_2^1, \chi_2^2, \dots, \chi_2^N\}$.

Estas análises são dados de entrada para a simulação com o *toolbox* Field II. Por fim, imagens de ultrassom bidimensionais são geradas e salvas em um arquivo *.mat*.

Para esta tese, foram utilizados dois tipos de phantoms com lesão circular, centrada na imagem e raios $r = 3,03mm$ (phantom 1) e $r = 5,5mm$ (phantom 2). Os parâmetros adotados em cada etapa são listados a seguir:

Etapa 1 - Características geométricas:

- Lesão circular, centrada na imagem;
- Módulos de elasticidade:
 - phantom 1:
 - * $E_{background} = 200KPa$, $E_{lesão} = 1MPa$;
 - phantom 2:
 - * $E_{background} = 200KPa$, $E_{lesão} = 1,8MPa$;

- Coeficiente de Poisson: $\nu = 0,49$;
- Velocidade do som: $c = 1540m/s$;
- Amplitude dos espalhadores:
 - $A_{background} = 1$, $A_{tecido} = 1,7$ (lesão hipereecólica);
- Quantidade de espalhadores: $N_{esp} = 1mm^{-3}$;
- Atenuação em cada tecido: $\alpha = 0dB$;

Parâmetros de simulação:

- Quantidade de forças aplicadas: $N_{sim} = 5$
- Valor das forças aplicadas:
 - $F = \{0; 0,2; 0,4; 0,6; 0,8; 1,0\} [Kgf]$;
- Resolução do phantom: $500pixels/\mu m$;

- Posição do probe: $(x_1, x_2) = (100, 70) pixels$;
- Dimensão do probe: $200 \times 140 pixels$;
- Condições de contorno de deslocamento: base localizada em $x_1 = 0$ completamente fixada, isto é, deslocamentos nulos em todas as direções e as laterais do phantom são livres;
- Direção da força aplicada: $\vec{f} = -F_i \vec{x}_2$ sobre o plano $x_1 = 200 pixels$;

Parâmetros para a formação da imagem em modo B:

- Quantidade de elementos: 192
- Largura de cada elemento: $440 mm$;
- Altura de cada elemento: $5000 \mu m$;
- Distância entre elementos adjacentes do transdutor, igualmente espaçados: $50 \mu m$;
- Distância entre os transdutores e o tecido: $30 \mu m$;
- Quantidade de elementos ativos na transmissão: 1;
- Quantidade de elementos ativos na recepção: 32;
- Quantidade de feixes de ultrassom emitidos: 75;
- Frequência central: $f_c = 3,5 MHz$;
- Frequência de amostragem: $f_A = 100 MHz$;
- Largura da imagem: $L = 100 mm$;

Os resultados para o phantom 2 em pré e pós deformação com $\vec{f} = -1.0 \vec{x}_2 Kgf$ são mostrados nas Figuras 6.1.4a e 6.1.4b.

Figura 6.1.4: phantom matemático $r = 5.5mm$: (a) Imagem pré-deformação; (b) Imagem pós-deformação.

Capítulo 7

Fluxo Óptico para Rastreamento de Ruído Speckle

7.1 Introdução

Para que seja possível determinar as constantes materiais utilizando imagens de ultrassom, é necessário o uso de técnicas que possam fazer a correspondência entre pixels entre duas imagens, isto é, determinar os deslocamentos que ocorrem em função da aplicação de um campo de pressões. Este procedimento é chamado de rastreamento de ruído speckle, pois utiliza o ruído speckle como marcador entre duas imagens de uma mesma seção.

As técnicas mais empregadas para rastreamento de ruído speckle podem ser divididas em duas categorias:

1. *Block Matching*;
2. Fluxo Óptico.

A primeira categoria consiste em encontrar parâmetros de uma dada transformação afim (deslocamentos axiais e laterais, ângulo de rotação e ângulo de cisalhamento) que melhor aproxime um dado bloco de uma imagem em um tempo t_0 de uma segunda imagem no tempo t_1 . Podem ser citados os trabalhos de DiBattista and Noble [20], Liu et al. [46], Pan et al. [59].

Por outro lado, a abordagem utilizando fluxo óptico, visa encontrar diretamente os deslocamentos entre pontos de uma mesma imagem. Este método tem sua aplicação principalmente em problemas de visão computacional e se baseia no fato de que a intensidade de um pixel não apresenta variação entre dois instantes de tempo.

Para esta tese, foram escolhidos os métodos baseados em fluxo óptico, descritos nas seções 7.2 e 7.3, que fazem uso de algoritmos de fluxo óptico de Lucas-Kanade e de Brox. A característica mais atraente para o uso de fluxo óptico para este trabalho é a solução pixel a pixel, isto é, as

técnicas de fluxo óptico permitem determinar os deslocamentos de cada pixel, permitindo o cálculo em pontos específicos nas malhas de elementos finitos e de contorno.

O algoritmo de Lucas-Kanade foi escolhido pois a literatura apresenta bons resultados em suas aplicações em imagens de ultrassom, enquanto o de Brox proporciona uma abordagem multiescala, que permite determinar grandes deslocamentos entre imagens.

7.2 Fluxo Óptico por Lucas-Kanade

O método de Lucas-Kanade [50] se baseia no princípio de intensidade constante entre duas imagens em tempos distintos, isto é, dada uma sequência de duas imagens I_1 e I_2 de uma mesma seção transversal, adquiridas em tempos t_1 e $t_2 = t_1 + \delta t$, respectivamente,

$$I_1(x_1, x_2, t_1) = I_2(x_1 + u_1, x_2 + u_2, t_1 + \delta t) \quad (7.2.1)$$

Sendo que u_1 e u_2 são os deslocamentos horizontais e verticais na imagem. A Equação 7.2.1 diz que, dado um pixel de coordenadas (x_1, x_2) e intensidade $I_1(x_1, x_2)$, é possível localizá-lo na imagem I_2 dados os deslocamentos (u_1, u_2) . É possível relacionar os deslocamentos entre as duas imagens aplicando uma “derivada material”, isto é, estabelecer uma relação diferencial entre os deslocamentos e a variação de intensidade da imagem:

$$\frac{dI}{dt} = \frac{\partial I}{\partial x_1} \frac{\partial x_1}{\partial t} + \frac{\partial I}{\partial x_2} \frac{\partial x_2}{\partial t} + \frac{\partial I}{\partial t} \quad (7.2.2)$$

Na Equação 7.2.2, a derivada temporal parcial de I representa a variação da intensidade em um ponto fixo da imagem. Considerando variação nula na intensidade de um pixel (Equação 7.2.1), o termo dI/dt na Equação 7.2.2 é zero, ficando

$$\frac{\partial I}{\partial x_1} \frac{\partial x_1}{\partial t} + \frac{\partial I}{\partial x_2} \frac{\partial x_2}{\partial t} = -\frac{\partial I}{\partial t} \quad (7.2.3)$$

Considerando pequenas variações da imagem em relação ao tempo, a derivada $\partial I/\partial t$ pode ser escrita como:

$$\frac{\partial I}{\partial t} = \frac{\Delta I}{\Delta t} \quad (7.2.4)$$

e desta forma, multiplicando a Equação 7.2.3 por Δt , tem-se

$$\frac{\partial I}{\partial x_1} \underbrace{\Delta x_1}_{u_1} + \frac{\partial I}{\partial x_2} \underbrace{\Delta x_2}_{u_2} = -\Delta I \quad (7.2.5)$$

O método de Lucas-Kanade implica que os deslocamentos são calculados pixel a pixel na imagem

e a Equação 7.2.5 é calculada em uma janela de tamanho $N_{w_1} \times N_{w_2}$.

Apostolakis et al. [2] utilizam uma janela retangular de $40 \times 20 \text{pixel}$, no entanto, após várias simulações com vários valores para $N_{w_1} \times N_{w_2}$, concluiu-se que $N_{w_1} = N_{w_2} = 60 \text{pixel}$ apresenta um campo de deslocamentos mais suave, como pode ser observado nas Figuras 7.2.1, 7.2.2a, 7.2.2b e 7.2.2c para os phantoms da Figura 7.2.1, que mostra os phantoms pré e pós deformação, sendo que a imagem I_2 foi obtida pela aplicação de uma força de $0.4Kgf$.

Figura 7.2.1: Phantoms transcutâneos com lesão, sendo a figura à esquerda a imagem pré-deformação e a imagem a direita pós-deformação.

(a)

(b)

(c)

Figura 7.2.2: Simulações com Lucas-Kanade variando o tamanho da janela de amostragem: (a) Janela $20 \times 20pixels$; (b) Janela $40 \times 40pixels$; (c) Janela $60 \times 60pixels$;

As derivadas espaciais são calculadas da seguinte forma:

$$\frac{\partial I}{\partial x_1} = \frac{1}{2} \left(\frac{\partial I_1}{\partial x_1} + \frac{\partial I_2}{\partial x_1} \right) \quad (7.2.6)$$

$$\frac{\partial I}{\partial x_2} = \frac{1}{2} \left(\frac{\partial I_1}{\partial x_2} + \frac{\partial I_2}{\partial x_2} \right) \quad (7.2.7)$$

E a derivada temporal:

$$\Delta I = I_2 - I_1 \quad (7.2.8)$$

As derivadas da Equação 7.2.3 resultam em matrizes de dimensão 60×60 , que reorganizadas em forma vetorial, resultam no sistema de Equações 7.2.9:

$$\left[\frac{\partial I}{\partial x_1} \right]_{60 \times 60} u_1 + \left[\frac{\partial I}{\partial x_2} \right]_{60 \times 60} u_2 = - \left[\frac{\partial I}{\partial t} \right]_{60 \times 60} \quad (7.2.9)$$

A Equação 7.2.9 é resolvida aplicando mínimos quadrados e as derivadas espaciais são calculadas utilizando filtro gaussiano.

7.3 Fluxo Óptico por Brox

O método proposto por Brox et al. [12], Brox and Malik [11] assume as seguintes premissas:

- Gradiente constante: O critério de intensidade constante é muito susceptível a pequenas variações de intensidade, e assim, é necessário algum critério que possibilite pequenas variações de intensidade entre as imagens, e seja invariante em relação a alteração dos valores de cinza. Este critério é o gradiente da escala de cinza da imagem:

$$\nabla I(x_1, x_2, t) = \nabla I(x_1 + du_1, x_2 + du_2, t + \delta t) \quad (7.3.1)$$

- Suavidade: Este critério assume que o campo de deslocamentos é suave por toda imagem;
- Abordagem multiescala: Em certos problemas, é interessante uma abordagem que permita tratar grandes deslocamentos. Isto elimina a restrição imposta pela aproximação da série de Taylor, que ignora termos de ordem $\mathcal{O}(n^2)$ em algoritmos de FO como os Horn and Schunck [34]. Assim, é utilizado um esquema de *downsampling* descrito por Burt and Adelson [13]. Os deslocamentos são calculados de forma iterativa para cada imagem, sendo que o deslocamento inicial para a imagem de nível l , é obtido no nível $l - 1$.

O método de Brox envolve a minimização de um funcional combinando os critérios de gradiente e intensidade invariáveis, como mostra a Equação 7.3.2

$$\Theta(\vec{u}) = \int_{\Omega} \Psi(|I_2 - I_1|^2 + \gamma |\nabla I_1 - \nabla I_2|^2) d\Omega + \alpha \int_{\Omega} \Psi(|\nabla_3 u_1|^2 + |\nabla_3 u_2|^2) d\Omega \quad (7.3.2)$$

A função $\Psi(a^2) = \sqrt{a^2 + 0,001^2}$, a constante α é um parâmetro de regularização com $15 \leq \alpha \leq 30$ e γ determina a importância do gradiente sobre a variação de textura, sendo valores típicos $20 \leq \gamma \leq 100$; ∇_3 é o gradiente espaço-temporal, isto é, o gradiente nas duas dimensões espaciais e a derivada temporal dos deslocamentos. O funcional da Equação 7.3.2 é minimizado utilizando Equações de Euler-Lagrange e assim, os valores de $\vec{u}$ para imagens com deslocamentos arbitrários são determinados.

7.4 Validação dos Métodos para rastreamento de ruído speckle

A metodologia proposta para a validação dos métodos de fluxo óptico aplicados à imagens de ultrassom é, aplicar os algoritmos discutidos acima e comparar os resultados com simulações por elementos finitos em pontos específicos da imagem.

Serão consideradas como medidas de erro a média da diferença absoluta $\bar{\epsilon}$ (Equação 7.4.2) e seu desvio padrão $\kappa(\epsilon)$, o erro quadrático médio ϵ_{RMS} (Equação 7.4.4):

$$\epsilon(x_i, y_i) = \|\vec{u}_{FEM}^i - \vec{u}_{FO}^i\| \quad (7.4.1)$$

$$\bar{\epsilon} = \frac{1}{N} \sum_{i=1}^N \epsilon(x_i, y_i) \quad (7.4.2)$$

$$\bar{\sigma}(\epsilon) = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (\epsilon(x_i, y_i) - \bar{\epsilon})^2} \quad (7.4.3)$$

$$\epsilon_{RMS} = \sqrt{\frac{1}{N} \sum_{i=1}^N \|\vec{u}_{FEM}^i - \vec{u}_{FO}^i\|^2} \quad (7.4.4)$$

Foram realizadas diversas simulações utilizando os phantoms descritos no Capítulo 6. Para ilustrar o desempenho de cada método, foi escolhido o phantom 1, com uma força aplicada de $0,4Kgf$. A análise de erro será considerada para cada algoritmo de fluxo óptico separadamente, considerando a aplicação de um filtro gaussiano com desvio padrão $\bar{\sigma} = 1$ e janela de 60×60 pixels.

7.4.1 Análise de Erro para o Algoritmo de Lucas-Kanade

A Figura 7.4.1 ilustra um phantom com uma lesão, sobreposto com a malha de elementos finitos e os deslocamentos calculados por elementos finitos (indicados por setas de cor azul) e fluxo óptico (indicados por setas de cor vermelha). O phantom pós-deformação foi obtido pela aplicação de uma pressão equivalente a uma carga de $0,4Kgf$.

Comparison between FEM and Lucas-Kanade

Figura 7.4.1: phantom com malha de elementos finitos e deslocamentos calculados por elementos finitos (marrom) e fluxo óptico (vermelho) utilizando o algoritmo de Lucas-Kanade.

A Figura 7.4.2 mostra em detalhe a região da lesão para uma melhor comparação qualitativa do método de Lucas-Kanade.

Comparison between FEM and Lucas-Kanade

Figura 7.4.2: Detalhe para comparação qualitativa do algoritmo de fluxo óptico por Lucas-Kanade e elementos finitos.

É possível observar que, qualitativamente, ou seja, visualmente, a técnica de fluxo óptico por Lucas-Kanade fornece resultados muito próximos dos simulados por elementos finitos. Resta saber se numericamente os resultados continuam válidos. Para isto, são aplicadas as métricas dadas pelas Equações 7.4.2, 7.4.4 e 7.4.3. A Tabela 7.1 mostra os valores de cada medida para o phantom 1 (ver subseção 6.1.2), considerando a sequência de imagens obtidas pelas simulações do Capítulo 6.

Tabela 7.1: Medidas de erro para o phantom 1.

$\vec{p}[KPa]$	$(\bar{\epsilon} + \sigma) \times 10^{-7} [m]$	$\epsilon_{RMS} \times 10^{-7} [m]$
2,8	$0,471 \pm 0,247$	0,531
4,3	$1,004 \pm 0,539$	1,140
5,8	$1,628 \pm 0.903$	1,862

7.4.2 Análise de Erro para o Algoritmo de Brox

A mesma metodologia foi aplicada para o algoritmo de Brox. A Figura 7.4.3 mostra a a malha de elementos finitos sobreposta a a imagem de ultrassom e os campos de deslocamento calculados por elementos finitos (indicados por setas de cor azul) e fluxo óptico por Brox (indicados por setas de cor vermelha) e o detalhe da lesão, mostrado na Figura 7.4.4.

Comparison between FEM and Brox

Figura 7.4.3: phantom com malha de elementos finitos e deslocamentos calculados por elementos finitos (marrom) e fluxo óptico (vermelho) utilizando o algoritmo de Brox.

Comparison between FEM and Brox

Figura 7.4.4: Detalhe para comparação qualitativa do algoritmo de fluxo óptico por Brox e elementos finitos.

Novamente, o algoritmo de rastreamento de ruído speckle é, qualitativamente, muito próximo das simulações por elementos finitos. A Tabela 7.2 mostra os resultados da aplicação das métricas

Tabela 7.2: Medidas de erro para o phantom 1.

$\vec{p}[KPa]$	$(\bar{\epsilon} + \sigma) \times 10^{-4} [m]$	$\epsilon_{RMS} \times 10^{-4} [m]$
2,8	$0,400 \pm 0,210$	0,452
4,3	$0,726 \pm 0,363$	0,812
5,8	$0,956 \pm 0,446$	1,055

Capítulo 8

Resultados

Neste capítulo serão mostrados os resultados da integração entre o módulo de fluxo óptico para cálculo de deslocamentos e os métodos numéricos, elementos de contorno e elemento finitos. Para a validação são propostos dois estudos utilizando phantoms matemáticos, seguindo o modelo geométrico da Figura 5.4.1. O módulo de elasticidade do background é considerado o mesmo para as duas simulações ($E_{back} = 200KPa$), enquanto o módulo de elasticidade e o raio para a lesão são considerados $E_{lesão} = 1MPa$, $r = 3,03mm$ e $E_{lesão} = 1,8MPa$ e $r = 5,5mm$ para os phantoms 1 e 2 respectivamente.

As simulações para phantoms foram realizadas com seis valores de pressão distintos, sendo $p_0 = 0Kgf$ a imagem de referência. O coeficiente de Poisson foi mantido constante para todos os materiais, $\nu = 0,49$, como aponta a literatura [18, 30, 33].

Para o phantom 2 foram realizados testes de convergência de erro em função da quantidade de elementos nas malhas e também uma comparação do número de iterações e tempo de processamento.

8.1 Phantom 1

As características físicas e geométricas são dadas pela Tabela 8.1:

Table 8.1: Propriedades físicas e materiais para o phantom 1.

Propriedade	Valor
$E_{background}$	$200KPa$
$E_{lesão}$	$1MPa$
ν	$0,490$
$r_{lesão}$	$3,03mm$

Os valores estimados para o módulo de elasticidade de cada simulação são detalhados nas Tabelas 8.2, 8.3, 8.4 e 8.5.

Table 8.2: Resultados para os módulos de elasticidade estimados utilizando MEC e Lucas-Kanade para o phantom 1.

$p [KPa]$	$E_{background} [KPa]$	$E_{lesion} [KPa]$	$Tempo [s]$
1, 4	222, 61	1, 20	41, 87
2, 8	223, 97	1, 01	46, 68
4, 3	221, 95	1, 17	44, 14
5, 8	223, 70	0, 98	53, 25
7, 3	225, 09	0, 91	45, 45

Table 8.3: Resultados para os módulos de elasticidade estimados utilizando MEC e Brox para o phantom 1.

$p [KPa]$	$E_{background} [KPa]$	$E_{lesão} [MPa]$	$Tempo [s]$
1, 4	218, 95	1, 68	262, 44
2, 8	220, 04	1, 23	262, 82
4, 3	218, 98	1, 32	258, 19
5, 8	219, 16	1, 34	261, 54
7, 3	219, 68	1, 25	257, 88

Table 8.4: Resultados para os módulos de elasticidade estimados utilizando MEF e Lucas-Kanade para o phantom 1.

$p [KPa]$	$E_{background} [KPa]$	$E_{lesão} [MPa]$	$Tempo [s]$
1, 4	206, 76	1, 07	105, 75
2, 8	205, 64	1, 14	109, 76
4, 3	204, 89	1, 24	111, 63
5, 8	206, 56	1, 01	106, 80
7, 3	207, 52	0, 97	106, 56

Table 8.5: Resultados para os módulos de elasticidade estimados utilizando MEF e Brox para o phantom 1.

$p [KPa]$	$E_{background} [KPa]$	$E_{lesão} [KPa]$	$Tempo [s]$
1,4	204,21	1,32	318,54
2,8	204,03	1,28	320,82
4,3	203,42	1,37	320,54
5,8	203,55	1,34	321,56
7,3	203,92	1,28	324,00

Os valores médios e os desvios padrão de erro para cada simulação, são dados pela Tabela 8.6.

Table 8.6: Erro médio e desvio padrão para cada associação de métodos para o phantom 1.

Método	$\bar{\epsilon}_{background} \pm \bar{\sigma}$	$\bar{\epsilon}_{lesão} \pm \bar{\sigma}$
	[%]	[%]
MEC + Lucas-Kanade	$11,73 \pm 0,61$	$9,79 \pm 8,58$
MEC + Brox	$9,68 \pm 0,23$	$36,39 \pm 18,25$
MEF + Lucas-Kanade	$3,13 \pm 0,51$	$9,80 \pm 9,36$
MEF + Brox	$1,91 \pm 0,16$	$31,80 \pm 3,89$

Os gráficos das Figuras 8.1.1 e 8.1.2 ilustram o erro em função da pressão aplicada para MEC e MEF, considerando uma malha de elementos de contorno e uma malha de elementos finitos com 60 e 324 elementos, respectivamente.

Figura 8.1.1: Erro ϵ (%) para o background em função da pressão aplicada.

Figura 8.1.2: Erro ϵ (%) para a lesão em função da pressão aplicada.

8.2 Phantom 2

As características físicas e geométricas para o phantom 2 são dadas pela Tabela 8.7:

Table 8.7: Material and geometric properties for phantom 2.

Property	Value
$E_{background}$	$200 KPa$
E_{lesion}	$1,8 MPa$
ν	0,490
r_{lesion}	$5,5 mm$

Os valores estimados para o módulo de elasticidade de cada simulação são detalhados nas Tabelas 8.8, 8.9, 8.10 e 8.11.

Table 8.8: Resultados para os módulos de elasticidade estimados utilizando MEC e Lucas-Kanade para o phantom2.

$p [KPa]$	$E_{background} [KPa]$	$E_{lesion} [MPa]$	$Tempo [s]$
1, 4	229, 97	1, 75	32, 44
2, 8	229, 10	1, 63	41, 95
4, 3	225, 19	2, 36	38, 93
5, 8	229, 67	1, 69	36, 59
7, 3	229, 45	1, 90	42, 10

Table 8.9: Resultados para os módulos de elasticidade estimados utilizando MEC e Brox para o phantom2.

$p [KPa]$	$E_{background} [KPa]$	$E_{lesão} [MPa]$	$Tempo [s]$
1, 4	224, 21	2, 36	224, 77
2, 8	225, 17	1, 96	240, 86
4, 3	224, 14	2, 40	234, 75
5, 8	227, 14	1, 98	235, 29
7, 3	225, 61	2, 15	244, 75

Table 8.10: Resultados para os módulos de elasticidade estimados utilizando MEF e Lucas-Kanade para o phantom2.

$p [KPa]$	$E_{background} [KPa]$	$E_{lesion} [MPa]$	$Tempo [s]$
1, 4	217, 46	1, 25	69, 86
2, 8	217, 19	1, 08	68, 21
4, 3	212, 90	2, 45	69, 43
5, 8	214, 84	1, 23	73, 57
7, 3	216, 19	1, 16	72, 38

Table 8.11: Resultados para os módulos de elasticidade estimados utilizando MEF e Brox para o phantom2.

$p [KPa]$	$E_{background} [KPa]$	$E_{lesion} [MPa]$	$Time [s]$
1.4	210,58	1,63	263,78
2.8	212,01	1,36	267,97
4.3	210,26	1,57	261,67
5.8	211,59	1,41	260,08
7.3	211,17	2,44	275,49

Os valores médios e os desvios padrão de erro para cada simulação, são dados pela Tabela 8.12.

Table 8.12: Erro médio e desvio padrão para cada associação de métodos para o phantom2.

Method	$\bar{\epsilon}_{background} \pm \bar{\sigma}$	$\bar{\epsilon}_{lesão} \pm \bar{\sigma}$
	(%)	(%)
MEC + Lucas-Kanade	$14,33 \pm 0,98$	$11,04 \pm 11,38$
MEC + Brox	$12,62 \pm 0,61$	$20,54 \pm 11,31$
MEF + Lucas-Kanade	$7,85 \pm 0,94$	$31,34 \pm 7,50$
MEF + Brox	$5,56 \pm 0,35$	$17,70 \pm 6,18$

Os gráficos das Figuras 8.2.1 e 8.2.2 ilustram o erro em função da pressão aplicada para MEC e MEF, considerando uma malha de elementos de contorno e uma malha de elementos finitos com 60 e 324 elementos, respectivamente.

Figura 8.2.1: Erro ϵ (%) para o background em função da pressão aplicada.

Figura 8.2.2: Erro ϵ (%) para a lesão em função da pressão aplicada.

Para cada combinação de método, as Figuras 8.2.3 e 8.2.4 apresentam o erro relativo em função da quantidade de elementos para a lesão do phantom 2. A convergência é assumida quando a diferença entre os vetores de módulo de elasticidade entre duas iterações é, em módulo, $\|\vec{\Delta E}\| \leq 10^{-6}$ ou quando a quantidade máxima de 1000 iterações é atingida. É considerado um valor inicial de $E_i = 1KPa$ para cada material.

Figure 8.2.3: Convergência do método em função da quantidade de elementos de contorno para a região da lesão.

Figure 8.2.4: Convergência do método em função da quantidade de elementos de finitos para a região da lesão.

A Tabela 8.13 apresenta a quantidade máxima de iterações e o tempo médio quando o critério de parada é atingido, considerando uma malha de elementos de contorno de 119 elementos e uma malha de elementos finitos com 3.002 elementos.

Table 8.13: Quantidade de iterações para a convergência e o tempo máximo para cada combinação de método.

Método	Quantidade máxima de iterações	Tempo [s]
MEC + Lucas-Kanade	24	60s
MEC + Brox	26	230s
MEF + Lucas-Kanade	1000	~6h
MEF + Brox	1000	~6h

Capítulo 9

Discussão dos Resultados, Conclusões e Trabalhos Futuros

9.1 Discussão dos Resultados e Conclusão

Esta tese apresenta uma metodologia para elastografia virtual utilizando o Método dos Elementos de Contorno (MEC) e de visão computacional. Até o presente momento, não é de conhecimento do autor sobre o uso do MEC em problemas de elastografia, em oposição ao uso do Método dos Elementos Finitos. Também não é do conhecimento do autor a integração destes métodos numéricos a métodos de visão computacional para análise inversa.

A validação do método de análise inversa, apresentada na seção 5.4, mostra que o MEC apresenta resultados comparáveis ao MEF para solução de problemas de análise inversa, tanto em problemas considerando simulações sem ruído como para simulações com ruído. Estes resultados levaram a considerar a aplicação de ambos os métodos em problemas com imagens de ultrassom.

No capítulo de Resultados, foram mostrados os resultados para a integração dos métodos de FO, MEC e MEF. As análises mostraram que o MEF associado ao algoritmo de Brox apresenta erros para estimação do módulo de elasticidade do background de $1,91 \pm 0,16\%$ e tempo de processamento médio de 321 segundos. No entanto, é necessário enfatizar que a princípio, o módulo de elasticidade do background é sempre conhecido, desta forma, é necessário identificar o módulo de elasticidade da lesão. Neste sentido, o MEC associado a Lucas-Kanade apresentou resultados em relação ao erro, de $9,79 \pm 8,58\%$ para o phantom 1 e $11,04 \pm 11,38\%$ para o phantom 2, considerando tempos médios de 46,3 e 38,4 segundos respectivamente.

Outro ponto importante a ser considerado, é a convergência de cada método como função da quantidade de elemento, o tempo necessário para cada simulação e a quantidade de iterações até que o critério de convergência fosse atingido. Levando em conta estes fatores, a associação entre MEC e Lucas-Kanade apresenta-se como a mais eficiente, como pode ser observados nas Figuras 8.2.3 e 8.2.4 e pela Tabela 8.13.

Considerando os resultados obtidos com diferentes phantoms e condições de pressão, conclui-se a associação MEC e Lucas-Kanade apresenta-se como uma ferramenta útil para outros métodos quantitativos em elastografia.

9.2 Sugestões para Trabalhos Futuros

Como sugestão para trabalhos futuros, tem-se os seguintes temas:

- Validação da metodologia proposta utilizando phantoms físicos;
- Validação da metodologia proposta utilizando imagens de pacientes;
- Aplicação da metodologia proposta em problemas de ultrassom intravascular;
- Implementação da metodologia em placas gráficas, implementando todas as etapas possíveis, como processamento das imagens, métodos numéricos, solvers, etc;
- Geração de malha automática utilizando detecção de bordas automáticas;
- Implementação da metodologia utilizando métodos sem malha, a fim de eliminar a necessidade de um gerador de malhas;
- Implementação da metodologia utilizando outras modalidades de imagem, como tomografia por coerência óptica (OCT);
- Estender o método para solução em 3D, de modo a utilizar informações de outras modalidades e não disponíveis nas imagens de ultrassom modo-B (2D);
- Aplicação de algoritmos de normalização de autovalores em problemas de sub-regiões em elementos de contorno, como descreve Zhou et al. [72], de forma a melhorar a resposta das estimativas para lesões.

Referências Bibliográficas

- [cud] *Cuda Programming Guide*.
- [2] Apostolakis, I. Z., Gastounioti, a., Golemati, S., and Nikita, K. S. (2012). Ultrasound-image-based displacement and strain maps of the human carotid artery using weighted- least-squares optical flow. pages 0–5.
- [3] Baldewsing, R. a., de Korte, C. L., Schaar, J. a., Mastik, F., and van der Steen, A. F. W. (2004). A finite element model for performing intravascular ultrasound elastography of human atherosclerotic coronary arteries. *Ultrasound in medicine & biology*, 30(6):803–13.
- [4] Baldewsing, R. a., Schaar, J. a., Mastik, F., Oomens, C. W. J., and van der Steen, A. F. W. (2005). Assessment of vulnerable plaque composition by matching the deformation of a parametric plaque model to measured plaque deformation. *IEEE transactions on medical imaging*, 24(4):514–28.
- [5] Bathe, K. J. (1996). *Finite Element Procedures*. Prentice Hall.
- [6] Becker, A. A. (1992). *The Boundary Element Method in Engineering*. McGraw Hill International.
- [7] Beer, G., Smith, I., and Deuenser, C. (2008). *The Boundary Element Method with Programming for Engineers and Scientists*. Springer Wien New York.
- [8] Bertrand, M. (1996). Linear Perturbation Method. 15(3):299–313.
- [9] Bilgen, M. and Insana, M. (1998). Covariance analysis of time delay estimates for strained signals. *IEEE Transactions on Signal Processing*, 46(10):2589–2600.
- [10] Bishop, J., Samani, a., Sciarretta, J., and Plewes, D. B. (2000). Two-dimensional MR elastography with linear inversion reconstruction: methodology and noise analysis. *Physics in medicine and biology*, 45:2081–2091.
- [11] Brox, T. and Malik, J. (2011). Large displacement optical flow: Descriptor matching in variational motion estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33:500–513.

- [12] Brox, T., Papenberg, N., and Weickert, J. (2004). High Accuracy Optical Flow Estimation Based on a Theory for Warping. *Computer Vision - ECCV 2004*, 4(May):25–36.
- [13] Burt, P. and Adelson, E. (1983). The Laplacian Pyramid as a Compact Image Code. *IEEE Transactions on Communications*, 31(4):532–540.
- [14] Cardoso, F. M., Moraes, M. C., and Furuie, S. S. (2012). Realistic IVUS image generation in different intraluminal pressures. *Ultrasound in medicine & biology*, 38(12):2104–19.
- [15] Carrion, R. (2002). *Uma Implementação do Método dos Elementos de Contorno para Problemas Viscoelástodinâmicos Estacionários Tridimensionais em Domínios Abertos e Fechados*. PhD thesis, Universidade Estadual de Campinas.
- [16] Cecka, C., Lew, A. J., and Darve, E. (2010). Assembly of finite element methods on graphics processors. *International Journal for Numerical Methods in Engineering*, pages n/a–n/a.
- [17] Chai, C.-K., Speelman, L., Oomens, C. W. J., and Baaijens, F. P. T. (2014a). Compressive mechanical properties of atherosclerotic plaques - indentation test to characterise the local anisotropic behavior. *Journal of Biomechanics*.
- [18] Chai, C.-K., Speelman, L., Oomens, C. W. J., and Baaijens, F. P. T. (2014b). Compressive mechanical properties of atherosclerotic plaques-indentation test to characterise the local anisotropic behaviour. *Journal of biomechanics*, 47(4):784–92.
- [19] Chan, R., Chau, a., Karl, W., Nadkarni, S., Khalil, a., Iftimia, N., Shishkov, M., Tearney, G., Kaazempur-Mofrad, M., and Bouma, B. (2004). OCT-based arterial elastography: robust estimation exploiting tissue biomechanics. *Optics express*, 12(19):4558–4572.
- [20] DiBattista, A. and Noble, J. A. (2014). An efficient block matching and spectral shift estimation algorithm with applications to ultrasound elastography. *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, 61(3):407–19.
- [21] dos Santos Hoeffl, S. (2006). *Análise de Estruturas Tridimensionais de Laminados Através do Método dos Elementos de Contorno*. PhD thesis, Universidade Estadual de Campinas - Unicamp.
- [22] Doyley, M. M. (2012). Model-based elastography: a survey of approaches to the inverse elasticity problem. *Physics in medicine and biology*, 29(3):997–1003.
- [23] Doyley, M. M. and Parker, K. J. (2014). Elastography: general principles and clinical applications. *Ultrasound clinics*, 9(1):1–11.
- [24] Fang, Q. and Boas, D. a. (2009). Tetrahedral mesh generation from volumetric binary and gray scale images. *International Symposium on Biomedical Imaging, (Isbi)*:1563–1569.

- [25] Freutel, M., Schmidt, H., Dürselen, L., Ignatius, A., and Galbusera, F. (2014). Finite element modeling of soft tissues: material models, tissue interaction and challenges. *Clinical biomechanics (Bristol, Avon)*, 29(4):363–72.
- [26] Fung, Y. C. and Tong, P. (2005). *Classical and Computational Solid Mechanics*. World Scientific.
- [27] Gao, X.-W. and Davies, T. (2002). *Boundary Element Programming in Mechanics*. Cambridge University Press.
- [28] Gao, X.-W. and He, M.-C. (2005). A new inverse analysis approach for multi-region heat conduction BEM using complex-variable-differentiation method. *Engineering Analysis with Boundary Elements*, 29(8):788–795.
- [29] Gaul, L., Kögl, M., and Wagner, M. (2003). *Boundary Element Methods for Engineers and Scientists: An Introductory Course with Advanced Topics*. Springer.
- [30] Gennisson, J.-L., Deffieux, T., Macé, E., Montaldo, G., Fink, M., and Tanter, M. (2010). Viscoelastic and anisotropic mechanical properties of in vivo muscle tissue assessed by supersonic shear imaging. *Ultrasound in medicine & biology*, 36(5):789–801.
- [31] Geuzaine, C. and Remacle, J.-F. (2009). Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, 79:1309–1331.
- [32] Guo, Z., You, S., Wan, X., and Bićanić, N. (2010). A FEM-based direct method for material reconstruction inverse problem in soft tissue elastography. *Computers & Structures*, 88(23-24):1459–1468.
- [33] Holzapfel, G. a., Sommer, G., and Regitnig, P. (2004). Anisotropic Mechanical Properties of Tissue Components in Human Atherosclerotic Plaques. *Journal of Biomechanical Engineering*, 126(5):657.
- [34] Horn, B. K. P. and Schunck, B. G. (1981). DETERMINING OPTICAL FLOW. In *Proceedings of SPIE - The International Society for Optical Engineering*, volume 281, pages 319–331.
- [35] Jensen, J. A. and Nikolov, S. I. (2000). Fast simulation of ultrasound images. In *IEEE Ultrasonics Symposium*.
- [36] Julia, M., Calas, G., Gutflen, B., Coelho, W., and Pereira, D. A. (2012). CAD and mammography : why use this tool ? *. *Radiol Bras*, 45(1):46–52.
- [37] Kallel, F. and Ophir, J. (1997). A Least-Square Strain Estimator for Elastography. *Ultrasonic Imaging*, 19:195–208.

- [38] Kamiya, N. and Koide, M. (1993). Computational Mechanics Adaptive boundary element for multiple subregions. pages 69–80.
- [39] Kane, J. H. (1994). *Boundary Element Analysis in Engineering Continuum Mechanics*. Prantice Hall.
- [40] Khalil, A. S., Chan, R. C., Chau, A. H., Bouma, B. E., and Mofrad, M. R. K. (2005). Tissue elasticity estimation with optical coherence elastography: Toward mechanical characterization of in vivo soft tissue. *Annals of Biomedical Engineering*, 33(11):1631–1639.
- [41] Kirkpatrick SJ, Wang RK, D. D. (2006). OCT-based elastography for large and small deformations. *Optical express*, 14(24):11585 – 11597.
- [42] Kreyszig, E. (2006). *Advanced Engineering Mathematics 9th Edition*. Jhon Wiley & Sons.
- [43] Krouskop, T. A. and Wheeler, T. M. (1998). Elastic Moduli of Breast and Prostate Tissues under Compression. *Ultrason Imaging*, 20:260–274.
- [44] Kwon, Y. W. and Bang, H. (1997). *The Finite Element Method Using MATLAB*. CRC Press.
- [45] Labaki, J., Ferreira, L., and Mesquita, E. (2011). Constant Boundary Elements on Graphics Hardware: a GPU-CPU Complementary Implementation. *SciELO Brasil*, XXXIII(4):475–482.
- [46] Liu, K., Zhang, P., Shao, J., Zhu, X., Zhang, Y., and Bai, J. (2009a). A 2d strain estimator with numerical optimization method for soft tissue elastography. *Ultrasonics*, 49:724–732.
- [47] Liu, K., Zhang, P., Shao, J., Zhu, X., Zhang, Y., and Bai, J. (2009b). A 2D strain estimator with numerical optimization method for soft-tissue elastography. *Ultrasonics*, 49(8):723–32.
- [48] Lu, X. (2003). A new subregion boundary element method. 34.
- [49] Lu, X. and Wu, W.-l. (2005). A new subregion boundary element technique based on the domain decomposition method. *Engineering Analysis with Boundary Elements*, 29(10):944–952.
- [50] Lucas, B. D. and Kanade, T. (1981). An iterative image registration technique with an application to stereo vision. *Proceedings of the 7th international joint Conference on Artificial Intelligence*, (x):674–679.
- [51] Marczak, R. J. (2008). Optimization of elastic structures using boundary elements and a topological-shape sensitivity formulation. *Latin American Journal of Solids and Structures*, 5:99–117.
- [52] Martins, J., Kroo, I., and Alonso, J. (2000). An automated method for sensitivity analysis using complex variables. *38th Aerospace Sciences Meeting and Exhibit*.

- [53] Maurice, R. L., Ohayon, J., Finet, G., and Cloutier, G. (2004). Adapting the Lagrangian speckle model estimator for endovascular elastography: theory and validation with simulated radio-frequency data. *The Journal of the Acoustical Society of America*, 116(2):1276–1286.
- [54] Mundstock, D. C. and Marczak, R. J. (2008). Boundary element sensitivity evaluation for elasticity problems using complex variable method. *Structural and Multidisciplinary Optimization*, 38(4):423–428.
- [55] Muthupillai R, Lomas DJ, Rossman PJ, Greenleaf JF, Manduca A, E. R. (1995). Magnetic-resonance elastography by direct visualization of propagating acoustic strain wave. *Science*, 7(269):1854–1857.
- [56] O’Donnell, M., a.R. Skovoroda, Shapo, B., and Emelianov, S. (1994). Internal displacement and strain imaging using ultrasonic speckle tracking. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, 41(3):314–325.
- [57] Ophir, J., Céspedes, I., Ponnekanti, H., Yazdi, Y., and Li, X. (1991). Elastography: a quantitative method for imaging the elasticity of biological tissues. *Ultrasonic imaging*, 13:111–134.
- [58] Pan, X., Gao, J., Tao, S., Liu, K., Bai, J., and Luo, J. (2014a). A two-step optical flow method for strain estimation in elastography: Simulation and phantom study. *Ultrasonics*, 54(4):990–6.
- [59] Pan, X., Gao, J., tao, S., Liu, K., Bai, J., and Luo, J. (2014b). A two-step optical flow method for strain estimation in elastography: Simulation and phantom study. *Ultrasonics*, 54:990–996.
- [60] Perriñez, P. R., Pattison, A. J., Kennedy, F. E., Weaver, J. B., and Paulsen, K. D. (2010). Contrast detection in fluid-saturated media with magnetic resonance poroelastography. *Medical physics*, 37(3):3518–3526.
- [61] Perrinez, P. R., Kennedy, F. E., Van Houten, E. E. W., Weaver, J. B., and Paulsen, K. D. (2010). Magnetic resonance poroelastography: An algorithm for estimating the mechanical properties of fluid-saturated soft tissues. *IEEE Transactions on Medical Imaging*, 29(3):746–755.
- [62] Powell, M. J. D. (1964). An efficient method for finding the minimum of a function of several variables without calculating derivatives. *The Computer Journal*, 7:155–162.
- [63] Qiu, Y., Sridhar, M., Tsou, J. K., Lindfors, K. K., and Insana, M. F. (2008). Ultrasonic Viscoelasticity Imaging of Nonpalpable Breast Tumors. Preliminary Results. *Academic Radiology*, 15(12):1526–1533.

- [64] Raghavan, K. R., a.E. Yagle, and Andrew, E. (1994). Forward and Inverse Problems in Elasticity Imaging of Soft Tissues. *IEEE Transactions on Nuclear Science*, 41(4):1639–1648.
- [65] Romano, a. J., Shirron, J. J., and Bucaro, J. a. (1998). On the noninvasive determination of material parameters from a knowledge of elastic displacements theory and numerical simulation. *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, 45(3):751–759.
- [66] Sanders, J. and Kandrot, E. (2011). *Cuda By Exemple*. Addison-Wesley.
- [67] Santiago, J., Telles, J., and Valentim, V. (1999). An optimized block matrix manipulation for boundary elements with subregions. *Advances in Engineering Software*, 30(9-11):701–713.
- [68] Sinkus, R., Tanter, M., Catheline, S., Lorenzen, J., Kuhl, C., Sondermann, E., and Fink, M. (2005a). Imaging anisotropic and viscous properties of breast tissue by magnetic resonance-elastography. *Magnetic Resonance in Medicine*, 53:372–387.
- [69] Sinkus, R., Tanter, M., Xydeas, T., Catheline, S., Bercoff, J., and Fink, M. (2005b). Viscoelastic shear properties of in vivo breast lesions measured by MR elastography. *Magnetic Resonance Imaging*, 23:159–165.
- [70] Weaver JB, Van Houten EE, Miga MI, Kennedy FE, P. K. (2001). Magnetic resonance elastography using 3D gradient echo measurements of steady-state motion. *Medical physics*, 8:1620–1628.
- [71] Zakaria, T., Qin, Z., and Maurice, R. L. (2010). Optical-flow-based B-mode elastography: application in the hypertensive rat carotid. *IEEE transactions on medical imaging*, 29(2):570–8.
- [72] Zhou, H., Jiang, W., Hu, H., and Niu, Z. (2013). Boundary element methods for boundary condition inverse problems in elasticity using PCGM and CGM regularization. *Engineering Analysis with Boundary Elements*, 37(11):1471–1482.
- [73] Zhu, Y., Hall, T. J., and Jiang, J. (2003). A finite-element approach for Young’s modulus reconstruction. *IEEE transactions on medical imaging*, 22(7):890–901.
- [Zienkiewicz and Emeritus] Zienkiewicz, O. C. and Emeritus, F. The Finite Element Method Fifth edition Volume 2 : Solid Mechanics. 2.