

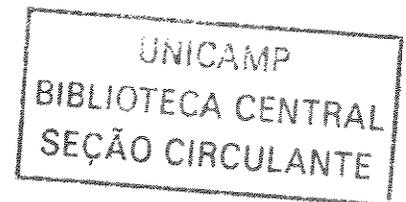
TESE DEFENDIDA POR Alberto Costa Nogueira
Junior E APROVADA PELO
COMISSÃO JULGADORA EM 13 / 11 / 2002
Marco Lúcio Bittencourt
ORIENTADOR

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
COMISSÃO DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA

FORMULAÇÃO P DO MÉTODO DE ELEMENTOS FINITOS
EM PROBLEMAS DE ELASTICIDADE LINEAR E
NÃO-LINEAR COM MALHAS 3D E EM MÉTODOS
MULTIGRID ALGÉBRICOS

Autor: Alberto Costa Nogueira Junior
Orientador: Prof. Dr. Marco Lúcio Bittencourt

11/02



UNICAMP
BIBLIOTECA CENTRAL

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
COMISSÃO DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA
DEPARTAMENTO DE PROJETO MECÂNICO

FORMULAÇÃO *P* DO MÉTODO DE ELEMENTOS FINITOS
EM PROBLEMAS DE ELASTICIDADE LINEAR E
NÃO-LINEAR COM MALHAS 3D E EM MÉTODOS
MULTIGRID ALGÉBRICOS

Autor: Alberto Costa Nogueira Junior
Orientador: Prof. Dr. Marco Lúcio Bittencourt

Curso: Engenharia Mecânica
Área de Concentração: Mecânica dos Sólidos e Projeto Mecânico

Tese de doutorado apresentada à comissão de Pós Graduação da Faculdade de Engenharia Mecânica como requisito para a obtenção do título de Doutor em Engenharia Mecânica.

Campinas, 2002
S.P. - Brasil

UNIDADE	Re
Nº CHAMADA	UNICAMP N689f
V	EX
TOMBO BCI	53208
PROC.	124103
C	<input type="checkbox"/>
D	<input checked="" type="checkbox"/>
PREÇO	R\$ 11,00
DATA	14/04/03
Nº CPD	

CMO0162147-2

,10 287891

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

N689f

Nogueira Junior, Alberto Costa

Formulação p do método de elementos finitos em problemas de elasticidade linear e não-linear com malhas 3D e em métodos multigrid algébricos / Alberto Costa Nogueira Junior. --Campinas, SP: [s.n.], 2002.

Orientador: Marco Lúcio Bittencourt.

Tese (doutorado) - Universidade Estadual de Campinas, Faculdade de Engenharia Mecânica.

1. Método de elementos finitos. 2. Elasticidade. 3. Programação orientada a objetos (Computação). I. Bittencourt, Marco Lúcio. II. Universidade Estadual de Campinas. Faculdade de Engenharia Mecânica. III. Título.

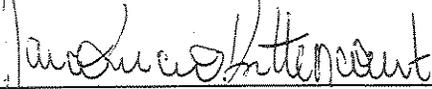
UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
COMISSÃO DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA
DEPARTAMENTO DE PROJETO MECÂNICO

TESE DE DOUTORADO

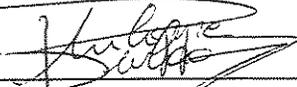
FORMULAÇÃO P DO MÉTODO DE ELEMENTOS FINITOS
EM PROBLEMAS DE ELASTICIDADE LINEAR E
NÃO-LINEAR COM MALHAS 3D E EM MÉTODOS
MULTIGRID ALGÉBRICOS

Autor: Alberto Costa Nogueira Junior

Orientador: Prof. Dr. Marco Lúcio Bittencourt



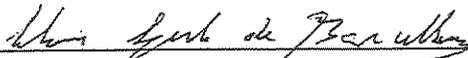
Prof. Dr. Marco Lúcio Bittencourt, Presidente
DPM/FEM/UNICAMP



Prof. Dr. Philippe Remy Bernard Devloo
DES/FEC/UNICAMP



Prof. Dr. Renato Pavanello
DMC/FEM/UNICAMP



Prof. Dr. Clóvis Sperber de Barcellos
PUC-MG



Prof. Dr. Carlos Armando Magalhães Duarte
EESC/USP

1744 8266

Campinas, 13 de novembro de 2002.

Dedico este trabalho às duas mulheres mais importantes em minha vida: minha mãe, Norma, por ter me inundado os olhos, até o último instante, com seu entusiasmo e alegria pela vida; e minha esposa, Juliete, por ter me apontado de forma sutil e delicada o caminho que me conduziu ao encontro do meu próprio eu. A vocês, o meu sincero e humilde obrigado.

Agradecimentos

Ao meu orientador Prof. Dr. Marco Lúcio Bittencourt pela oportunidade que me ofereceu e pela confiança que depositou em mim e no meu trabalho.

Ao meu pai Alberto Costa Nogueira e meus irmãos Jeferson Nogueira e Danielle Costa Nogueira pela amizade, compreensão e carinho em todos os momentos que vivemos juntos. Gostaria de repartir com vocês a autoria de minha homenagem àquela que nos presenteou com o mais claro exemplo de simplicidade, sabedoria e amor.

Aos meus companheiros de luta e amigos Cláudio Alessandro de Carvalho Silva e Renato Fernandes Cantão por terem contribuído de forma decisiva para o aprimoramento da minha formação técnica e humana.

Aos meus companheiros de laboratório Luciano Driemeier, Wallace Gusmão, Carlos Eduardo Leite Pereira e Maurício Barbatto pela amizade, incentivo e bom humor ao longo do período em que trabalhamos juntos.

Ao Prof. Petrônio Pulino do IMECC pelo exemplo de postura profissional, amizade e incentivo ao longo de todo o desenvolvimento deste trabalho. Devo agradecer-lhe ainda pela sua enorme solicitude e espírito colaborador que tanto acrescentaram à minha formação como pesquisador.

Ao Prof. Mario Arlindo Casarin Jr. do IMECC pela sua colaboração no direcionamento inicial deste projeto de doutorado.

Ao Prof. Milton Dias Jr. pelo apoio dado quando exerceu a chefia do departamento DPM/FEM.

Ao Prof. Euclides Mesquita Neto pela sua colaboração como coordenador da Sub-CPG/FEM.

Aos grandes amigos de república Ed, Lúcio, Lucy, Stela, Ivan, Dani, Junior e Juliete pela alegria de ter podido conviver com pessoas tão admiráveis quanto vocês.

À minha tão adorada esposa Juliete pelo seu carinho, compreensão e amor.

À FAPESP, DPM, FEM e UNICAMP pelos suportes institucional e financeiro que possibilitaram a realização deste trabalho.

A todos aqueles de quem eu possa não ter me lembrado e que de uma maneira ou de outra contribuíram para que eu me tornasse melhor do que já fui um dia.

Of all the things we wear, our smile is the most important.

Dadi Janki

Resumo

NOGUEIRA Jr., Alberto Costa, *Formulação p do Método de Elementos Finitos em Problemas de Elasticidade Linear e Não-Linear com Malhas 3D e em Métodos Multigrid Algébricos*, Campinas : Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, 2002. 185p. Tese (Doutorado)

O presente trabalho tem por objetivo principal desenvolver e implementar (em linguagem C++) a versão- p do Método de Elementos Finitos (MEF) para problemas elásticos lineares e não-lineares com malhas 3D não-estruturadas. Para se atingir essa meta, realiza-se primeiramente um estudo detalhado do comportamento teórico-numérico de diversas funções de base hierárquicas para triângulos e tetraedros disponíveis na literatura. Como consequência desse estudo preliminar, adota-se a formulação p 3D apresentada em (Karniadakis e Sherwin, 1999) para o desenvolvimento de aplicações em elasticidade linear e não-linear. Esse desenvolvimento é posteriormente utilizado como ponto de partida para o estudo e a implementação de técnicas multigrid algébricas associadas a versão- p do MEF. As aplicações desse tipo de técnica são voltadas para a solução de problemas elípticos lineares 2D e 3D. A validação das técnicas multigrid algébricas do ponto de vista da eficiência numérica envolve ainda o estudo de pré-condicionadores para a versão- p do MEF e o desenvolvimento de estratégias multi- hp adaptáveis para problemas 2D.

Palavras Chave

Elementos Finitos, Versão- p , Elasticidade Linear e Não-Linear, Malhas Não-Estruturadas, Multigrid Algébrico, Programação Orientada a Objetos, C++

Abstract

NOGUEIRA Jr., Alberto Costa, *p-Version of the Finite Element Method applied to Linear and Non-Linear Structural Elastic Problems with 3D Meshes and to Algebraic Multigrid Methods*, Campinas : Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, 2002. 185p. Tese (Doutorado)

The main purpose of this work is to develop and implement (in C++ language) the p -version of the Finite Element Method (**FEM**) applied to linear and non-linear structural elastic problems with **3D** non-structured meshes. To achieve this goal, we conduct a previous study which compares some sets of triangular and tetrahedral hierarchical basis functions available in the literature. As a consequence of this study, we adopt the formulation suggested by (Karniadakis e Sherwin, 1999) to construct some applications related to linear and non-linear structural elastic problems. This development is subsequently used as the starting point to the study and implementation of the algebraic multigrid techniques based on the p -version. This kind of technique is applied to **2D** and **3D** elliptic linear problems. To validate the proposed algebraic multigrid schemes from a computational point of view we also consider the implementation of preconditioning algorithms for the p -version of the **FEM** as well as adaptive multi- hp strategies applied to **2D** problems.

Keywords

Finite Elements, p -Version, Linear and Non-Linear Elasticity, Non-Structured Meshes, Algebraic Multigrid, Object-Oriented Programming, C++

Índice

1	Introdução	1
1.1	Motivação	1
1.2	Revisão Bibliográfica	5
1.2.1	Breve histórico sobre a evolução da versão- p do MEF	5
1.2.2	Funções de forma	7
1.2.3	Mapeamento geométrico	10
1.2.4	Técnicas de solução para sistemas lineares	11
1.3	Objetivos	15
1.4	Contribuições do Trabalho	16
1.5	Organização do Texto	17
2	Funções de Base Hierárquicas	19
2.1	Elementos Hierárquicos para Malhas Estruturadas e Não-estruturadas	20
2.1.1	Quadriláteros	20
2.1.2	Hexaedros	21
2.1.3	Triângulos	23
2.1.4	Tetraedros	28
2.2	Ortogonalidade dos Modos Internos Associados às Novas Funções de Base	32
2.2.1	Elemento triangular	32
2.2.2	Elemento tetraédrico	34
2.3	Condicionamento de Matrizes Locais	36
2.3.1	Triângulos	37
2.3.2	Tetraedros	38
2.4	Esparsidade de Matrizes Locais	39
2.4.1	Triângulos	40
2.4.2	Tetraedros	41
2.5	Esparsidade e Condicionamento de Matrizes Locais sob o Efeito de Distorção	43
2.5.1	Triângulos	43
2.5.2	Tetraedros	44
2.6	Condicionamento de Matrizes Globais para o Problema de <i>Poisson 2D</i>	46
2.7	Solução do Problema de <i>Poisson 2D</i> Usando o Método de Gradientes Conjugados	48
2.8	Conclusões	51
3	Formulação p em 3 Dimensões Aplicada à Elasticidade Linear e Não-Linear	53
3.1	Operações Locais	54
3.1.1	Integração numérica sobre regiões tetraédricas	54
3.1.2	Diferenciação numérica sobre regiões tetraédricas	58
3.1.3	Mapeamento de elementos	60

3.2	Operações Globais	61
3.2.1	Conectividade entre elementos tetraédricos	61
3.2.2	Numeração de graus de liberdade	66
3.3	Descrição de Fronteiras em Problemas de Valor de Contorno	68
3.3.1	Tipos de condições de contorno associados a problemas de elasticidade	68
3.3.2	Aplicação de condições de contorno do tipo Neumann não-homogêneas	69
3.4	Aplicações em Elasticidade Linear	73
3.4.1	Formulação do problema de elasticidade infinitesimal	73
3.4.2	Resultados numéricos para elasticidade linear	76
3.5	Aplicação em Elasticidade Não-Linear	89
3.5.1	Formulação do problema de elasticidade finita	89
3.5.2	Resultados Numéricos para elasticidade não-linear	90
3.6	Conclusões	96
4	Métodos Multigrid Algébricos para a Versão-p do MEF	99
4.1	Métodos Multi- p	100
4.2	Pré-Condicionadores para a Versão- p	107
4.2.1	Pré-condicionadores multi- p	107
4.2.2	Pré-condicionador 2D de <i>Babuska</i> ¹	110
4.3	Métodos Multi- hp Adaptáveis	111
4.4	Resultados Numéricos para o Problema de <i>Poisson 2D</i>	112
4.4.1	Método multi- p sem condensação estática	114
4.4.2	Método multi- p com condensação estática	117
4.4.3	Método multi- hp adaptável	120
4.5	Resultados Numéricos para Problemas de Elasticidade Linear 3D	124
4.6	Conclusões	130
5	Implementação de um Código Orientado por Objetos para a Versão-p	133
5.1	Arquitetura do Módulo de Solução	134
5.2	Modificações Efetuadas nas Classes Básicas para Acomodação da Versão- p	138
5.3	Estruturas Locais que Distinguem as Versões h e p	141
5.4	Estruturas Globais Associadas à Versão- p 3D	144
5.5	Estruturas Básicas em Métodos Multigrid Algébricos	146
6	Conclusões Gerais e Perspectivas Futuras	149
	Referências Bibliográficas	151
A	Polinômios Ortogonais	163
A.1	Polinômios de Legendre	163
A.2	Polinômios de Jacobi	164
B	Integração Numérica 1D do Tipo <i>Gauss-Jacobi</i>	167
B.1	Elementos Básicos das Regras de Integração Numéricas ²	167
B.2	Regras do Tipo <i>Gauss-Jacobi</i>	169
B.2.1	Regra de <i>Gauss-Jacobi</i>	169
B.2.2	Regra de <i>Gauss-Jacobi-Radau</i>	170
B.2.3	Regra de <i>Gauss-Jacobi-Lobatto</i>	171
B.3	Avaliação das Raízes dos Polinômios de <i>Jacobi</i>	171

C	Discretização da Formulação Variacional para Elasticidade Finita	175
C.1	Discretização para Não-Linearidades Geométricas	175
C.2	Formas Matriciais de Discretização	177
D	Custo Computacional dos Métodos Iterativos e Multigrid Algébricos	183
D.1	Métodos Iterativos Clássicos	183
D.2	Método Multigrid Algébrico para a Versão- p	183

Lista de Figuras

1.1	Exemplo de um problema estrutural não-linear discretizado com uma malha de elementos tetraédricos a partir do programa comercial ANSYS 6.0. Para esse tipo de problema, a formulação p não está disponível.	3
2.1	Elementos de referência quadrilateral e hexaédrico de (Szabó e Babuška, 1991).	21
2.2	Elementos de referência triangulares.	24
2.3	Mapeamento de coordenadas do elemento quadrilateral para o elemento triangular (Sherwin e Karniadakis, 1995).	25
2.4	Funções hierárquicas de ordens 4 e 5 associadas às arestas e às faces, respectivamente, dos diferentes elementos estudados.	27
2.5	Elementos de referência tetraédricos.	28
2.6	Mapeamento de coordenadas do elemento hexaédrico para o elemento tetraédrico de (Sherwin e Karniadakis, 1995).	29
2.7	Número de condição das matrizes de rigidez e massa locais vs. ordem polinomial p , considerando-se as definições de κ_1 , κ_2 e κ_3 (triângulos).	38
2.8	Número de condição das matrizes de rigidez e massa locais vs. ordem polinomial p , considerando-se as definições de κ_1 , κ_2 e κ_3 (tetraedros).	39
2.9	Porcentagem de elementos nulos das matrizes de rigidez e massa locais vs. ordem polinomial p (triângulos).	40
2.10	Padrão de esparsidade das matrizes de rigidez e massa locais associadas aos cinco tipos de funções hierárquicas estudados (triângulos).	41
2.11	Porcentagem de elementos nulos das matrizes de rigidez e massa locais vs. ordem polinomial p (tetraedros).	42
2.12	Padrão de esparsidade das matrizes de rigidez e massa locais associadas aos cinco tipos de funções hierárquicas estudados (tetraedros).	42
2.13	Número de condição das matrizes de rigidez locais vs. ordem polinomial p considerando-se as definições de κ_1 , κ_2 e κ_3 , e ambos os casos de distorção: (a) alongamento e (b) achatamento (triângulos).	44

2.14	Porcentagem de elementos nulos das matrizes de rigidez locais vs. ordem polinomial p : (a) triângulos padrões alongados; (b) triângulos padrões achatados.	45
2.15	Número de condição das matrizes de rigidez locais vs. ordem polinomial p considerando-se as definições de κ_1 , κ_2 e κ_3 , e ambos os casos de distorção: (a) alongamento e (b) achatamento (tetraedros).	45
2.16	Porcentagem de elementos nulos das matrizes de rigidez locais vs. ordem polinomial p : (a) tetraedros padrões alongados; (b) tetraedros padrões achatados.	46
2.17	Número de condição das matrizes de rigidez globais vs. ordem polinomial p	47
2.18	Domínios e malhas utilizadas nos dois tipos de problemas analisados (Novotny et al., 1999).	49
2.19	Solução analítica do problema de <i>Poisson</i> para Ω quadrado (Novotny et al., 1999).	50
2.20	Solução analítica do problema de <i>Poisson</i> para Ω em formato L (Novotny et al., 1999).	50
2.21	Número de iterações executadas pelo método GC nas versões padrão e com pré-condicionamento diagonal vs. ordem polinomial p	51
3.1	Pontos de quadratura sobre o tetraedro de referência T^3 com $Q_1 = Q_2 = Q_3 = 7$. Distribuição de pontos de <i>Gauss-Lobatto</i> na direção ξ e <i>Gauss-Radau</i> nas direções η e ζ (Karniadakis e Sherwin, 1999).	57
3.2	Linhas de coordenadas sobre as faces do elemento tetraédrico (Sherwin e Karniadakis, 1996).	62
3.3	Conectividade entre faces de elementos tetraédricos definida segundo (Sherwin e Karniadakis, 1996).	63
3.4	Identificação das entidades topológicas sobre o elemento tetraédrico (Sherwin e Karniadakis, 1996).	64
3.5	Representação 2D do comportamento das funções de base associadas as faces de número 1 compartilhadas por dois elementos tetraédricos distintos quando não há inversão de sinal dos modos ímpares.	65
3.6	Representação paramétrica de superfícies com condição de contorno de <i>Neumann</i> não-homogênea.	70
3.7	Malha de elementos tetraédricos sobre a viga submetida ao seu peso próprio e engastada em sua extremidade superior.	77
3.8	Decaimento do erro da solução do problema da viga submetida ao peso próprio em função do número de graus de liberdade associados aos refinamentos p de ordem 1 a 8. Considera-se a viga com comprimentos $c = 3, 9$ e $18 m$	79
3.9	Erro percentual cometido na aproximação do deslocamento vertical u_y do ponto $(x, y, z) = (1, c, 1)$ pertencente à extremidade livre da viga submetida ao peso próprio em função da ordem polinomial p	80

3.10	Malhas empregadas na análise da viga em balanço submetida ao peso próprio e a uma carga de superfície de valor constante.	81
3.11	Condições de contorno aplicadas sobre a viga em balanço.	81
3.12	Configurações inicial (linha tracejada) e final (linha cheia) da viga em balanço (malha de 34 elementos).	81
3.13	Decaimento do erro relativo na norma energia em função do número de graus de liberdade envolvidos no problema da viga em balanço.	83
3.14	Decaimento do erro percentual em função do número de graus de liberdade associados às três malhas consideradas no problema da viga em balanço.	84
3.15	Malhas empregadas na análise da viga bi-engastada submetida ao peso próprio e a uma carga de superfície de valor constante.	86
3.16	Condições de contorno aplicadas sobre a viga bi-engastada.	86
3.17	Configurações inicial (linha tracejada) e final (linha cheia) da viga bi-engastada (malha de 35 elementos).	87
3.18	Decaimento do erro relativo na norma energia em função do número de graus de liberdade envolvidos no problema da viga bi-engastada.	87
3.19	Decaimento do erro percentual em função do número de graus de liberdade associados às três malhas consideradas no problema da viga bi-engastada.	88
3.20	Viga em balanço não-linear submetida ao peso próprio e a uma carga de superfície de valor constante.	92
3.21	Configurações inicial (linha tracejada) e final (linha cheia) da viga em balanço não-linear (malha de 23 elementos).	92
3.22	Decaimento do erro da solução do problema da viga em balanço não-linear em função do número de graus de liberdade associados aos refinamentos p de ordem 1 a 8.	94
3.23	Convergência das iterações de <i>Newton-Raphson</i> medida na norma-2 (euclidiana) segundo três critérios diferentes em função do número total de iterações necessárias para se chegar à solução do problema da viga em balanço não-linear. Considera-se uma malha com ordem polinomial uniforme $p = 5$	95
3.24	Passos de carregamento em função do deslocamento na direção y de um nó sobre a viga em balanço considerada.	95
4.1	Estratégias multi- p	103
4.2	Domínios e malhas utilizadas nos dois exemplos analisados (Novotny et al., 1999).	113
4.3	Problema 1: Ω quadrado. Método multi- p sem condensação.	116
4.4	Problema 2: Ω em L . Método multi- p sem condensação.	117
4.5	Problema 1: Ω quadrado. Método multi- p com condensação.	119
4.6	Problema 2: Ω em L . Método multi- p com condensação.	121

4.7	Erro relativo η (%) em cada malha do processo adaptável vs. Número de graus de liberdade.	122
4.8	Número de operações (Mflop) vs. Número de graus de liberdade. Método multi- <i>hp</i> sem condensação.	123
4.9	Número de operações (Mflop) vs. Número de graus de liberdade. Método multi- <i>hp</i> com condensação.	125
4.10	Malha de 18 elementos tetraédricos sobre a viga submetida ao seu peso próprio. Exemplo 1 associado à análise dos métodos multi- <i>p</i> 3D lineares.	126
4.11	Malhas empregadas na análise da viga em balanço submetida ao peso próprio e a uma carga de superfície de valor constante. Exemplo 2 associado à análise dos métodos multi- <i>p</i> 3D lineares.	127
4.12	Malhas empregadas na análise da viga bi-engastada submetida ao peso próprio e a uma carga de superfície de valor constante. Exemplo 3 associado à análise dos métodos multi- <i>p</i> 3D lineares.	127
4.13	Viga submetida ao peso próprio. Malha de 18 elementos.	129
4.14	Viga em balanço.	129
4.15	Viga bi-engastada.	130
5.1	Composição da classe DiscreteModel (Silva, 2002).	135

Lista de Tabelas

2.1	Expressões para as funções hierárquicas associadas aos elementos triangulares.	26
2.2	Expressões para as funções hierárquicas associadas aos elementos tetraédricos.	31
3.1	Comparação pontual entre as soluções produzidas pela formulação p proposta e aquela empregada no programa ANSYS 6.0 para o caso da viga em balanço.	85
3.2	Comparação pontual entre as soluções produzidas pela formulação p proposta e aquela empregada no programa ANSYS 6.0 para o caso da viga bi-engastada.	89
3.3	Comparação pontual entre as soluções produzidas pela formulação p não-linear proposta e pela formulação h para grandes deslocamentos utilizada pelo programa ANSYS 6.0 para o caso da viga em balanço considerada.	96
4.1	Multi- p padrão: domínio Ω quadrado.	114
4.2	Multi- p padrão: domínio Ω em L	114
4.3	Multi- p padrão: domínio Ω quadrado.	115
4.4	Multi- p padrão: domínio Ω em L	115
4.5	Multi- p com condensação: domínio Ω quadrado.	118
4.6	Multi- p com condensação: domínio Ω em L	118
4.7	Multi- p com condensação: domínio Ω quadrado.	118
4.8	Multi- p com condensação: domínio Ω em L	120
4.9	Multi- hp padrão: domínio Ω quadrado.	121
4.10	Multi- hp padrão: domínio Ω em L	121
4.11	Multi- hp com condensação: domínio Ω quadrado.	121
4.12	Multi- hp com condensação: domínio Ω em L	121
4.13	Multi- hp padrão: domínio Ω quadrado.	122
4.14	Multi- hp padrão: domínio Ω em L	123
4.15	Multi- hp com condensação: domínio Ω quadrado.	124
4.16	Multi- hp com condensação: domínio Ω em L	124
4.17	Viga submetida ao peso próprio.	126
4.18	Viga em balanço.	127

4.19	Viga bi-engastada.	127
4.20	Multi- p 3D com condensação: Viga submetida ao peso próprio.	128
4.21	Multi- p 3D com condensação: Viga em balanço.	128
4.22	Multi- p 3D com condensação: Viga bi-engastada.	128
D.1	Expressões para o cálculo do custo computacional dos métodos iterativos empregados neste trabalho.	184

Capítulo 1

Introdução

1.1 Motivação

Ferramentas de simulação computacional têm se tornado cada vez mais populares no desenvolvimento de projetos de engenharia. Uma das razões que podem ser apontadas para explicar tal fenômeno é a necessidade de se obter informações rápidas, seguras e de baixo custo que facilitem a tomada de decisões concernentes ao direcionamento desses projetos. Dentro desse contexto e particularmente na simulação de componentes estruturais, torna-se necessário resolver problemas físicos descritos por equações diferenciais parciais (**EDP**) lineares ou não-lineares definidas, em geral, sobre complexas geometrias. Uma forma consagrada de se obter soluções aproximadas dessas equações é a utilização de métodos numéricos que se baseiam em técnicas de discretização. Entre as técnicas de discretização mais usadas, sobretudo para a classe de problemas de elasticidade estrutural, o Método de Elementos Finitos (**MEF**) tem se destacado amplamente.

Basicamente, é possível distinguir três diferentes variantes do **MEF**: a versão- h , a versão- p e a versão- hp . Na versão- h (Hughes, 1987; Johnson, 1990; Oden et al., 1981; Zienkiewicz e Taylor, 1989), obtém-se convergência da solução aproximada reduzindo-se o tamanho médio (h) dos elementos da malha de discretização, mantendo-se fixa a ordem polinomial das funções de base em cada elemento. Na versão- p (Babuška et al., 1981; Babuška e Suri, 1987; Babuška e Suri, 1990; Szabó e Babuška, 1991), ao contrário, obtém-se convergência aumentando-se a ordem polinomial (p) das funções de base (local ou globalmente), mantendo-se fixa a malha de discretização. A combinação dessas duas

versões pela aplicação simultânea da redução do tamanho dos elementos e do aumento da ordem polinomial das funções de base identifica a versão- hp (Babuška, 1988; Babuška e Guo, 1988; Babuška e Guo, 1989; Babuška e Suri, 1990; Szabó e Babuška, 1991).

A versão- p do MEF tem sido investigada de forma intensiva ao longo dos últimos 15 anos e tem se demonstrado superior à versão- h clássica num número significativo de aplicações de importância prática. A maior parte dessas aplicações estiveram inicialmente restritas a problemas elípticos lineares como a equação de *Poisson*, as equações de *Lamé*, o modelo de *Reissner-Mindlin* para placas (Nogueira Jr., 1998), entre outros. Mais recentemente, mostrou-se que a versão- p pode ser estendida com êxito a problemas elásticos com não-linearidades geométricas (Krause et al., 1995), análise elastoplástica de estruturas bidimensionais (2D) (Holzer e Yosibash, 1996; Szabó et al., 1995; Nie e Niu, 1997), problemas de contato (Volpert et al., 1997; Paczelt et al., 1999), teoria de deformações plásticas (Duster e Rank, 2001) e análise de deformações em materiais quasi-incompressíveis (Chilton e Suri, 2001). Embora a versão- p tenha sido aplicada com sucesso a esses vários tipos de problemas estruturais não-lineares, em geral, essas aplicações se restringem a casos 2D com malhas estruturadas (frequentemente usando elementos quadrilaterais). Isso demonstra, de certa forma, o nível de dificuldade existente para se manipular tanto teórica quanto computacionalmente situações envolvendo discretizações p em problemas tridimensionais (3D) com malhas não-estruturadas.

A Figura 1.1 ilustra a solução de um problema de elasticidade não-linear discretizado com elementos tetraédricos quadráticos do tipo h . Essa solução foi obtida a partir do programa de uso comercial ANSYS 6.0 o qual não permite o tratamento desse tipo de problema com base na formulação p .

Tanto o tratamento 2D quanto o 3D da versão- p exigem proficiência em vários aspectos teórico-computacionais que o distinguem da versão- h clássica do MEF, tais como:

1. integração numérica de alta ordem;
2. diferenciação numérica (particularmente em malhas 3D);
3. seleção de funções de forma adequadas;
4. mapeamento geométrico para domínios arbitrários;

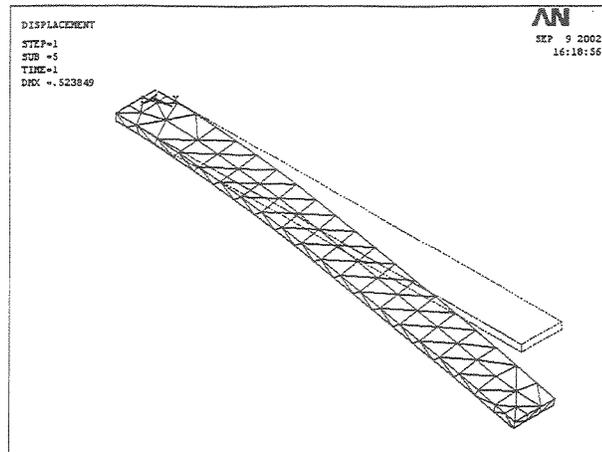


Figura 1.1: Exemplo de um problema estrutural não-linear discretizado com uma malha de elementos tetraédricos a partir do programa comercial ANSYS 6.0. Para esse tipo de problema, a formulação p não está disponível.

5. imposição de continuidade da solução global nas regiões inter-elementos;
6. numeração de graus de liberdade;
7. aplicação de condições de contorno;
8. pós-processamento de resultados.

Entre os itens listados, os de número 1, 3 e 4 merecem atenção especial tanto do ponto de vista da formulação quanto da implementação da versão- p . Particularmente, esses itens constituem a base para se obter soluções a partir do processo de discretização. Na versão- p , assim como na versão- h , requisitos mínimos são exigidos dos procedimentos de integração numérica (Banerjee e Suri, 1992; Kim e Suri, 1993), das funções de forma (Babuška et al., 1981) e do mapeamento geométrico (particularmente no caso de domínios curvos) (Babuška e Guo, 1988; Ciarlet e Raviart, 1972; Lenoir, 1986; Bernardi, 1989) para que se garanta a convergência da aproximação. Além disso, a escolha das funções de forma assume um papel fundamental na eficiência e flexibilidade do método (Webb e Abouchakra, 1995; Nogueira Jr. e Bittencourt, 2001b).

Ao se optar por uma aproximação de alta ordem, além do interesse pelo bom desempenho dos itens salientados acima, outro aspecto essencial na busca por performance diz respeito à rapidez com que se pode obter soluções uma vez tendo sido produzidos os sistemas algébricos lineares resultantes do processo de discretização. Esse aspecto torna-se ainda mais relevante quando se

considera a solução de problemas de aplicação prática em que se manipulam dezenas ou até centenas de milhares de graus de liberdade. Uma das técnicas mais eficientes para se tratar esse tipo de obstáculo, no caso da versão- h , é o uso de métodos multigrid (Brandt, 1977; Hackbush, 1985; McCormick, 1987; Briggs, 1987; Bittencourt, 1996). Esse tipo de método aproveita as vantagens das técnicas de relaxação clássicas como *Gauss-Seidel* e *Jordan* para reduzir rapidamente as altas frequências dos erros de aproximação a cada iteração. Essa característica associada a utilização de diferentes malhas (ou níveis) produz um decaimento global do erro proveniente do processo iterativo de solução. A limitação dessa abordagem, sobretudo em casos **3D** com malhas não-estruturadas, reside na dificuldade de se gerar uma sequência de malhas para o domínio do problema estudado. Essa dificuldade é inerente à discretização do tipo h mas não à do tipo p uma vez que neste último caso os espaços de aproximação são naturalmente aninhados não dependendo da estrutura física da malha. Esse fato tem estimulado a busca por métodos multigrid algébricos que se baseiam na versão- p (Hu e Katz, 1995; Hu et al., 1997).

Em geral, as análises sobre métodos multigrid algébricos para a versão- p se concentram na estimativa teórica de sua taxa de convergência bem como na eficiência computacional dos algoritmos envolvidos. No caso da eficiência computacional, particularmente no que se refere ao uso de malhas não-estruturadas, os estudos realizados até o momento (Hu e Katz, 1995; Hu et al., 1997) não estabelecem a superioridade desse método sobre algumas técnicas iterativas comumente empregadas na versão- p como as de gradiente conjugado pré-condicionado (Babuška et al., 1991; Mandel, 1990d; Mandel, 1990b; Mandel, 1990a; Mandel, 1994). Desse modo, um estudo sobre a eficiência e aplicabilidade desse tipo de método a problemas com malhas não-estruturadas ainda está em aberto.

A partir do que foi exposto, pretende-se definir dois focos principais para o direcionamento do presente trabalho. O primeiro está relacionado a escolha de uma formulação adequada para a aplicação da versão- p do **MEF** a problemas de elasticidade linear e não-linear em malhas não-estruturadas **3D**. O segundo, diz respeito a aplicação de técnicas multigrid algébricas na solução de sistemas lineares oriundos da discretização de problemas através desse tipo de formulação.

A escolha pelo método p pode ser justificada de várias maneiras, entre elas, talvez a mais importante, seja a taxa de convergência exponencial alcançada por essa formulação para uma vasta classe de problemas. O uso de malhas não-estruturadas (elementos triangulares e tetraédricos) está

relacionado à obtenção de maior flexibilidade na geração de partições sobre domínios arbitrários. O emprego de técnicas multigrid para a versão- p foi motivado pela notória eficiência demonstrada pelas técnicas multigrid clássicas, quando aplicadas à formulação h , associada a propriedade natural de aninhamento dos espaços de aproximação p (espaços aninhados possibilitam a definição de vários níveis de iteração a partir de uma única malha física para o problema). Finalmente, a opção particular por geometrias **3D** aliada à solução de problemas de elasticidade linear, utilizando-se métodos multigrid algébricos, e de problemas de elasticidade não-linear, ambos em malhas não-estruturadas, confere originalidade à presente proposta.

A próxima seção apresenta uma revisão bibliográfica detalhada dos principais temas associados à versão- p do **MEF**. A seguir, destacam-se os objetivos do trabalho, suas contribuições, além de uma breve exposição a respeito da organização dos capítulos subsequentes.

1.2 Revisão Bibliográfica

1.2.1 Breve histórico sobre a evolução da versão- p do **MEF**

Os primeiros passos no sentido de se desenvolver uma versão- p para o **MEF** datam de antes de 1970. (Zienkiewicz et al., 1970) propuseram o uso de elementos de ordem polinomial variável (denominados à época de elementos mistos) em problemas de elasticidade **3D** como uma maneira eficiente de se adicionar graus de liberdade extras em regiões com elevados gradientes. Em lugar de se criar novos nós na malha de discretização, variáveis associadas às arestas dos elementos foram usadas para acomodar uma interpolação de alta ordem. Ainda nesse trabalho, foi introduzido o conceito de refinamento hierárquico o qual ocupa uma posição central tanto na formulação quanto na implementação do método p .

(Peano, 1975; Peano, 1976) apresentou algumas famílias de funções de interpolação hierárquicas para elementos triangulares (com arestas retilíneas) que, pelo menos em teoria, poderiam gerar elementos de qualquer ordem polinomial. Nesse caso, cada grau de liberdade adicionado, para $p \geq 2$, correspondia a p -ésima derivada da aproximação corrente. Essas funções de interpolação eram mapeadas diretamente sobre os elementos da malha física do problema utilizando-se coordenadas de área (ou baricêntricas).

(Katz, 1978) e (Rossow e Katz, 1993) estenderam o trabalho de *Peano* incluindo a idéia de *elemento de referência* com o objetivo de melhorar a eficiência computacional do método. Um mapeamento geométrico para elementos triangulares com arestas retilíneas também foi apresentado. Foi sugerido ainda o uso de estruturas (matrizes e vetores) pré-computadas, novamente visando o aumento da eficiência computacional. Além disso, foi observada a vantagem computacional de se utilizar espaços hierárquicos que, por serem aninhados, dispensam o cálculo completo das matrizes locais numa sucessão de níveis de aproximação p .

(Zienkiewicz et al., 1981) retomaram as funções de forma apresentadas em 1970 e mostraram como elementos quadrilaterais hierárquicos poderiam ser formulados. As vantagens dessa abordagem estavam relacionadas ao bom condicionamento das matrizes geradas, à facilidade na compatibilização da continuidade da função de aproximação global ao se empregar malhas com ordens polinomiais não-uniformes e à introdução de estimadores de erro.

(Babuška et al., 1981) apresentaram pela primeira vez, num formalismo matemático rigoroso, as bases do método p . Essa publicação, que se tornou a principal referência no assunto, mostrou que a convergência desse tipo de método, quando $p_{\min} \rightarrow \infty$, estaria assegurada desde que as funções de interpolação do espaço de aproximação fossem adequadamente construídas. Uma estimativa *a priori* para a taxa de convergência assintótica desse método também foi sugerida. Essa estimativa mostrou que, mesmo na presença de singularidades (desde que posicionadas sobre os nós físicos da malha), a taxa de convergência do método- p era o dobro daquela apresentada pelo método- h quando se empregavam malhas quasi-uniformes.

(Szabó e Babuška, 1991) publicaram o primeiro livro inteiramente devotado ao método p popularizando termos como versão- p (direcionada aos aspectos de implementação), extensão- p (direcionada aos aspectos de estimativa e controle de erros) e convergência- p (direcionada aos aspectos de convergência do método). Essa obra contém uma investigação detalhada do método p 1D e 2D com capítulos de aplicações em elasticidade 2D e 3D, escoamento potencial 2D e formulação clássica de elementos de placa e casca.

(Karniadakis e Sherwin, 1999) realizaram um trabalho pioneiro ao unificarem as abordagens hp e espectral do MEF num livro dedicado à solução de problemas de mecânica dos fluidos computacional em malhas não-estruturadas. Nesse trabalho, os pontos fortes do método espectral multi-domínios (Patera, 1984) e da versão- hp (Szabó e Babuška, 1991) de alta ordem foram apre-

sentados numa estrutura matemática rigorosa e com um forte apelo computacional. Do ponto de vista da implementação, esse trabalho se esmerou em fornecer e detalhar os aspectos chaves para se obter a máxima eficiência numérica nas aproximações de alta ordem.

1.2.2 Funções de forma

A qualidade das soluções obtidas através do **MEF** depende de vários fatores, entre eles, o tamanho e a forma dos elementos, as propriedades do espaço de aproximação e a regularidade da solução procurada. Do ponto de vista computacional, a escolha de uma base para o espaço de aproximação é crítica no que se refere à estabilidade e eficiência dos procedimentos envolvidos na discretização. Por razões de simplicidade, o espaço de aproximação geralmente consiste de funções polinomiais por partes definidas sobre os elementos de uma partição que preenche o domínio do problema. Bases constituídas por funções polinomiais por partes são usualmente construídas por: (i) uma transformação que leva elementos K^e da malha física do problema a um elemento de referência \hat{K} através de uma função de mapeamento suave e geralmente bijetiva e (ii) um conjunto de funções de forma $\{\hat{\phi}_i\}_{i=1}^{n_p}$ que gera uma base para o espaço de polinômios $S^p(\hat{K})$ de ordem não superior a p definido sobre o elemento de referência \hat{K} . Corriqueiramente, costuma-se utilizar o termo elemento finito em substituição à definição matemática rigorosa fornecida pelos itens (i) e (ii). É usual, ainda, empregar os termos modos, funções de forma, funções de interpolação ou funções de base como sinônimos.

Os elementos mais simples empregados no **MEF** utilizam funções polinomiais de baixa ordem definidas num espaço de coordenadas local. Elementos mais elaborados usam funções de forma constituídas por polinômios de alta ordem. Entre os elementos de alta ordem, aqueles que preservam a continuidade da função de aproximação ao longo de malhas com ordem polinomial não-uniforme são conhecidos como elementos *hierárquicos* (Zienkiewicz et al., 1989). O conceito de hierarquia (Zienkiewicz et al., 1983; Zienkiewicz e Craig, 1986; Babuška et al., 1981) fundamenta-se na propriedade de que o conjunto de funções de base que gera o espaço de aproximação de uma dada ordem p contém integralmente o conjunto de funções de base do espaço de aproximação de ordem $p - 1$.

As funções de forma hierárquicas tipicamente empregadas na formulação p do **MEF** são comumente associadas a entidades topológicas dos elementos como vértices, arestas, faces e interiores. O tratamento clássico dessas funções envolve geralmente o uso de polinômios unidimensionais de

Legendre ou Chebyshev (Szabó e Babuška, 1991; Vijayakar et al., 1987; Devloo et al., 1988). Os elementos mais comuns empregados nas formulações hierárquicas têm sido os de formato quadrilateral ou hexaédrico uma vez que a utilização de malhas estruturadas é bastante popular. Nesse caso, é usual construir as funções hierárquicas a partir do produto tensorial de funções polinomiais unidimensionais (Edgar e Surana, 1996). A utilização do produto tensorial na obtenção de funções hierárquicas para domínios triangulares ou tetraédricos não é imediata necessitando de uma abordagem especial.

(Sherwin e Karniadakis, 1995) propuseram um conjunto de funções hierárquicas para triângulos e tetraedros baseado em coordenadas cartesianas que preserva as propriedades do produto tensorial. As funções de base sugeridas nessa formulação empregam polinômios de *Jacobi* de peso misto e acomodam uma integração numérica exata usando apenas a tensorização das regras de quadratura unidimensionais de *Gauss-Jacobi* (Karniadakis e Sherwin, 1999). A vantagem de não requerer procedimentos especiais de integração numérica decorre da utilização de um conjunto de transformações de coordenadas que permite definir domínios triangulares e tetraédricos a partir de sistemas de coordenadas retangulares e hexaédricos, respectivamente (Sherwin e Karniadakis, 1995).

(Webb e Abouchakra, 1995) descreveram um elemento triangular hierárquico que também utiliza os polinômios de *Jacobi* para gerar as funções de interpolação. Esse desenvolvimento adota o elemento de referência $[0,1]$ 2-Simplex definido em coordenadas baricêntricas e emprega normalmente a regra de quadratura para elementos triangulares elaborada em (Dunavant, 1985). Uma extensão desse elemento para o caso de malhas tetraédricas foi sugerida em (Abouchakra, 1996). Nesse caso, as matrizes locais geradas pelo elemento tetraédrico foram construídas com base no método de matrizes pré-computadas ou matrizes universais (um tipo particular de soma ponderada de matrizes) que dispensa o uso de regras de quadratura.

Além das considerações com respeito ao tipo de malha empregada e ao uso de coordenadas baricêntricas (e.g., funções não-tensorisáveis em malhas não-estruturadas), um aspecto essencial na escolha das funções de forma para a versão- p é a busca por boas propriedades de condicionamento e esparsidade das matrizes locais (rigidez e massa) associadas ao elemento (Babuška et al., 1989; Carnevali et al., 1993; Edgar e Surana, 1996; Szabó e Peano, 1987). Os padrões de esparsidade e condicionamento das matrizes locais são determinados pela definição das funções de forma e indiretamente induzem esses mesmos padrões nas matrizes globais (Carnevali et al., 1993; Edgar e

Surana, 1996; Zumbusch, 1995). O bom condicionamento das matrizes globais demonstra-se particularmente importante quando se empregam aproximações de alta ordem e/ou métodos iterativos na solução dos problemas algébricos decorrentes da discretização por elementos finitos. No caso do uso de métodos iterativos, o bom condicionamento resulta na redução do número de iterações. Da mesma maneira, a esparsidade dessas matrizes resulta num menor custo por iteração (Carnevali et al., 1993).

As funções hierárquicas clássicas para elementos quadrilaterais e hexaédricos propostas em (Szabó e Babuška, 1991) apresentam excelentes propriedades quanto à esparsidade e ao condicionamento (Edgar e Surana, 1996; Maitre e Pourquier, 1995). Tais características decorrem diretamente das propriedades das integrais dos polinômios de *Legendre* e do caráter tensorial dessas funções. No entanto, para elementos triangulares e tetraédricos, as funções sugeridas por estes mesmos autores (Szabó e Babuška, 1991) não exibem propriedades tão boas provocando um crescimento exponencial do número de condição das matrizes locais com o aumento da ordem polinomial p (Carnevali et al., 1993; Adjerid et al., 2001).

(Carnevali et al., 1993) apresentaram um conjunto de funções de base hierárquicas para triângulos e tetraedros construído de modo que as funções de aresta, face e interior de ordem p fossem ortogonais (com relação ao operador de Laplace) a essas mesmas funções de ordens não superiores a $p - 2$, $p - 3$ e $p - 4$, respectivamente. Essa particularidade resultou na obtenção de matrizes de rigidez locais com maior grau de esparsidade e melhor condicionamento que as matrizes associadas às funções hierárquicas propostas em (Szabó e Babuška, 1991). Esse conjunto de funções de base demonstrou ainda conservar muitos termos nulos nas matrizes de rigidez globais mesmo quando se empregavam funções de mapeamento não-lineares (Carnevali et al., 1993).

(Nogueira Jr. e Bittencourt, 1999; Nogueira Jr. e Bittencourt, 2001b) demonstraram a vantagem, tanto do ponto de vista do condicionamento quanto da esparsidade, do uso dos polinômios de *Jacobi* na construção de funções de base para malhas não-estruturadas. A flexibilidade gerada pelo emprego de funções multi-dimensionais tensorizadas, no que se refere aos procedimentos de integração numérica, também foi ressaltada nesses trabalhos. Observou-se ainda que as funções de base propostas em (Carnevali et al., 1993) ainda resultavam num crescimento exponencial do condicionamento local ao se elevar a ordem polinomial p embora esse crescimento fosse inferior ao das funções propostas em (Szabó e Babuška, 1991).

Mais recentemente, (Adjerid et al., 2001) propuseram novas bases para triângulos e tetraedros que apresentaram melhores padrões de esparsidade e condicionamento que as respectivas funções hierárquicas sugeridas em (Szabó e Babuška, 1991) e (Carnevali et al., 1993). Nesse caso, a construção das bases consiste na modificação da base desenvolvida em (Szabó e Babuška, 1991) pela ortogonalização das funções associadas às faces, no caso **2D**, e às faces e interiores, no caso **3D**. Essa estratégia de ortogonalização parcial ou completa de funções de base é bastante similar àquela adotada nos trabalhos de (Mandel, 1990d; Mandel, 1990a) para se criar pré-condicionadores baseados na técnica de decomposição de domínios em malhas de elementos hexaédricos.

1.2.3 Mapeamento geométrico

A quantidade de elementos necessários para se atingir uma precisão desejada numa malha com discretização p é sensivelmente menor que aquela utilizada numa malha com discretização h . Em decorrência disso, boa parte dos elementos empregados em malhas p possuem tamanhos relativamente extensos. Por isso, o uso de técnicas de mapeamento são muito mais importantes para a versão- p do que para a versão- h . Nesse último caso, a precisão da representação geométrica cresce com o processo de convergência do método uma vez que o diâmetro do maior elemento da malha tende a zero a medida que o número de graus de liberdade vai a infinito.

Na versão- p , é possível representar domínios arbitrários utilizando-se apenas as funções de base nodais tipicamente empregadas na versão- h . Nesse caso, a representação de contornos curvos é aproximada por funções polinomiais de ordem fixa, em geral, polinômios de *Lagrange* quadráticos (Szabó e Babuška, 1991). No entanto, existe outro tipo de mapeamento geométrico muito mais apurado que é especialmente indicado no caso de discretizações do tipo p . Esse tipo de mapeamento recebe o nome de método de *blending-functions*.

O método de *blending-functions* foi originalmente desenvolvido por (Coons, 1964) com o objetivo de se construir superfícies curvas arbitrárias para uso em aplicações de desenho assistido por computador (**CAD**). (Gordon, 1971) estendeu esse trabalho apresentando uma nova classe de métodos para a aproximação de funções de múltiplas variáveis. Esses métodos foram desenvolvidos com o objetivo de se representar complexas formas geométricas em ambientes **CAD** e em aplicações de controle numérico de manufatura (**CAM**).

A idéia básica por trás desses métodos é usar determinados tipos de funções (*blending-functions*) como meio de aproximação. As funções originalmente utilizadas em (Coons, 1964) foram as funções cardinais para a interpolação de *Hermite* em dois pontos extremos por polinômios de grau ímpar. (Gordon, 1971) utilizou uma variedade de funções em seu trabalho incluindo polinômios de *Hermite* e de *Lagrange*. (Gordon e Hall, 1973a; Gordon e Hall, 1973b) mostraram ainda como o método de *blending-functions* pode ser usado para se construir interpolações sobre domínios curvos arbitrários e definir sistemas de coordenadas curvilíneas para aplicação na geração de malhas.

(Sherwin e Karniadakis, 1996) propuseram um tipo de mapeamento isoparamétrico de alta ordem para malhas 3D não-estruturadas baseado na transformação local (*modificada*) das funções de forma associadas às faces dos elementos de fronteira. Esse tipo de mapeamento adota como ponto de partida a continuidade C^0 da representação global do contorno do domínio. A transformação *modificada* por sua vez consiste na solução de problemas de interpolação locais baseados na projeção de *Galerkin*. (Karniadakis e Sherwin, 1999) mostraram ainda que esse tipo de técnica de mapeamento, quando empregada em malhas estruturadas com elementos quadrilaterais ou hexaédricos, é equivalente ao método de *blending-functions*.

(Dey et al., 1997) apresentaram um esquema de mapeamento para malhas 3D não-estruturadas baseado na representação exata do contorno do domínio de discretização tal como descrito pelo modelador geométrico de pré-processamento empregado. Os resultados numéricos apresentados revelaram, no entanto, que o mapeamento geométrico pela aproximação polinomial isoparamétrica em pontos de interpolação ótimos (Chen e Babuška, 1995; Hesthaven, 1998) ainda era superior ao esquema proposto em termos da convergência exponencial da aproximação do tipo p .

1.2.4 Técnicas de solução para sistemas lineares

O emprego efetivo da versão- p a problemas de aplicação prática requer o uso de algoritmos eficientes na resolução do sistema de equações lineares resultante do processo de discretização. Na maioria dos problemas 2D, os métodos diretos de solução de sistemas lineares são superiores, em termos de desempenho, aos métodos iterativos (Babuška e Helman, 1989). Entretanto, no caso 3D, o tempo de solução e a capacidade de armazenamento requeridos pelos métodos diretos podem se tornar exageradamente custosos tornando os métodos iterativos muito mais atraentes (Mandel, 1990d; Foresti et al., 1991; Bittencourt, 1996). Entre os métodos iterativos, os de gradiente

conjugado preconditionado (**GCP**) (Golub e O’Leary, 1989; Golub e Loan, 1989; Hestenes e Stiefel, 1952) têm sido considerados os mais apropriados para a solução de sistemas de equações lineares oriundos da versão- p do **MEF** (Hu et al., 1997). Nesse sentido, muito esforço tem sido aplicado na elaboração de pré-condicionadores para essa versão.

Técnicas de pré-condicionamento para a versão- p utilizando métodos diretos foram propostas em (Foresti et al., 1991). Nessa abordagem, emprega-se a fatoração direta das matrizes correspondentes às ordens polinomiais baixas como pré-condicionador para as matrizes de ordens polinomiais mais altas. (Kapotin et al., 1991) aplicaram variantes aprimoradas do clássico pré-condicionador **SSOR** à versão- p , tais como bloco **SSOR** e bloco incompleto **SSOR**, tendo obtido sucesso.

Uma vasta classe de pré-condicionadores para a versão- p baseados na estrutura topológica das matrizes de rigidez e relacionados às técnicas de decomposição de domínios foram exaustivamente estudados tanto do ponto de vista teórico quanto numérico (Babuška et al., 1989; Babuška et al., 1991; Mandel, 1990d; Mandel, 1990b; Mandel, 1994; Casarin, 1997). A idéia básica usada na construção desses tipos de pré-condicionadores é condensar os modos internos dos elementos e usar funções de forma de baixa ordem (geralmente lineares ou quadráticas) para compor a matriz de pré-condicionamento. O processo de condensação, nesse caso, é normalmente efetuado computando-se o complemento de Schur em cada elemento separadamente favorecendo assim a paralelização do método (Babuška et al., 1989; Mandel, 1990c; Casarin, 1997; Korneev e Jensen, 1997). Várias estruturas bloco diagonais podem ser derivadas desse procedimento gerando uma série de variantes nessa categoria de pré-condicionadores (Mandel, 1990a; Korneev e Jensen, 1999). No caso de problemas **2D** usando malhas de elementos quadrilaterais ou triangulares, é possível mostrar que, para grandes valores de p , o número de condição das matrizes pré-condicionadas deteriora-se na pior das hipóteses proporcionalmente a $\log^2 p$ (Babuška et al., 1991). Esse resultado é análogo ao que foi obtido para a versão- h em (Bramble et al., 1986; Bramble et al., 1989). Não obstante a ausência de resultados teóricos similares para o caso **3D**, experimentos numéricos demonstram que os números de condição gerados por esses pré-condicionadores não crescem significativamente com o aumento da ordem polinomial p (Mandel, 1990d; Mandel, 1994).

Ainda dentro da classe de pré-condicionadores baseados na técnica de decomposição de domínios, é relevante mencionar os métodos de superposição como o Método de *Schwarz* Aditivo (**MSA**) proposto em (Pavarino, 1994a; Pavarino, 1994b) e os de subestruturação propostos em (Pavarino e

Widlund, 1996; Pavarino e Widlund, 1997). No primeiro caso, para malhas estruturadas **2D** e **3D**, demonstra-se que o número de condição associado ao operador iterativo **MSA** é limitado por uma constante que independe da ordem polinomial p e do número de subdomínios N que particionam o domínio global do problema (Pavarino, 1994a). No segundo caso, demonstra-se que o número de condição das matrizes pré-condicionadas cresce a uma taxa polilogarítmica (i.e., do tipo $\log^2 p$) com o aumento da ordem polinomial p .

(Cao e Guo, 2002) propuseram uma modificação ao trabalho desenvolvido por (Pavarino e Widlund, 1997) ao resolverem de forma inexata os procedimentos de desacoplamento entre as funções de base globais associadas ao grupo vértices-arestas e aquelas associadas às faces dos elementos bem como entre as funções associadas às faces dos diferentes elementos entre si. Essa modificação resultou na redução do custo de construção do pré-condicionador sem afetar a estimativa polilogarítmica de crescimento do número de condição do sistema pré-condicionado com a ordem polinomial p .

(Sherwin e Casarin, 2001) também empregaram a técnica de subestruturação na obtenção de um pré-condicionador para a solução da equação de *Navier-Stokes* incompressível em malhas **3D** não-estruturadas. De forma similar a abordagem utilizada em (Mandel, 1990d) para malhas de elementos hexaédricos, essa técnica se baseia na obtenção de um conjunto de funções de base de *baixa energia* a partir de uma transformação sobre as funções de base usadas na discretização do problema. Essa nova base, derivada numericamente, reduz o acentuado acoplamento entre os modos da base original tornando-a adequada a um pré-condicionamento bloco-diagonal. A aplicação do operador **MSA** por bloco à base de *baixa energia* combinada a um método de solução direto para fatorar o bloco associado às funções de vértice originais constitui o pré-condicionador propriamente dito. Resultados numéricos obtidos a partir dessa técnica revelaram um ganho de velocidade de até três vezes em relação ao pré-condicionamento diagonal padrão para o caso de malhas tetraédricas.

Outra classe de métodos iterativos que merece amplo destaque é a de métodos multi-níveis. No caso da versão- h do **MEF**, os métodos multigrid clássicos alcançaram um grande sucesso (Bittencourt, 1996; Bittencourt e Feijóo, 1997; Bittencourt e Feijóo, 1998; Bittencourt et al., 1998a; Bittencourt et al., 1998b). Motivados pelo desempenho dos métodos multigrid, alguns autores decidiram aplicar tais idéias na solução de sistemas de equações algébricas decorrentes da aplicação da versão- p (Craig e Zienkiewicz, 1985).

(Babuška et al., 1989) conduziram uma série de experimentos numéricos nessa linha propondo

originalmente a denominação *métodos multi- p* . Essa abordagem consiste na relaxação sucessiva dos graus de liberdade não nodais ao se variar os níveis p . Baseado nos resultados das simulações numéricas para problemas elípticos **2D** em malhas estruturadas, concluiu-se que o ciclo **V** dos métodos multi- p convergiam mais rapidamente do que os métodos iterativos clássicos de *Gauss-Seidel* e **SOR** porém, ainda eram mais lentos que os métodos multigrid clássicos. Os mesmos resultados foram confirmados em (Brussino et al., 1989) para uma arquitetura computacional paralela.

(Foresti et al., 1989) testaram um tipo de algoritmo multi- p empregando ciclos **V** em que o método bloco *Gauss-Seidel* foi usado como esquema de relaxação. Os resultados numéricos desse teste demonstraram-se superiores aos anteriores mas ainda apresentaram uma taxa de convergência lenta, mesmo em ordens polinomiais moderadas.

(Hu e Katz, 1995) investigaram de maneira sistemática os métodos multi- p estabelecendo as primeiras bases teóricas para essa metodologia. Desenvolveu-se nesse trabalho a classe dos métodos multi- p algébricos de ciclo **V**, incluindo os ciclos **V** padrões, modificados e variáveis. Métodos multi- p aninhados e a combinação destes com os algoritmos de ciclo **V** também foram estudados tendo essa última associação, denominada algoritmos de ciclo **V** acelerados, demonstrado a maior eficiência entre as variantes consideradas. Além disso, foram conduzidas investigações sobre a convergência dos métodos multi- p tanto teórica quanto numericamente. Nesse caso, as aplicações numéricas estiveram restritas a problemas de *Poisson* em malhas **2D** estruturadas.

(Hu et al., 1997) estenderam o conceito de métodos multi- p algébricos com o objetivo de incorporar a idéia de pré-condicionadores multi- p . Nesse caso, os pré-condicionadores foram aplicados com o uso de elementos estaticamente condensados além do uso de um esquema de relaxação de *Gauss-Seidel* simétrico com o intuito de se garantir a convergência do método. Experimentos numéricos envolvendo problemas de elasticidade linear em malhas **2D** e **3D** estruturadas revelaram que os números de condição, após o pré-condicionamento multi- p , eram independentes da ordem polinomial p .

(Guo e Katz, 1998), visando melhorar a performance dos métodos multi- p , propuseram a substituição do esquema de relaxação de *Gauss-Seidel* pelo de *Jacobi* amortecido priorizando a minimização do esforço computacional em detrimento da velocidade de convergência. (Guo e Katz, 2000) ainda estenderam a aplicação dos métodos multi- p para o caso de uma arquitetura computa-

cional paralela.

Recentemente, (Nogueira Jr. e Bittencourt, 2001a; Nogueira Jr. e Bittencourt, 2002) investigaram, em termos do custo computacional, a aplicação dos métodos multi- p a problemas elípticos lineares (equação de *Poisson* e equações de elasticidade) em malhas **2D** e **3D** não-estruturadas. Para o tipo de análise adotada, os resultados obtidos demonstraram que a performance dos esquemas multi- p é sempre inferior àquela exibida pelo método de gradientes conjugados com pré-condicionamento de *Gauss-Seidel* simetrizado. O clássico pré-condicionador proposto por (Babuška et al., 1991) para problemas **2D** também se mostrou surpreendentemente inferior a este último. No entanto, foi possível notar que na maioria dos casos **3D** os métodos multi- p apresentaram taxas de crescimento do número de operações executadas em relação à quantidade de graus de liberdade envolvida menor que a dos demais métodos estudados.

1.3 Objetivos

Este trabalho tem como um de seus objetivos centrais a formulação e implementação (em linguagem C++) da versão- p do **MEF** para problemas elásticos lineares e não-lineares em malhas **3D** não-estruturadas. Esse tema pretende dar continuidade ao trabalho relacionado a malhas não-estruturadas do tipo h desenvolvido em (Bittencourt, 1996). Para se concretizar tal tarefa, pretende-se inicialmente conduzir um estudo detalhado do comportamento teórico-numérico das funções de base **2D** e **3D** de alta ordem usualmente utilizadas. Além disso, pretende-se identificar os elementos básicos necessários à implementação de uma estrutura de código baseada na versão- p **3D** do **MEF**. A implementação dos aspectos essenciais dessa formulação tomará por base o trabalho desenvolvido por (Karniadakis e Sherwin, 1999). *A priori*, planejou-se implementar um código que contemplasse apenas domínios sem curvatura (retilíneos ou planos por partes) ou, melhor dizendo, que pudessem ser exatamente descritos por mapeamentos afins. Desse modo, o quarto item enumerado na primeira seção (mapeamento geométrico para domínios arbitrários) não receberá nenhum enfoque especial neste trabalho.

Como segundo objetivo básico, destaca-se o estudo e desenvolvimento de técnicas multigrid algébricas para a versão- p do **MEF** quando aplicadas a problemas elípticos lineares. Dentro desse tema, com a intenção de se nortear a busca de desempenho para problemas de elasticidade **3D** em

malhas não-estruturadas, realiza-se um estudo prévio da aplicação desse tipo de técnica com base na equação de *Poisson* 2D. Esse estudo preliminar envolve a comparação de diferentes métodos iterativos de solução de sistemas lineares incluindo técnicas de pré-condicionamento para gradientes conjugados e análise *hp* adaptável. Uma vez concluído o estudo preliminar, aplicam-se as técnicas multi-*p* em problemas 3D.

Para a solução dos exemplos 2D estudados, será utilizada a estrutura de classes, desenvolvida com base na versão-*p*, gentilmente cedida por *A.A. Novotny* (LNCC - Rio de Janeiro). Essa base de códigos foi totalmente escrita em linguagem C++ tendo revelado aspectos chave para a futura implementação da formulação *p* 3D adotada.

1.4 Contribuições do Trabalho

Como contribuições resultantes do desenvolvimento do presente trabalho de doutorado, pode-se ressaltar os seguintes itens:

- Estudo comparativo entre os conjuntos de funções de base hierárquicas associados a elementos triangulares e tetraédricos mais comumente empregados na literatura (Capítulo 2);
- Desenvolvimento de conjuntos de funções de base hierárquicas para elementos triangulares e tetraédricos projetados de forma a se obter ortogonalidade total entre os modos internos associados aos elementos (Capítulo 2);
- Implementação e aplicação da formulação *p* do MEF, originariamente proposta por (Karniadakis e Sherwin, 1999) para problemas de mecânica dos fluidos computacional em malhas não-estruturadas, na solução de problemas de elasticidade linear e não-linear 3D (Capítulo 3);
- Formulação e implementação de métodos multi-*p* algébricos para problemas elípticos lineares em malhas 2D e 3D não-estruturadas com base no trabalho desenvolvido por (Hu e Katz, 1995) para malhas estruturadas (Capítulo 4);
- Desenvolvimento de algoritmos generalizados para os ciclos usualmente utilizados nos métodos multi-*p* (Capítulo 4);

- Desenvolvimento de um pré-condicionador para problemas elípticos lineares **2D** baseado no pré-condicionador modelo para formulações p proposto por (Babuška et al., 1991) (Capítulo 4);
- Utilização das técnicas de adaptabilidade hp para problemas elípticos lineares em malhas **2D** não-estruturadas propostas por (Novotny et al., 1999) para o desenvolvimento de uma estratégia multi- hp adaptável (Capítulo 4).
- Desenvolvimento de procedimentos de discretização para a versão- p do **MEF** a partir da base de programas construída em (Silva, 2002) (Capítulo 5).

1.5 Organização do Texto

Os capítulos subsequentes deste trabalho estão organizados da seguinte maneira: o Capítulo 2 é dedicado ao estudo de funções de base hierárquicas para elementos triangulares e tetrédricos; o Capítulo 3 apresenta os elementos essenciais da formulação p com base no trabalho desenvolvido em (Karniadakis e Sherwin, 1999) para problemas de mecânica dos fluidos computacional. Esse capítulo cobre ainda a aplicação dessa formulação a problemas de elasticidade linear e não-linear em malhas **3D** não-estruturadas; o Capítulo 4 traz um estudo sobre a formulação e aplicação de métodos iterativos e multigrid algébricos para a versão- p do **MEF** aplicados a problemas lineares **2D** e **3D** em malhas não-estruturadas; o Capítulo 5 salienta os aspectos computacionais necessários à implementação, em linguagem C++, de um código baseado na versão- p **3D** do **MEF**; por fim, o Capítulo 6 sintetiza os principais resultados deste trabalho apresentando suas conclusões e perspectivas futuras.

Capítulo 2

Funções de Base Hierárquicas

Como salientado no capítulo introdutório, a escolha das funções de base na versão- p desempenha um papel fundamental na determinação da eficiência e flexibilidade do método de discretização. Foi observado também que a propriedade hierárquica dos espaços de aproximação p permite que as matrizes locais não sejam totalmente reconstruídas quando se aumenta a ordem polinomial desses espaços. Essa propriedade bem como a de ortogonalidade entre as funções de base, a qual resulta em matrizes de rigidez globais acentuadamente esparsas e bem condicionadas, é bastante desejável sob o ponto de vista de implementação.

O presente capítulo realiza um estudo comparativo entre funções de base hierárquicas associadas a triângulos e tetraedros com o objetivo de selecionar o conjunto de funções mais adequado para uso de malhas não-estruturadas. Discute-se, a partir da análise das propriedades de esparsidade e condicionamento locais, o desempenho de vários conjuntos de funções sugeridos na literatura (Abouchakra, 1996; Carnevali et al., 1993; Sherwin e Karniadakis, 1995; Szabó e Babuška, 1991; Webb e Abouchakra, 1995). Com base na formulação desenvolvida em (Sherwin e Karniadakis, 1995), apresenta-se uma nova proposta para funções hierárquicas associadas a triângulos e tetraedros. Esse conjunto de funções, construído a partir do produto tensorial generalizado de polinômios de *Jacobi*, foi projetado para gerar ortogonalidade total entre os modos internos associados ao elemento de maneira a produzir matrizes de massa locais com elevado padrão de esparsidade.

Na seção 1, são apresentadas as bases hierárquicas bi e tridimensionais consideradas. A seção 2 apresenta resultados teóricos sobre a ortogonalidade das novas funções de base sugeridas. As seções

3, 4 e 5 trazem resultados numéricos comparando as propriedades de esparsidade e condicionamento das matrizes locais (rigidez e massa) associadas aos diferentes grupos de funções hierárquicas definidos sobre triângulos e tetraedros. A obtenção desses resultados foi realizada com o auxílio do software de manipulação simbólica MATHEMATICA 3.0 o que tornou possível a visualização dos padrões de esparsidade gerados em cada caso. A seção 6 é dedicada à validação, em termos globais, dos resultados relativos ao condicionamento das matrizes locais. A análise do comportamento das funções de base do ponto de vista global se estende para a seção 7 na qual se considera a solução de problemas simples baseados na equação de *Poisson 2D* através do método de gradientes conjugados. As conclusões a respeito das investigações realizadas são apresentadas na seção 8.

2.1 Elementos Hierárquicos para Malhas Estruturadas e Não-estruturadas

2.1.1 Quadriláteros

As funções hierárquicas associadas a quadriláteros sugeridas em (Szabó e Babuška, 1991) são definidas sobre o elemento de referência $\hat{Q} = \{(\xi, \eta) \in \mathbb{R}^2 \mid |\xi| < 1, |\eta| < 1\}$ (Figura 2.1(a)) e construídas com base na integração dos polinômios de *Legendre* (Apêndice A). Essas funções são tipicamente associadas a entidades topológicas da malha como vértices, arestas e faces (Babuška et al., 1989; Szabó e Babuška, 1991).

As primeiras quatro funções associadas aos vértices desse elemento são exatamente as mesmas empregadas nos elementos de quatro nós utilizados na versão- h do **MEF**, ou seja,

$$N_i(\xi, \eta) = \frac{1}{4}(1 + \xi_i \xi)(1 + \eta_i \eta), \quad i = 1, \dots, 4; \quad \xi_i = \eta_i = \pm 1. \quad (2.1)$$

Definem-se $4(p - 1)$ funções de forma associadas às arestas de número 1 a 4 desse elemento, para $p \geq 2$, as quais são dadas pelas seguintes expressões:

$$\begin{aligned} N_i^{(1)}(\xi, \eta) &= \frac{1}{2}(1 - \eta)\phi_i(\xi), \\ N_i^{(2)}(\xi, \eta) &= \frac{1}{2}(1 + \xi)\phi_i(\eta), \\ N_i^{(3)}(\xi, \eta) &= \frac{(-1)^i}{2}(1 + \eta)\phi_i(\xi), \end{aligned} \quad (2.2)$$

$$N_i^{(4)}(\xi, \eta) = \frac{(-1)^i}{2}(1 - \xi)\phi_i(\eta),$$

com $i = 2, \dots, p$. A função $\phi_i(\zeta)$ é definida por

$$\phi_i(\zeta) = \sqrt{\frac{2i-1}{2}} \int_{-1}^{\zeta} P_{i-1}(t), \quad (2.3)$$

sendo $P_{i-1}(t)$ o polinômio de *Legendre* de grau $i-1$ (Apêndice A). O termo $(-1)^i$ presente nas funções $N_i^{(3)}$ e $N_i^{(4)}$ é necessário para se obter invariância com respeito à rotação de coordenadas (Babuška et al., 1989).

Finalmente, definem-se $(p-2)(p-3)/2$ funções associadas à face do elemento, para $p \geq 4$, as quais são dadas por

$$N_{i,j}^{(0)}(\xi, \eta) = \phi_i(\xi)\phi_j(\eta), \quad i, j = 2, \dots, p-2; \quad 0 \leq i+j \leq p. \quad (2.4)$$

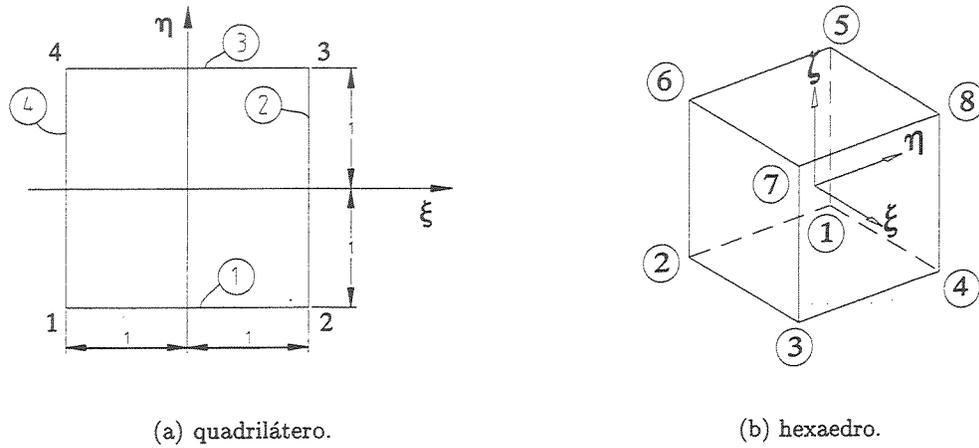


Figura 2.1: Elementos de referência quadrilateral e hexaédrico de (Szabó e Babuška, 1991).

2.1.2 Hexaedros

De maneira geral, as funções de forma associadas a hexaedros são análogas àquelas definidas para elementos quadrilaterais. Nesse caso, além das funções associadas a vértices, arestas e faces, definem-se ainda funções associadas ao interior do elemento de referência.

Para o elemento hexaédrico desenvolvido por (Szabó e Babuška, 1991), representado por $\hat{Q} = \{(\xi, \eta, \zeta) \in \mathbb{R}^3 \mid |\xi| < 1, |\eta| < 1, |\zeta| < 1\}$ (Figura 2.1(b)), definem-se oito funções de forma hierárquicas

associadas a seus vértices as quais são as mesmas empregadas na versão- h do **MEF**, ou seja,

$$N_i(\xi, \eta, \zeta) = \frac{1}{8}(1 + \xi_i\xi)(1 + \eta_i\eta)(1 + \zeta_i\zeta), \quad i = 1, \dots, 8; \quad \xi_i = \eta_i = \zeta_i = \pm 1. \quad (2.5)$$

Para $p \geq 2$, definem-se $12(p - 1)$ funções de forma associadas às arestas do elemento. Essas funções são similares às funções de aresta definidas para o elemento quadrilateral. As funções que conectam os vértices (1, 2) e (2, 3) (Figura 2.1(b)), por exemplo, são representadas por

$$\begin{aligned} N_{i-1}^{(1,2)}(\xi, \eta, \zeta) &= \frac{1}{4}\phi_i(\xi)(1 - \eta)(1 - \zeta), \\ N_{i-1}^{(2,3)}(\xi, \eta, \zeta) &= \frac{1}{4}\phi_i(\eta)(1 + \xi)(1 - \zeta), \end{aligned} \quad (2.6)$$

com $\phi_i(\zeta)$ dado pela relação (2.3) e $i = 2, \dots, p$. As demais funções de aresta são obtidas analogamente, permutando-se as variáveis ξ, η, ζ e seus respectivos sinais (± 1).

Para $p \geq 4$, definem-se $3(p - 2)(p - 3)$ funções de forma associadas às faces do elemento as quais são similares às funções de face descritas para o elemento quadrilateral. As funções associadas à face definida pelos vértices (1, 2, 5, 6) (Figura 2.1(b)), por exemplo, é dada pela seguinte expressão:

$$N_m^{(1,2,5,6)}(\xi, \eta, \zeta) = \frac{1}{2}(1 - \eta)\phi_i(\xi)\phi_j(\zeta) \quad i, j = 2, 3, \dots, p - 2; i + j = 4, 5, \dots, p; \quad (2.7)$$

sendo $m(i, j)$ o índice convencionado para a numeração das funções de face (Szabó e Babuška, 1991). As demais funções de face são obtidas analogamente, permutando-se as variáveis ξ, η, ζ e seus respectivos sinais (± 1).

Finalmente, para $p \geq 6$, definem-se $(p - 3)(p - 4)(p - 5)/6$ funções de forma associadas ao interior do elemento. Essas funções são dadas pela seguinte expressão geral:

$$N_m^{(0)} = \phi_i(\xi)\phi_j(\eta)\phi_k(\zeta), \quad i, j, k = 2, 3, \dots, p - 4; \quad i + j + k = 6, 7, \dots, p; \quad (2.8)$$

sendo $m = m(i, j, k)$ o índice convencionado para a numeração das funções internas (Szabó e Babuška, 1991).

Como constatado pelo trabalho de (Edgar e Surana, 1996), o emprego das funções hierárquicas associadas a quadriláteros e hexaedros sugeridas em (Szabó e Babuška, 1991) resultam em excelentes propriedades de esparsidade e condicionamento numérico das matrizes locais. Segundo (Carnevali et al., 1993), no entanto, não se pode esperar o mesmo resultado para as funções associadas a triângulos e tetraedros desenvolvidas por esses mesmos autores. Dessa forma, as seções restantes

concentram-se apenas no estudo de funções hierárquicas para triângulos e tetraedros.

2.1.3 Triângulos

As funções hierárquicas associadas a elementos triangulares propostas em (Szabó e Babuška, 1991) são definidas sobre o triângulo de referência equilátero $\hat{T} = \{(\xi, \eta) \in \mathbb{R}^2 \mid 0 < \eta < \sqrt{3}(1 + \xi), -1 < \xi < 0 \text{ ou } 0 < \eta < \sqrt{3}(1 - \xi), 0 < \xi < 1\}$ (Figura 2.2(a)) e, assim como no caso dos elementos quadrilaterais, são construídas com base na integração dos polinômios de *Legendre* (Apêndice A). Essas funções são dadas em termos das coordenadas de área L_i ($i = 1, 2, 3$) do elemento de referência as quais obedecem a relação $L_1 + L_2 + L_3 = 1$. Definem-se, nesse caso, 3 funções de vértice, $3(p - 1)$ funções de aresta, para $p \geq 2$, e $(p - 1)(p - 2)/2$ funções de face, para $p \geq 3$. As funções associadas aos vértices são dadas pelas próprias coordenadas de área. Essas funções possuem um comportamento linear, assumindo valor unitário no vértice i do elemento de referência e anulando-se nos lados opostos a esse mesmo vértice. As funções de aresta são construídas de modo que, em cada lado do triângulo, elas se encaixem exatamente nas funções de aresta correspondentes definidas para o elemento quadrilateral e ainda se anulem ao longo dos outros dois lados do triângulo. As funções de face possuem um aspecto do tipo bolha e são construídas de forma a se anularem em cada um dos lados do triângulo de referência.

As funções hierárquicas associadas a triângulos definidas adiante possuem, em geral, as mesmas características descritas acima. A quantidade de funções por elemento a cada nível de aproximação p também segue o mesmo padrão sugerido anteriormente. As diferenças aparecem na forma do elemento de referência utilizado e nos tipos de polinômios que compõem a base do espaço de interpolação. A utilização de coordenadas de área é bastante comum mas não obrigatória (Sherwin e Karniadakis, 1995).

As funções hierárquicas associadas ao elemento triangular de (Carnevali et al., 1993) são definidas sobre o triângulo de referência retângulo $\hat{T} = \{(\xi, \eta) \in \mathbb{R}^2 \mid 0 < \eta < 1 - \xi, 0 < \xi < 1\}$, denominado $[0, 1] 2 - simplex$ (Figura 2.2(b)). Essas funções, assim como no caso anterior, são dadas em termos das coordenadas de área L_i ($i = 1, 2, 3$) do elemento de referência. Com o objetivo de se obter ortogonalidade (em relação ao operador de *Laplace*) entre essas funções, emprega-se o procedimento de *Gram-Schmidt* na construção das mesmas.

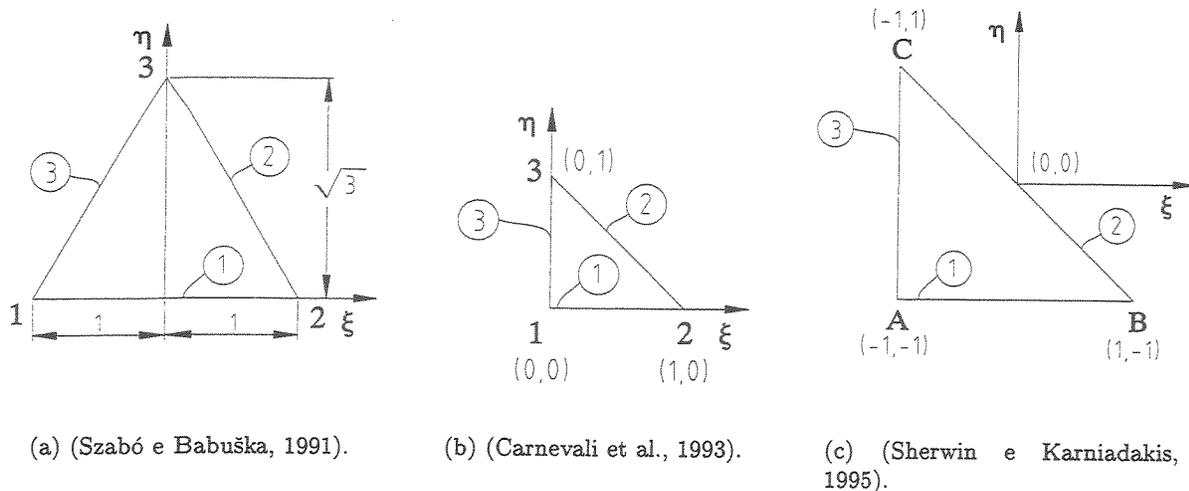


Figura 2.2: Elementos de referência triangulares.

As funções hierárquicas de (Webb e Abouchakra, 1995) são construídas com base nos polinômios de *Jacobi* de peso misto. Essas funções são definidas exatamente sobre o mesmo triângulo de referência usado por (Carnevali et al., 1993) sendo descritas através de coordenadas de área. Esse conjunto de funções foi elaborado de forma que tanto os modos associados às arestas quanto os associados às faces do elemento de referência fossem mutuamente ortogonais. Essa característica resulta na obtenção de matrizes de massa locais bem condicionadas. Verifica-se ainda que essa atracente propriedade se estende para o caso das matrizes de rigidez locais (Nogueira Jr. e Bittencourt, 2001b).

O conjunto de funções hierárquicas proposto por (Sherwin e Karniadakis, 1995) é definido sobre o triângulo de referência $T^2 = \{(\xi, \eta) \mid -1 \leq \xi, \eta; \xi + \eta \leq 0\}$, mapeado a partir do domínio quadrilateral $R^2 = \{(a, b) \mid -1 \leq a, b \leq 1\}$ (Figuras 2.2(c) e 2.3). Essas funções também são construídas com base nos polinômios de *Jacobi* além de poderem ser escritas em coordenadas de área. A principal característica associada a esse conjunto de funções é a definição do elemento de referência a partir de uma transformação de coordenadas. Isso permite que se escreva as expressões das funções de base sobre o triângulo de forma tensorizada tal como é feito naturalmente no caso de bases usadas em malhas estruturadas. A Figura 2.3 ilustra o procedimento de construção do domínio triangular T^2 a partir do colapsamento de um dos vértices do domínio quadrilateral R^2 . As expressões para as funções de base associadas ao elemento de referência triangular colapsado são escritas nas coorde-

nadas tensorizáveis (a, b) do quadrado de modo que a aplicação das equações de mapeamento sobre essas coordenadas resulta na obtenção de uma base polinomial em termos das coordenadas (ξ, η) (formalmente, tem-se: $f(a, b) = f(a(\xi, \eta), b(\xi, \eta))$). Como a mudança de coordenadas $(a, b) \mapsto (\xi, \eta)$ envolve uma transformação racional (i.e., $a = \frac{1+\xi}{1-\eta} - 1$), a obtenção de uma base polinomial em termos das coordenadas (ξ, η) exige a inclusão de termos especiais nas expressões das funções de base locais escritas nas coordenadas (a, b) . Essa estratégia garante que a forma polinomial da base seja preservada em ambos os domínios locais: o quadrilateral e o triangular.

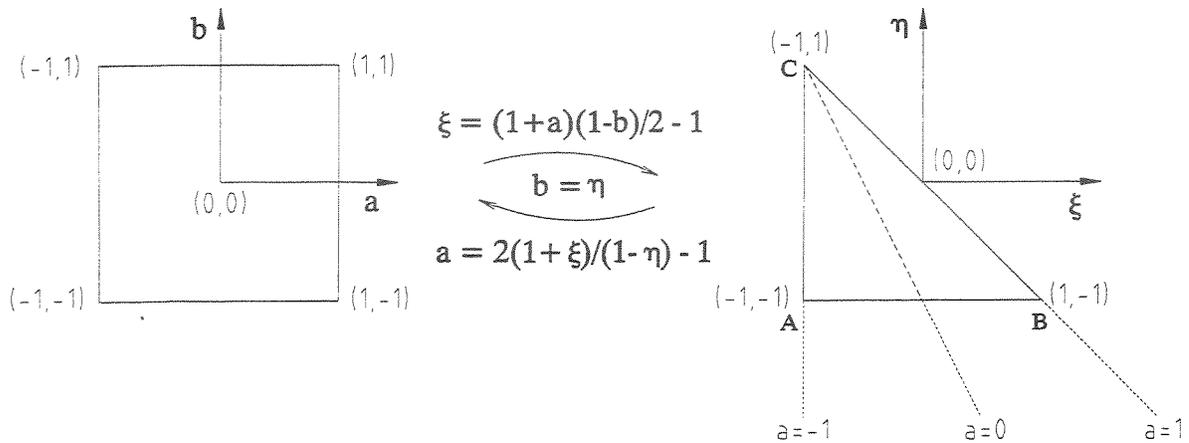


Figura 2.3: Mapeamento de coordenadas do elemento quadrilateral para o elemento triangular (Sherwin e Karniadakis, 1995).

A Tabela 2.1 apresenta, em termos das coordenadas de área L_i ($i = 1, 2, 3$), as expressões para as funções hierárquicas associadas aos diferentes tipos de elementos triangulares mencionados acima. Essa tabela traz ainda um novo conjunto de funções hierárquicas para malhas triangulares proposto com base na referência (Sherwin e Karniadakis, 1995). Partindo-se do conjunto de funções de base ortogonal proposto por (Dubiner, 1991) e aproveitando a formulação multi-domínios sugerida por (Sherwin e Karniadakis, 1995), a qual preserva a continuidade C^0 sobre malhas triangulares conformes, obteve-se uma base hierárquica adequada para os espaços de aproximação comumente utilizados no MEF. As funções associadas a esse novo elemento são definidas sobre o mesmo triângulo de referência sugerido em (Sherwin e Karniadakis, 1995). O novo conjunto de funções hierárquicas foi idealizado de forma que as funções associadas à face do elemento de referência fossem mutuamente ortogonais.

(Szabó e Babuška, 1991)	vértice	$N_1^{(1)} = L_1, \quad N_1^{(2)} = L_2, \quad N_1^{(3)} = L_3$
	aresta ¹	$N_i^{(1)} = L_1 L_2 \varphi_i(L_2 - L_1)$ $N_i^{(2)} = L_2 L_3 \varphi_i(L_3 - L_2)$ $N_i^{(3)} = L_3 L_1 \varphi_i(L_1 - L_3)$
	face ²	$N_{(p-2)(p-3)/2+i}^{(0)} = L_1 L_2 L_3 P_{(p-3)+1-i}(L_2 - L_1) P_{i-1}(2L_3 - 1)$
(Carnevali et al., 1993)	vértice	$h_1^{(1)} = L_1, \quad h_2^{(1)} = L_2, \quad h_3^{(1)} = L_3$
	aresta ³	$h_1^{(p)} = -2L_1 L_2 E_{p-2}(L_1, L_2)$ $h_2^{(p)} = -2L_2 L_3 E_{p-2}(L_2, L_3)$ $h_3^{(p)} = -2L_3 L_1 E_{p-2}(L_3, L_1)$
	face ⁴	$h_{\alpha,\beta}^{(p)} = L_1 L_2 L_3 F_{\alpha,\beta}(L_1, L_2)$
(Webb e Abouchakra, 1995)	vértice	$N_1 = L_1, \quad N_2 = L_2, \quad N_3 = L_3$
	aresta ⁵	$N_{a+1} = L_1 L_2 P_{p-2}^{(2,2)}(L_2 - L_1)$ $N_{a+2} = L_2 L_3 P_{p-2}^{(2,2)}(L_3 - L_2)$ $N_{a+3} = L_3 L_1 P_{p-2}^{(2,2)}(L_1 - L_3)$
	face ⁶	$N_{a+4+i} = L_1 L_2 L_3 (1 - L_3)^k P_i^{(2,2k+5)} (1 - 2L_3) P_k^{(2,2)} \left(\frac{L_2 - L_1}{1 - L_3} \right)$
(Sherwin e Karniadakis, 1995)	vértice	$\overset{1-2}{g}_{10}^{vert A} = L_1, \quad \overset{1-2}{g}_{01}^{vert B} = L_2, \quad \overset{1-2}{g}_{01}^{vert C} = L_3$
	aresta	$\overset{1-2}{g}_{10}^{aresta 1} = L_1 L_2 P_{l-2}^{1,1} \left(\frac{L_2 - L_1}{1 - L_3} \right) (1 - L_3)^{l-2}$
		$\overset{1-2}{g}_{1m}^{aresta 2} = L_2 L_3 P_{m-1}^{1,1} (2L_3 - 1)$
		$\overset{1-2}{g}_{1m}^{aresta 3} = L_3 L_1 P_{m-1}^{1,1} (2L_3 - 1)$
face ⁷	$\overset{1-2}{g}_{lm}^{face} = L_1 L_2 L_3 P_{l-2}^{1,1} \left(\frac{L_2 - L_1}{1 - L_3} \right) (1 - L_3)^{l-2} P_{m-1}^{2l-1,1} (2L_3 - 1)$	
Propostas	vértice	$\overset{1-2}{N}_{10}^{(vert A)} = L_1, \quad \overset{1-2}{N}_{01}^{(vert B)} = L_2, \quad \overset{1-2}{N}_{01}^{(vert C)} = L_3$
	aresta	$\overset{1-2}{N}_{10}^{(aresta 1)} = L_1 L_2 P_{l-2}^{2,2} \left(\frac{L_2 - L_1}{1 - L_3} \right) (1 - L_3)^{l-2}$
		$\overset{1-2}{N}_{1m}^{(aresta 2)} = L_2 L_3 P_{m-1}^{2,2} (2L_3 - 1)$
		$\overset{1-2}{N}_{1m}^{(aresta 3)} = L_3 L_1 P_{m-1}^{2,2} (2L_3 - 1)$
face ⁸	$\overset{1-2}{N}_{lm}^{(face)} = L_1 L_2 L_3 P_{l-2}^{2,2} \left(\frac{L_2 - L_1}{1 - L_3} \right) (1 - L_3)^{l-2} P_{m-1}^{2l+1,2} (2L_3 - 1)$	

Tabela 2.1: Expressões para as funções hierárquicas associadas aos elementos triangulares.

¹O termo $\varphi_i(\zeta)$ é definido através da seguinte expressão: $\frac{1}{4}(1 - \zeta^2)\varphi_i(\zeta) = \phi_i(\zeta) = \sqrt{\frac{2i-1}{2}} \int_{-1}^{\zeta} P_{i-1}(t)dt$.

² P_k é o polinômio de Legendre de ordem k (vide Apêndice A).

³O termo $E_{p-2}(L_i, L_j)$ é definido através da seguinte expressão, para $p \geq 2$: $E_{p-2}(L_i, L_j) = \sum_{k=0}^{p-2} (-1)^k \frac{1}{k+1} \binom{p-2}{k} \binom{p-1}{k} L_i^k L_j^{p-2-k}$.

⁴O termo $F_{\alpha,\beta}(L_r, L_s)$ é definido através da seguinte expressão, para $\alpha, \beta = 0, \dots, p-3$; $\alpha + \beta = p-3$ e $p \geq 3$: $F_{\alpha,\beta}(L_r, L_s) = \sum_{i=0}^{\beta} \sum_{j=0}^{\alpha} (-\frac{1}{2})^{i+j} i! j! (i+j)! \binom{\alpha}{j} \binom{\alpha+1}{j} \binom{\beta}{i} \binom{\beta+1}{i} \times \frac{1}{\prod_{k=1}^{i+j} [k(\alpha+\beta+2) - k(k-1)/2]} (L_r)^{\alpha-j} (L_s)^{\beta-i}$.

⁵O índice a , para $p = 2, 3, \dots, n$, é definido por $a = \frac{1}{2}p(p+1)$. $P_k^{(\alpha,\beta)}$ são polinômios de Jacobi de ordem k e pesos α e β (vide Apêndice A).

⁶O índice k , para $p \geq 3$, é dado por $k = p-3-i$.

⁷O índice lm é definido de modo que $(2 \leq l; 1 \leq m | l < L; l+m < M)$, com L e M indicando o número total de

Os gráficos de algumas funções hierárquicas associadas às arestas e às faces dos diferentes elementos estudados são ilustrados na Figura 2.4.

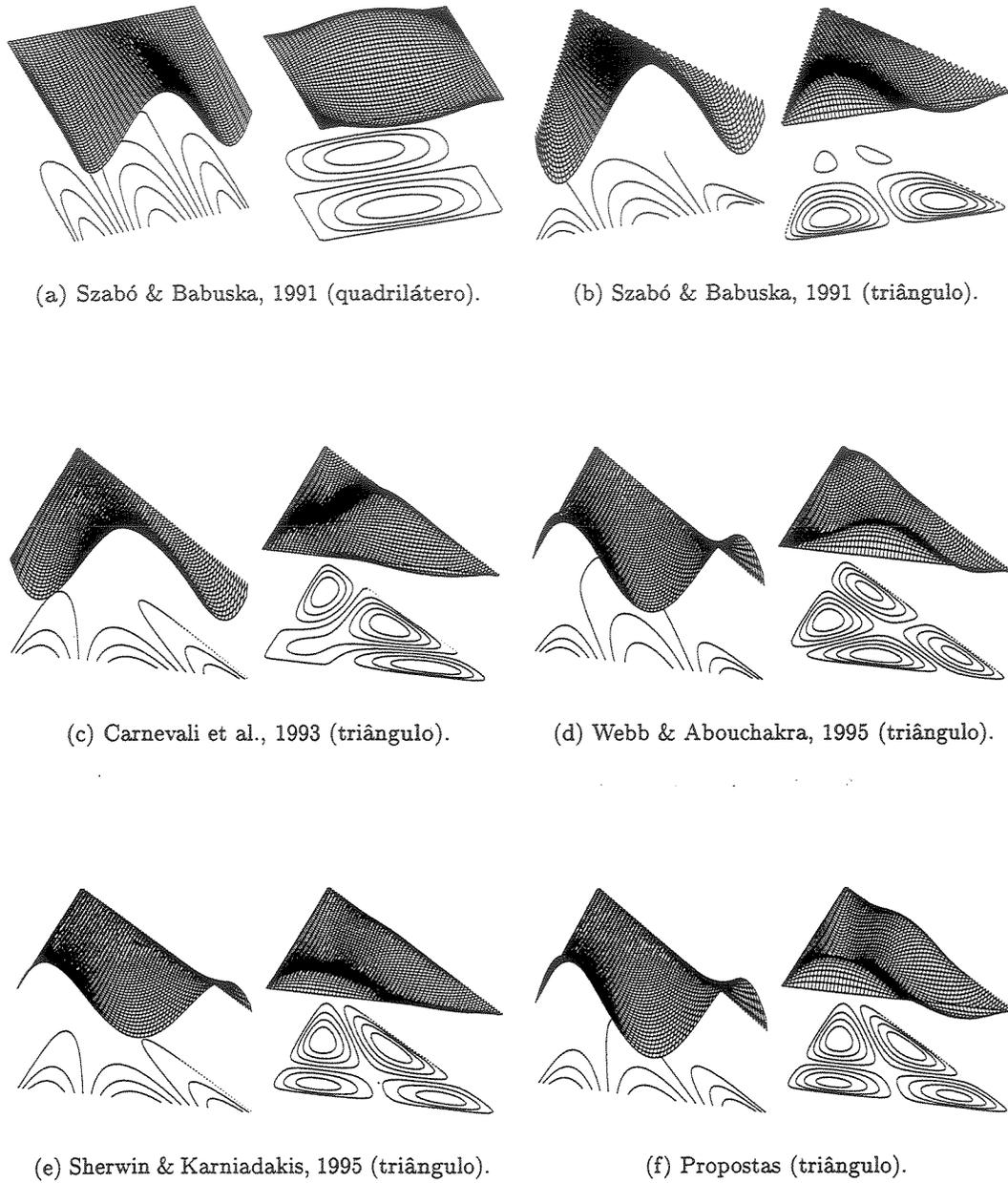


Figura 2.4: Funções hierárquicas de ordens 4 e 5 associadas às arestas e às faces, respectivamente, dos diferentes elementos estudados.

funções de base. $P_k^{\alpha,\beta}$ são polinômios de *Jacobi* de ordem k e pesos α e β .

⁸O índice lm e os polinômios $P_k^{\alpha,\beta}$ são definidos tal como em (Sherwin e Karniadakis, 1995) (vide Apêndice A).

2.1.4 Tetraedros

As funções hierárquicas associadas ao elemento tetraédrico de (Szabó e Babuška, 1991) são definidas sobre o tetraedro de referência derivado do triângulo equilátero descrito anteriormente (Figura 2.5(a)). Essas funções são elaboradas com base na integração dos polinômios de *Legendre* e representadas em termos das coordenadas de volume L_i ($i = 1, 2, 3, 4$) do tetraedro de tal forma que $L_1 + L_2 + L_3 + L_4 = 1$. Definem-se, nesse caso, 4 funções de vértice, $6(p - 1)$ funções de aresta, para $p \geq 2$, $2(p - 1)(p - 2)$ funções de face, para $p \geq 3$, e $(p - 1)(p - 2)(p - 3)/6$ funções internas, para $p \geq 4$. As funções associadas a esse elemento possuem características semelhantes às encontradas nas funções hierárquicas definidas para o elemento triangular de (Szabó e Babuška, 1991).

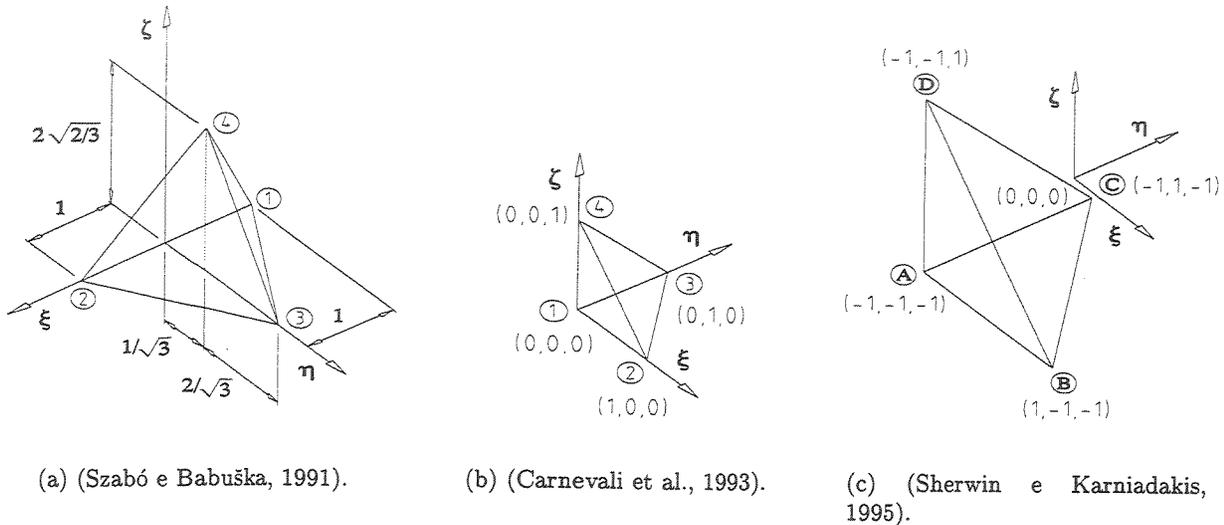


Figura 2.5: Elementos de referência tetraédricos.

Os demais elementos tetraédricos sugeridos na literatura (Carnevali et al., 1993; Sherwin e Karniadakis, 1995; Abouchakra, 1996) são extensões imediatas dos respectivos elementos triangulares descritos anteriormente e portanto preservam boa parte das propriedades destes últimos. Observa-se, por exemplo, que as funções hierárquicas associadas aos elementos tetraédricos de (Carnevali et al., 1993) e (Abouchakra, 1996) são definidas sobre o elemento de referência $[0, 1]^3$ - *simplex* (Figura 2.5(b)). Já as funções elaboradas por (Sherwin e Karniadakis, 1995) são definidas sobre o elemento de referência $T^3 = \{(\xi, \eta, \zeta) \mid -1 \leq \xi, \eta, \zeta; \xi + \eta + \zeta \leq -1\}$, mapeado a partir do domínio hexaédrico $R^3 = \{(a, b, c) \mid -1 \leq a, b, c \leq 1\}$ (Figuras 2.5(c) e 2.6). No caso tridimensional, a

quantidade de funções por elemento a cada nível de aproximação p normalmente respeita o padrão adotado em (Szabó e Babuška, 1991).

A Figura 2.6 ilustra o procedimento de construção do domínio tetraédrico T^3 , associado às funções hierárquicas desenvolvidas em (Sherwin e Karniadakis, 1995), a partir do domínio hexaédrico R^3 . Diferentemente do caso 2D, observa-se que o mapeamento do elemento de referência hexaédrico para o elemento tetraédrico é realizado em três etapas. Primeiramente transforma-se o domínio hexaédrico em um domínio prismático. Em seguida, transforma-se este último num domínio com formato piramidal. Finalmente, converte-se o elemento piramidal num elemento de referência tetraédrico. A construção das expressões para as funções de base associadas a esse elemento tetraédrico é feita de forma análoga ao caso do elemento triangular T^2 obtido a partir do elemento quadrilateral R^2 .

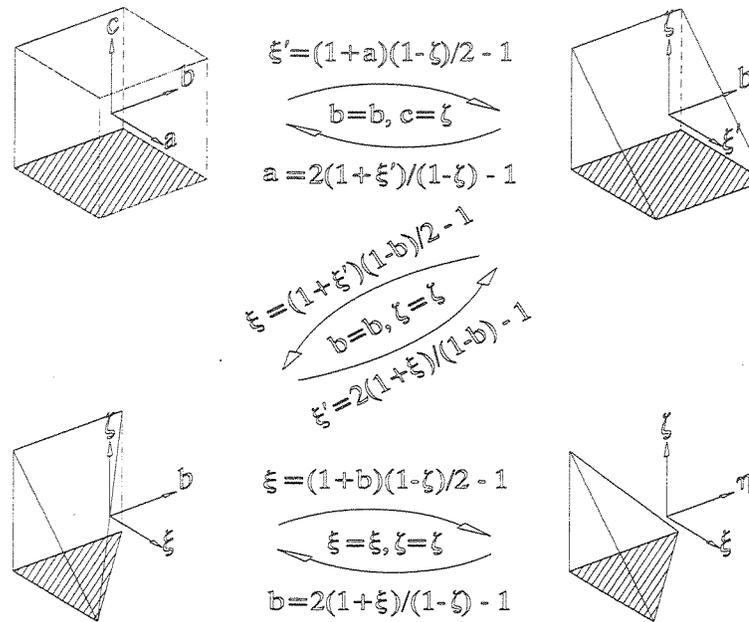


Figura 2.6: Mapeamento de coordenadas do elemento hexaédrico para o elemento tetraédrico de (Sherwin e Karniadakis, 1995).

A Tabela 2.2 apresenta as expressões para as funções hierárquicas associadas aos diferentes tipos de elementos tetraédricos mencionados acima. Essa tabela traz ainda um novo conjunto de funções hierárquicas para malhas tetraédricas concebido com base na referência (Sherwin e Karniadakis, 1995). Esse conjunto de funções é uma extensão imediata do conjunto associado a malhas triangulares proposto anteriormente.

(Szabó e Babuška, 1991)	vértice	$N_1^{(1)} = L_1, N_1^{(2)} = L_2, N_1^{(3)} = L_3, N_1^{(4)} = L_4$
	aresta ⁹	$N_{i-1}^{(1,2)} = L_1 L_2 \varphi_i(L_2 - L_1)$ $N_{i-1}^{(2,3)} = L_2 L_3 \varphi_i(L_3 - L_2)$ $N_{i-1}^{(3,1)} = L_3 L_1 \varphi_i(L_1 - L_3)$ $N_{i-1}^{(4,1)} = L_4 L_1 \varphi_i(L_1 - L_4)$ $N_{i-1}^{(2,4)} = L_2 L_4 \varphi_i(L_4 - L_2)$ $N_{i-1}^{(3,4)} = L_3 L_4 \varphi_i(L_4 - L_3)$
	face ¹⁰	$N_m^{(1,2,3)} = L_1 L_2 L_3 P_i(L_2 - L_1) P_j(2L_3 - 1)$ $N_m^{(1,2,4)} = L_1 L_2 L_4 P_i(L_2 - L_1) P_j(2L_4 - 1)$ $N_m^{(2,3,4)} = L_2 L_3 L_4 P_i(L_3 - L_2) P_j(2L_4 - 1)$ $N_m^{(3,1,4)} = L_3 L_1 L_4 P_i(L_1 - L_3) P_j(2L_4 - 1)$
	interior ¹¹	$N_m^{(0)} = L_1 L_2 L_3 L_4 P_i(L_2 - L_1) P_j(2L_3 - 1) P_k(2L_4 - 1)$
(Carnevali et al., 1993)	vértice	$h_1^{(1)} = L_1, h_2^{(1)} = L_2, h_3^{(1)} = L_3, h_4^{(1)} = L_4$
	aresta ¹²	$h^{(p)} = -2L_i L_j E_{p-2}(L_i, L_j)$
	face ¹³	$h_{\alpha\beta}^{(p)} = L_r L_s L_t F_{\alpha,\beta}(L_r, L_s)$
	interior ¹⁴	$h_{\alpha\beta\gamma}^{(p)} = L_1 L_2 L_3 L_4 B_{\alpha\beta\gamma}(L_1, L_2, L_3)$
(Abouchakra, 1996)	vértice	$N_1 = L_1, N_2 = L_2, N_3 = L_3, N_4 = L_4$
	aresta ¹⁵	$N_{a+1} = L_1 L_2 P_{p-2}^{(2,2)}(L_2 - L_1)$ $N_{a+2} = L_1 L_3 P_{p-2}^{(2,2)}(L_3 - L_1)$ $N_{a+3} = L_1 L_4 P_{p-2}^{(2,2)}(L_4 - L_1)$ $N_{a+4} = L_2 L_3 P_{p-2}^{(2,2)}(L_3 - L_2)$ $N_{a+5} = L_2 L_4 P_{p-2}^{(2,2)}(L_4 - L_2)$ $N_{a+6} = L_3 L_4 P_{p-2}^{(2,2)}(L_4 - L_3)$
	face ¹⁶	$N_{a+7+i} = L_1 L_2 L_3 (1 - L_3)^k P_i^{(2,2k+5)}(1 - 2L_3) P_k^{(2,2)}\left(\frac{L_2 - L_1}{1 - L_3}\right)$ $N_{a+8+i+(p-3)} = L_1 L_2 L_4 (1 - L_4)^k P_i^{(2,2k+5)}(1 - 2L_4) P_k^{(2,2)}\left(\frac{L_2 - L_1}{1 - L_4}\right)$ $N_{a+9+i+2(p-3)} = L_1 L_3 L_4 (1 - L_4)^k P_i^{(2,2k+5)}(1 - 2L_4) P_k^{(2,2)}\left(\frac{L_3 - L_1}{1 - L_4}\right)$ $N_{a+10+i+3(p-3)} = L_2 L_3 L_4 (1 - L_4)^k P_i^{(2,2k+5)}(1 - 2L_4) P_k^{(2,2)}\left(\frac{L_3 - L_2}{1 - L_4}\right)$
	interior ¹⁷	$N_{a+11+j+\frac{i(i+1)}{2}+4(p-3)} = L_1 L_2 L_3 L_4 (1 - L_2 - L_3)^k (1 - L_3)^q P_k^{(2,2)}\left(\frac{L_4 - L_1}{1 - L_2 - L_3}\right)$ $\times P_q^{(2,2k+5)}\left(\frac{1 - L_3 - 2L_2}{1 - L_3}\right) P_j^{(2,2k+2q+8)}(1 - 2L_3)$

⁹O termo $\varphi_i(\zeta)$ é definido exatamente como no caso do elemento triangular.

¹⁰ $m = m(i, j)$ é o índice convencionado para a numeração das funções de face do tetraedro; P_i, P_j são polinômios de Legendre de ordens i e j respectivamente, com $i, j = 0, 1, 2, \dots, p-3$ e $i+j = 0, 1, \dots, p-3$.

¹¹ $m = m(i, j, k)$ é o índice convencionado para a numeração das funções internas do tetraedro padrão; P_i, P_j, P_k são polinômios de Legendre de ordens i, j e k respectivamente, com $i, j, k = 0, 1, 2, \dots, p-4$ e $i+j+k = 0, 1, \dots, p-4$.

¹²O termo $E_{p-2}(L_i, L_j)$ é definido exatamente como no caso do elemento triangular.

¹³O termo $F_{\alpha,\beta}(L_r, L_s)$ é definido exatamente como no caso do elemento triangular.

¹⁴O termo $B_{\alpha\beta\gamma}(L_1, L_2, L_3)$ é definido pela seguinte expressão, para $\alpha, \beta, \gamma \geq 0$; $\alpha + \beta + \gamma = p - 4$; e $p \geq 4$: $B_{\alpha\beta\gamma}(L_1, L_2, L_3) = \bar{B}_{\alpha 0 0}^{(\alpha+\beta+\gamma)}(L_1) \times \bar{B}_{0 \beta 0}^{(\beta+\gamma)}(L_1, L_2) \times \bar{B}_{0 0 \gamma}^{(\gamma)}(L_1, L_2, L_3)$, com $\bar{B}_{\alpha 0 0}^{(m)}(L_1) = \sum_{i=0}^{\alpha} (-1)^i i! \binom{\alpha}{i} \binom{\alpha+1}{i} \frac{(2m+5-i)!}{(2m+5)!} L_1^{\alpha-i}$; $\bar{B}_{0 \beta 0}^{(m)}(L_1, L_2) = \sum_{i=0}^{\beta} i! \binom{\beta}{i} \binom{\beta+1}{i} \frac{(2m+3-i)!}{(2m+3)!} L_2^{\beta-i} (L_1 - 1)^i$; $\bar{B}_{0 0 \gamma}^{(m)}(L_1, L_2, L_3) = \sum_{i=0}^{\beta} i! \binom{\gamma}{i} \binom{\gamma+1}{i} \frac{(2m+1-i)!}{(2m+1)!} L_3^{\gamma-i} (L_2 + L_1 - 1)^i$.

¹⁵O índice a , para $p = 2, 3, \dots, n$, é definido por $a = p(p+1)(p+2)/6$ e $P_m^{(a,b)}$ são polinômios de Jacobi de ordem m e pesos a e b .

¹⁶O índice k , para $p \geq 3$ e $i = 0, 1, \dots, p-3$, é dado por $k = p-3-i$.

¹⁷Os índices k e q , para $p \geq 4$, $i = 0, 1, \dots, p-4$ e $j = 0, 1, \dots, i$, são dados respectivamente por $k = p-4-i$ e $q = i-j$.

(Sherwin e Karniadakis, 1995)	vértice	$\overset{1-3 \text{ vert } A}{g_{100}} = L_1, \overset{1-3 \text{ vert } B}{g_{100}} = L_2, \overset{1-3 \text{ vert } C}{g_{010}} = L_3, \overset{1-3 \text{ vert } D}{g_{001}} = L_4$
	aresta	$\overset{1-3 \text{ aresta } 1}{g_{100}} = L_1 L_2 P_{l-2}^{1,1} \left(\frac{L_2 - L_1}{1 - L_3 - L_4} \right) (1 - L_3 - L_4)^{l-2}$ $\overset{1-3 \text{ aresta } 2}{g_{1m0}} = L_2 L_3 P_{m-1}^{1,1} \left(\frac{2L_3}{1 - L_4} - 1 \right) (1 - L_4)^{m-1}$ $\overset{1-3 \text{ aresta } 3}{g_{1m0}} = L_1 L_3 P_{m-1}^{1,1} \left(\frac{2L_3}{1 - L_4} - 1 \right) (1 - L_4)^{m-1}$ $\overset{1-3 \text{ aresta } 4}{g_{10n}} = L_1 L_4 P_{n-1}^{1,1} (2L_4 - 1)$ $\overset{1-3 \text{ aresta } 5}{g_{10n}} = L_2 L_4 P_{n-1}^{1,1} (2L_4 - 1)$ $\overset{1-3 \text{ aresta } 6}{g_{01n}} = L_3 L_4 P_{n-1}^{1,1} (2L_4 - 1)$
	face	$\overset{1-3 \text{ face } 1}{g_{lm0}} = L_1 L_2 L_3 P_{l-2}^{1,1} \left(\frac{L_2 - L_1}{1 - L_3 - L_4} \right) (1 - L_3 - L_4)^{l-2} P_{m-1}^{2l-1,1} \left(\frac{2L_3}{1 - L_4} - 1 \right) (1 - L_4)^{m-1}$ $\overset{1-3 \text{ face } 2}{g_{10n}} = L_1 L_2 L_4 P_{l-2}^{1,1} \left(\frac{L_2 - L_1}{1 - L_3 - L_4} \right) (1 - L_3 - L_4)^{l-2} P_{n-1}^{2l-1,1} (2L_4 - 1)$ $\overset{1-3 \text{ face } 3}{g_{1mn}} = L_2 L_3 L_4 P_{m-1}^{1,1} \left(\frac{2L_3}{1 - L_4} - 1 \right) (1 - L_4)^{m-1} P_{n-1}^{2m+1,1} (2L_4 - 1)$ $\overset{1-3 \text{ face } 4}{g_{1mn}} = L_1 L_3 L_4 P_{m-1}^{1,1} \left(\frac{2L_3}{1 - L_4} - 1 \right) (1 - L_4)^{m-1} P_{n-1}^{2m+1,1} (2L_4 - 1)$
	interior ¹⁸	$\overset{1-3 \text{ interior}}{g_{lmn}} = L_1 L_2 L_3 L_4 P_{l-2}^{1,1} \left(\frac{L_2 - L_1}{1 - L_3 - L_4} \right) (1 - L_3 - L_4)^{l-2} \times P_{m-1}^{2l-1,1} \left(\frac{2L_3}{1 - L_4} - 1 \right) (1 - L_4)^{m-1} P_{n-1}^{2l+2m-1,1} (2L_4 - 1)$
Propostas	vértice	$\overset{1-3 \text{ vert } A}{N_{100}} = L_1, \overset{1-3 \text{ vert } B}{N_{100}} = L_2, \overset{1-3 \text{ vert } C}{N_{010}} = L_2, \overset{1-3 \text{ vert } D}{N_{001}} = L_4$
	aresta	$\overset{1-3 \text{ aresta } 1}{N_{100}} = L_1 L_2 P_{l-2}^{2,2} \left(\frac{L_2 - L_1}{1 - L_3 - L_4} \right) (1 - L_3 - L_4)^{l-2}$ $\overset{1-3 \text{ aresta } 2}{N_{1m0}} = L_2 L_3 P_{m-1}^{2,2} \left(\frac{2L_3}{1 - L_4} - 1 \right) (1 - L_4)^{m-1}$ $\overset{1-3 \text{ aresta } 3}{N_{1m0}} = L_1 L_3 P_{m-1}^{2,2} \left(\frac{2L_3}{1 - L_4} - 1 \right) (1 - L_4)^{m-1}$ $\overset{1-3 \text{ aresta } 4}{N_{10n}} = L_1 L_4 P_{n-1}^{2,2} (2L_4 - 1)$ $\overset{1-3 \text{ aresta } 5}{N_{10n}} = L_2 L_4 P_{n-1}^{2,2} (2L_4 - 1)$ $\overset{1-3 \text{ aresta } 6}{N_{01n}} = L_3 L_4 P_{n-1}^{2,2} (2L_4 - 1)$
	face	$\overset{1-3 \text{ face } 1}{N_{lm0}} = L_1 L_2 L_3 P_{l-2}^{2,2} \left(\frac{L_2 - L_1}{1 - L_3 - L_4} \right) (1 - L_3 - L_4)^{l-2} P_{m-1}^{2l+1,2} \left(\frac{2L_3}{1 - L_4} - 1 \right) (1 - L_4)^{m-1}$ $\overset{1-3 \text{ face } 2}{N_{10n}} = L_1 L_2 L_4 P_{l-2}^{2,2} \left(\frac{L_2 - L_1}{1 - L_3 - L_4} \right) (1 - L_3 - L_4)^{l-2} P_{n-1}^{2l+2,2} (2L_4 - 1)$ $\overset{1-3 \text{ face } 3}{N_{1mn}} = L_2 L_3 L_4 P_{m-1}^{2,2} \left(\frac{2L_3}{1 - L_4} - 1 \right) (1 - L_4)^{m-1} P_{n-1}^{2m+4,2} (2L_4 - 1)$ $\overset{1-3 \text{ face } 4}{N_{1mn}} = L_1 L_3 L_4 P_{m-1}^{2,2} \left(\frac{2L_3}{1 - L_4} - 1 \right) (1 - L_4)^{m-1} P_{n-1}^{2m+4,2} (2L_4 - 1)$
	interior ¹⁹	$\overset{1-3 \text{ interior}}{N_{lmn}} = L_1 L_2 L_3 L_4 P_{l-2}^{2,2} \left(\frac{L_2 - L_1}{1 - L_3 - L_4} \right) (1 - L_3 - L_4)^{l-2} \times P_{m-1}^{2l+1,2} \left(\frac{2L_3}{1 - L_4} - 1 \right) (1 - L_4)^{m-1} P_{n-1}^{2l+2m+2,2} (2L_4 - 1)$

Tabela 2.2: Expressões para as funções hierárquicas associadas aos elementos tetraédricos.

¹⁸O índice lmn é definido de tal forma que $(2 \leq l; 1 \leq m, n \mid l < L; l + m < M; l + m + n < N)$, com L, M e N indicando o número total de funções de base. $P_k^{\alpha, \beta}$ são polinômios de *Jacobi* de ordem k e pesos α e β .

¹⁹O índice lmn e os polinômios $P_k^{\alpha, \beta}$ são definidos tal como em (Sherwin e Karniadakis, 1995).

2.2 Ortogonalidade dos Modos Internos Associados às Novas Funções de Base

2.2.1 Elemento triangular

As novas funções hierárquicas associadas a domínios triangulares apresentadas na Tabela 2.1, quando escritas nas coordenadas cartesianas (a, b) do elemento de referência sugerido em (Sherwin e Karniadakis, 1995) (veja Figura 2.3), apresentam o seguinte formato:

- Funções de Vértice

$$\begin{aligned} N_{10}^{1-2(\text{vert } A)} &= \left(\frac{1-a}{2}\right) \left(\frac{1-b}{2}\right), \\ N_{01}^{1-2(\text{vert } B)} &= \left(\frac{1+a}{2}\right) \left(\frac{1-b}{2}\right), \\ N_{01}^{1-2(\text{vert } C)} &= 1 \left(\frac{1+b}{2}\right). \end{aligned} \quad (2.9)$$

- Funções de Aresta

$$\begin{aligned} N_{l0}^{1-2(\text{aresta } 1)} &= \left(\frac{1+a}{2}\right) \left(\frac{1-a}{2}\right) P_{l-2}^{2,2}(a) \left(\frac{1-b}{2}\right)^l, \\ N_{1m}^{1-2(\text{aresta } 2)} &= \left(\frac{1+a}{2}\right) \left(\frac{1-b}{2}\right) \left(\frac{1+b}{2}\right) P_{m-1}^{2,2}(b), \\ N_{1m}^{1-2(\text{aresta } 3)} &= \left(\frac{1-a}{2}\right) \left(\frac{1-b}{2}\right) \left(\frac{1+b}{2}\right) P_{m-1}^{2,2}(b). \end{aligned} \quad (2.10)$$

- Funções de Face

$$N_{lm}^{1-2(\text{face})} = \left(\frac{1+a}{2}\right) \left(\frac{1-a}{2}\right) P_{l-2}^{2,2}(a) \left(\frac{1-b}{2}\right)^l \left(\frac{1+b}{2}\right) P_{m-1}^{2l+1,2}(b). \quad (2.11)$$

As propriedades de ortogonalidade dessas funções em relação ao produto interno usual (i.e., a integral do produto de duas funções quaisquer) podem ser claramente observadas ao se tomar como exemplo o produto entre duas funções de face. Efetuando-se o produto interno entre duas funções de face quaisquer sobre o triângulo de referência T^2 , chega-se ao seguinte resultado:

$$\begin{aligned} \left(N_{lm}^{1-2(face)}, N_{pq}^{1-2(face)} \right)_{T^2} &= \int_{-1}^1 \left(\frac{1+a}{2} \right)^2 \left(\frac{1-a}{2} \right)^2 P_{l-2}^{2,2}(a) P_{p-2}^{2,2}(a) da \\ &\times \int_{-1}^1 \left(\frac{1+b}{2} \right)^2 \left(\frac{1-b}{2} \right)^{l+p} P_{m-1}^{2l+1,2}(b) P_{q-1}^{2p+1,2}(b) \left(\frac{1-b}{2} \right) db, \end{aligned} \quad (2.12)$$

sendo que o termo $\left(\frac{1-b}{2}\right)$, na segunda integral, representa o Jacobiano da transformação de R^2 para T^2 . Dessa forma, tem-se

$$\begin{aligned} \left(N_{lm}^{1-2(face)}, N_{pq}^{1-2(face)} \right)_{T^2} &= \left(\frac{1}{2} \right)^4 \int_{-1}^1 (1+a)^2 (1-a)^2 P_{l-2}^{2,2}(a) P_{p-2}^{2,2}(a) da \\ &\times \left(\frac{1}{2} \right)^{l+p+3} \int_{-1}^1 (1+b)^2 (1-b)^{l+p+1} P_{m-1}^{2l+1,2}(b) P_{q-1}^{2p+1,2}(b) db. \end{aligned} \quad (2.13)$$

Observa-se nessas expressões que $P_n^{\alpha,\beta}(x)$ são polinômios de *Jacobi* de ordem n no intervalo $[-1, 1]$. Assim, pela propriedade de ortogonalidade (A.9) do Apêndice A.2, é possível concluir que, se $l \neq p$, então a primeira integral em (2.13) será nula pois

$$\int_{-1}^1 (1+a)^2 (1-a)^2 P_{l-2}^{2,2}(a) P_{p-2}^{2,2}(a) da = C_1 \delta_{lp}, \quad (2.14)$$

sendo C_1 uma constante que depende dos pesos α, β e da ordem polinomial n de $P_n^{\alpha,\beta}(x)$ (vide Apêndice A.2).

Analogamente, se $l = p$ mas $m \neq q$, então a segunda integral em (2.13) será nula pois

$$\int_{-1}^1 (1+b)^2 (1-b)^{l+p+1} P_{m-1}^{2l+1,2}(b) P_{q-1}^{2p+1,2}(b) db = C_2 \delta_{mq}. \quad (2.15)$$

Dessa forma, nota-se que para qualquer função associada à face do elemento de referência, vale a seguinte relação de ortogonalidade:

$$\left(N_{lm}^{1-2(face)}, N_{pq}^{1-2(face)} \right)_{T^2} = \bar{C} \left(\frac{1}{2} \right)^{l+p+7} \delta_{lp} \delta_{mq}, \quad (2.16)$$

sendo \bar{C} uma constante obtida a partir do produto das constantes C_1 e C_2 .

Esse resultado garante uma estrutura esparsa à matriz de massa do elemento uma vez que os acoplamentos entre as funções de face dessa matriz reduzem-se a um padrão diagonal. Da mesma forma que no caso das funções de face entre si, é possível observar outras relações de ortogonalidade similares a (2.16) combinando-se as funções de face e as de aresta propostas anteriormente. Essas relações acentuam a esparsidade da estrutura da matriz de massa do elemento.

2.2.2 Elemento tetraédrico

As novas funções hierárquicas associadas a domínios tetraédricos apresentadas na Tabela 2.2, quando escritas nas coordenadas cartesianas (a, b, c) do elemento de referência sugerido em (Sherwin e Karniadakis, 1995) (veja Figura 2.6), apresentam o seguinte formato:

- Funções de Vértice

$$\begin{aligned}
 N_{100}^{1-3(\text{vert } A)} &= \left(\frac{1-a}{2}\right) \left(\frac{1-b}{2}\right) \left(\frac{1-c}{2}\right), \\
 N_{100}^{1-3(\text{vert } B)} &= \left(\frac{1+a}{2}\right) \left(\frac{1-b}{2}\right) \left(\frac{1-c}{2}\right), \\
 N_{010}^{1-3(\text{vert } C)} &= \left(\frac{1+b}{2}\right) \left(\frac{1-c}{2}\right), \\
 N_{001}^{1-3(\text{vert } D)} &= \left(\frac{1+c}{2}\right).
 \end{aligned} \tag{2.17}$$

- Funções de Aresta

$$\begin{aligned}
 N_{100}^{1-3(\text{aresta } 1)} &= \left(\frac{1+a}{2}\right) \left(\frac{1-a}{2}\right) P_{l-2}^{2,2}(a) \left(\frac{1-b}{2}\right)^l \left(\frac{1-c}{2}\right)^l, \\
 N_{1m0}^{1-3(\text{aresta } 2)} &= \left(\frac{1+a}{2}\right) \left(\frac{1+b}{2}\right) \left(\frac{1-b}{2}\right) P_{m-1}^{3,2}(b) \left(\frac{1-c}{2}\right)^{m+1}, \\
 N_{1m0}^{1-3(\text{aresta } 3)} &= \left(\frac{1-a}{2}\right) \left(\frac{1+b}{2}\right) \left(\frac{1-b}{2}\right) P_{m-1}^{3,2}(b) \left(\frac{1-c}{2}\right)^{m+1}, \\
 N_{10n}^{1-3(\text{aresta } 4)} &= \left(\frac{1-a}{2}\right) \left(\frac{1-b}{2}\right) \left(\frac{1+c}{2}\right) \left(\frac{1-c}{2}\right) P_{n-1}^{4,2}(c), \\
 N_{10n}^{1-3(\text{aresta } 5)} &= \left(\frac{1+a}{2}\right) \left(\frac{1-b}{2}\right) \left(\frac{1+c}{2}\right) \left(\frac{1-c}{2}\right) P_{n-1}^{4,2}(c), \\
 N_{01n}^{1-3(\text{aresta } 6)} &= \left(\frac{1+b}{2}\right) \left(\frac{1+c}{2}\right) \left(\frac{1-c}{2}\right) P_{n-1}^{4,2}(c).
 \end{aligned} \tag{2.18}$$

- Funções de Face

$$\begin{aligned}
 N_{lm0}^{1-3(\text{face } 1)} &= \left(\frac{1+a}{2}\right) \left(\frac{1-a}{2}\right) P_{l-2}^{2,2}(a) \left(\frac{1+b}{2}\right) \left(\frac{1-b}{2}\right)^l P_{m-1}^{2l+1,2}(b) \left(\frac{1-c}{2}\right)^{l+m}, \\
 N_{l0n}^{1-3(\text{face } 2)} &= \left(\frac{1+a}{2}\right) \left(\frac{1-a}{2}\right) P_{l-2}^{2,2}(a) \left(\frac{1-b}{2}\right)^l \left(\frac{1+c}{2}\right) \left(\frac{1-c}{2}\right)^l P_{n-1}^{2l+2,2}(c), \\
 N_{lmn}^{1-3(\text{face } 3)} &= \left(\frac{1+a}{2}\right) \left(\frac{1+b}{2}\right) \left(\frac{1-b}{2}\right) P_{m-1}^{3,2}(b) \left(\frac{1+c}{2}\right) \left(\frac{1-c}{2}\right)^{m+1} P_{n-1}^{2m+4,2}(c),
 \end{aligned}$$

$$N_{lmn}^{1-3(\text{face } 4)} = \left(\frac{1-a}{2}\right) \left(\frac{1+b}{2}\right) \left(\frac{1-b}{2}\right) P_{m-1}^{3,2}(b) \left(\frac{1+c}{2}\right) \left(\frac{1-c}{2}\right)^{m+1} P_{n-1}^{2m+4,2}(c). \quad (2.19)$$

• Funções Internas

$$\begin{aligned} N_{lmn}^{1-3(\text{interior})} &= \left(\frac{1+a}{2}\right) \left(\frac{1-a}{2}\right) P_{l-2}^{2,2}(a) \left(\frac{1+b}{2}\right) \left(\frac{1-b}{2}\right)^l \\ &\quad \times P_{m-1}^{2l+1,2}(b) \left(\frac{1+c}{2}\right) \left(\frac{1-c}{2}\right)^{l+m} P_{n-1}^{2l+2m+2,2}(c). \end{aligned} \quad (2.20)$$

Analogamente ao caso **2D**, pode-se verificar as propriedades de ortogonalidade dessas funções tomando-se como exemplo o produto interno entre duas funções associadas ao interior do elemento de referência. Assim, efetuando-se o produto interno usual entre duas funções internas quaisquer sobre o tetraedro de referência T^3 , chega-se ao seguinte resultado:

$$\begin{aligned} (N_{lmn}^{1-3(\text{interior})}, N_{pqr}^{1-3(\text{interior})})_{T^3} &= \int_{-1}^1 \left(\frac{1+a}{2}\right)^2 \left(\frac{1-a}{2}\right)^2 P_{l-2}^{2,2}(a) P_{p-2}^{2,2}(a) da \\ &\quad \times \int_{-1}^1 \left(\frac{1+b}{2}\right)^2 \left(\frac{1-b}{2}\right)^{l+p} P_{m-1}^{2l+1,2}(b) P_{q-1}^{2p+1,2}(b) \left(\frac{1-b}{2}\right) db \\ &\quad \times \int_{-1}^1 \left(\frac{1+c}{2}\right)^2 \left(\frac{1-c}{2}\right)^{l+m+p+q} P_{n-1}^{2l+2m+2,2}(c) P_{r-1}^{2p+2q+2,2}(c) \left(\frac{1-c}{2}\right)^2 dc, \end{aligned} \quad (2.21)$$

sendo que os termos $\left(\frac{1-b}{2}\right)$ e $\left(\frac{1-c}{2}\right)^2$, nas duas últimas integrais, constituem o Jacobiano da transformação de R^3 para T^3 , i.e., $|J| = \left(\frac{1-b}{2}\right) \left(\frac{1-c}{2}\right)^2$. Dessa forma, tem-se

$$\begin{aligned} (N_{lmn}^{1-3(\text{interior})}, N_{pqr}^{1-3(\text{interior})})_{T^3} &= \left(\frac{1}{2}\right)^4 \int_{-1}^1 (1+a)^2 (1-a)^2 P_{l-2}^{2,2}(a) P_{p-2}^{2,2}(a) da \\ &\quad \times \left(\frac{1}{2}\right)^{l+p+3} \int_{-1}^1 (1+b)^2 (1-b)^{l+p+1} P_{m-1}^{2l+1,2}(b) P_{q-1}^{2p+1,2}(b) db \\ &\quad \times \left(\frac{1}{2}\right)^{l+m+p+q+4} \int_{-1}^1 (1+c)^2 (1-c)^{l+m+p+q+2} P_{n-1}^{2l+2m+2,2}(c) P_{r-1}^{2p+2q+2,2}(c) dc. \end{aligned} \quad (2.22)$$

Observa-se, a partir dessa relação, que $P_n^{\alpha,\beta}(x)$ são polinômios de *Jacobi* de ordem n no intervalo $[-1, 1]$. Novamente, pela propriedade de ortogonalidade (A.9) do Apêndice A.2, é possível concluir que, se $l \neq p$, então a primeira integral em (2.22) será nula pois

$$\int_{-1}^1 (1+a)^2 (1-a)^2 P_{l-2}^{2,2}(a) P_{p-2}^{2,2}(a) da = C_1 \delta_{lp}. \quad (2.23)$$

sendo C_1 uma constante que depende dos pesos α, β e da ordem polinomial n de $P_n^{\alpha,\beta}(x)$ (vide Apêndice A.2).

Se $l = p$ mas $m \neq q$, então a segunda integral em (2.22) será nula pois

$$\int_{-1}^1 (1+b)^2 (1-b)^{l+p+1} P_{m-1}^{2l+1,2}(b) P_{q-1}^{2p+1,2}(b) db = C_2 \delta_{mq} . \quad (2.24)$$

Finalmente, se $l = p$, $m = q$ mas $n \neq r$, então a terceira integral em (2.22) será nula pois

$$\int_{-1}^1 (1+c)^2 (1-c)^{l+m+p+q+2} P_{n-1}^{2l+2m+2,2}(c) P_{r-1}^{2p+2q+2,2}(c) dc = C_3 \delta_{nr} . \quad (2.25)$$

Dessa forma, nota-se que para qualquer função associada ao interior do elemento de referência, vale a seguinte relação de ortogonalidade

$$\left(N_{lmn}^{1-3(\text{interior})} , N_{pqr}^{1-3(\text{interior})} \right)_{T^3} = \bar{C} \left(\frac{1}{2} \right)^{2l+2p+m+q+11} \delta_{lp} \delta_{mq} \delta_{nr} . \quad (2.26)$$

sendo \bar{C} uma constante obtida a partir do produto das constantes C_1, C_2 e C_3 .

O padrão de esparsidade das matrizes de massa locais resultante da propriedade de ortogonalidade entre as funções internas desse elemento tetrédrico é análogo àquele obtido para o elemento triangular. Do mesmo modo que no exemplo apresentado acima, é possível observar outras relações de ortogonalidade combinando-se as funções internas, as de face e as de aresta propostas anteriormente.

2.3 Condicionamento de Matrizes Locais

A seguir, são apresentados resultados sobre o condicionamento das matrizes de rigidez e massa locais para os diferentes tipos de elementos triangulares e tetraédricos considerados anteriormente. O mesmo tipo de análise, para elementos quadrilaterais e hexaédricos, pode ser encontrado em (Babuška et al., 1989; Edgar e Surana, 1996). As matrizes de rigidez locais são obtidas a partir do operador de Laplace ($-\Delta u = f$) sobre um único triângulo ou tetraedro de referência. Calcula-se o número de condição das matrizes locais com base em três procedimentos distintos utilizados em (Zumbusch, 1995), (Webb e Abouchakra, 1995) e (Carnevali et al., 1993), respectivamente.

No primeiro caso, computa-se o número de condição das matrizes de rigidez através da seguinte relação $\kappa_1 = \frac{\max \mu}{\min \mu \neq 0}$, sendo $\max \mu$ o maior valor singular e $\min \mu$, o menor valor singular diferente de zero (ou mais próximo de zero) dessas matrizes, respectivamente. No caso das matrizes de massa, procede-se analogamente sem desconsiderar o valor singular mais próximo de zero. Esse modo de avaliação equivale a calcular o número de condição a partir da sua definição na norma

euclidiana (norma-2), i.e., $\kappa_1 = \|A\|_2 \|A^{-1}\|_2$ uma vez que as matrizes $[A]$ são reais, simétricas e definidas-positivas. Nessa situação, os valores singulares μ e os auto-valores λ das matrizes locais são idênticos.

No segundo caso, calcula-se o número de condição considerando o mesmo procedimento mas efetuando-se, desta vez, a seguinte mudança de escala nas matrizes locais

$$\tilde{A} = DAD,$$

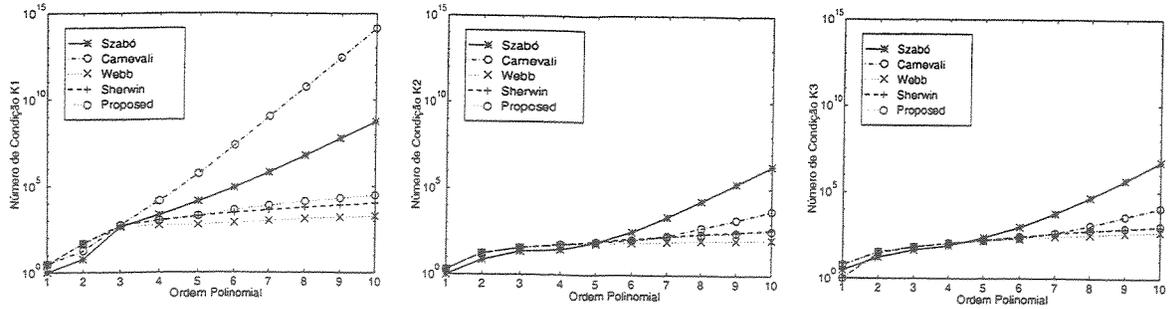
sendo D uma matriz diagonal cujos elementos são fornecidos pela relação $D_{ii} = \frac{1}{\sqrt{A_{ii}}}$, com A_{ii} dado pelos elementos diagonais das matrizes de rigidez e massa originais. Nesse caso, o número de condição será dado pela relação $\kappa_2 = \|\tilde{A}\|_2 \|\tilde{A}^{-1}\|_2$. Esse procedimento é equivalente a normalizar as funções de base multiplicando-as por fatores constantes. A mudança de escala nas matrizes locais permite tomar como parâmetro um conjunto de funções de base ideal (conjunto ortogonal) cujo número de condição seria 1 (Webb e Abouchakra, 1995).

Finalmente, utiliza-se a definição do número de condição considerando a norma-1, ou seja, $\kappa_3 = \|\tilde{A}\|_1 \|\tilde{A}^{-1}\|_1$. Repete-se novamente a mudança de escala sobre as matrizes de rigidez e massa locais. Remove-se ainda um grau de liberdade nodal das matrizes de rigidez para eliminar os modos de corpo rígido, o que equivale a tornar as matrizes $[\tilde{A}]$ não-singulares.

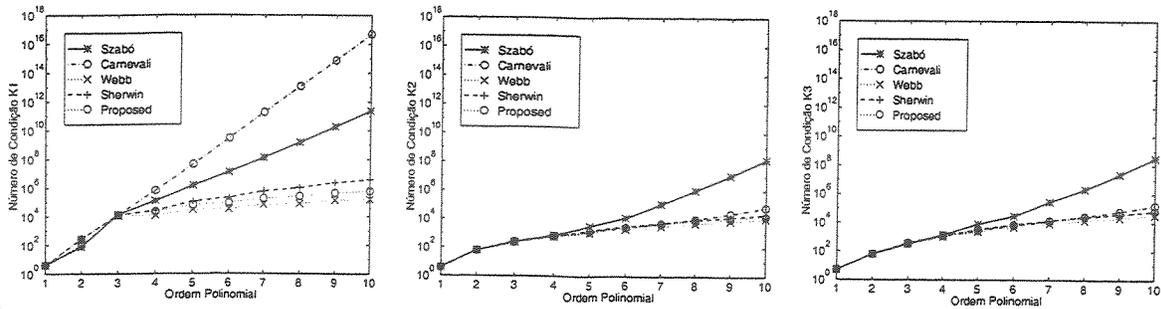
2.3.1 Triângulos

A Figura 2.7 apresenta os números de condição obtidos a partir das matrizes de rigidez e massa locais em função da ordem polinomial p das funções hierárquicas analisadas ($p = 1, \dots, 10$). Consideram-se as três metodologias de cálculo apresentadas.

Observa-se a partir desses gráficos que, para qualquer método de cálculo do número de condição, o conjunto de funções propostas em (Webb e Abouchakra, 1995) forneceu os melhores resultados. As funções de base apresentadas em (Sherwin e Karniadakis, 1995; Webb e Abouchakra, 1995) e aquelas propostas neste trabalho demonstraram, em todos os casos, um comportamento de estabilização do crescimento do número de condição com o aumento da ordem polinomial p . Nota-se que o desempenho apresentado pelas funções de base fornecidas em (Carnevali et al., 1993) é fortemente dependente do procedimento de normalização. Confirmou-se ainda a tendência de de-



(a) Matriz de Rigidez.



(b) Matriz de Massa.

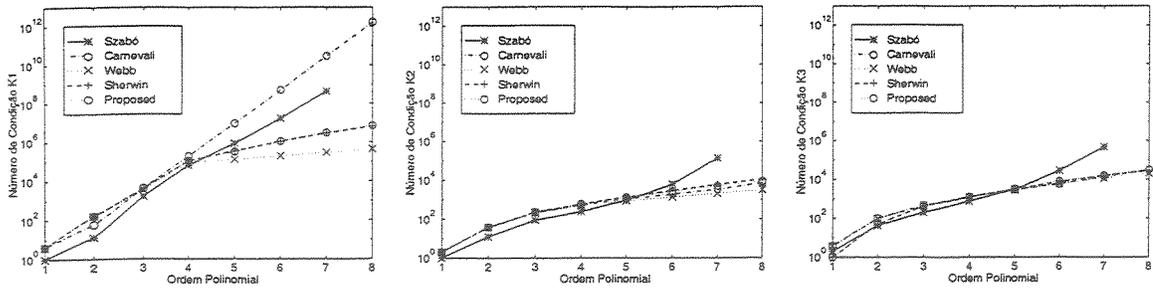
Figura 2.7: Número de condição das matrizes de rigidez e massa locais vs. ordem polinomial p , considerando-se as definições de κ_1 , κ_2 e κ_3 (triângulos).

geração do número de condição das funções clássicas propostas em (Szabó e Babuška, 1991) com o aumento da ordem polinomial p . Os mesmos comentários feitos em relação ao condicionamento das matrizes de rigidez locais se aplicam ao caso das matrizes de massa.

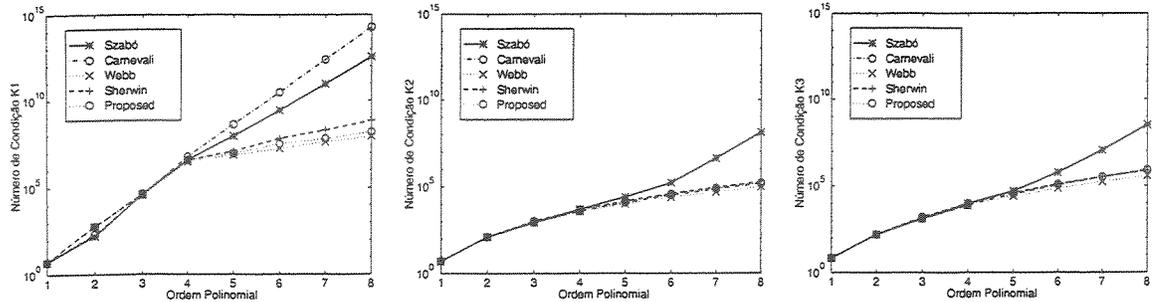
2.3.2 Tetraedros

A Figura 2.8 apresenta os números de condição obtidos a partir das matrizes de rigidez e massa locais em função da ordem polinomial p das funções hierárquicas analisadas ($p = 1, \dots, 8$). Consideram-se as três metodologias de cálculo apresentadas.

Observa-se a partir desses gráficos que, para a segunda e terceira metodologias de cálculo do número de condição, com exceção das funções sugeridas em (Szabó e Babuška, 1991), todas as demais apresentaram um desempenho bastante similar, demonstrando um comportamento de estabilização do crescimento do número de condição com o aumento da ordem polinomial p . Analogamente ao



(a) Matriz de Rigidez.



(b) Matriz de Massa.

Figura 2.8: Número de condição das matrizes de rigidez e massa locais vs. ordem polinomial p , considerando-se as definições de κ_1 , κ_2 e κ_3 (tetraedros).

caso dos elementos triangulares, empregando-se a primeira metodologia de cálculo do número de condição, é possível notar que o desempenho das funções de base fornecidas por (Carnevali et al., 1993) é fortemente dependente do procedimento de normalização. Do mesmo modo, constata-se a tendência de degeneração do número de condição das funções clássicas propostas por (Szabó e Babuška, 1991) com o aumento da ordem polinomial p . Os comentários relativos às matrizes de massa são idênticos aos que acabam de ser apresentados para o caso das matrizes de rigidez.

2.4 Esparsidade de Matrizes Locais

A seguir são apresentados resultados sobre o grau de esparsidade das matrizes de rigidez e massa locais para os diferentes tipos de elementos triangulares e tetraédricos considerados previamente. Observa-se que os coeficientes das matrizes de rigidez e massa locais foram obtidas de forma exata a partir do programa de manipulação simbólica MATHEMATICA 3.0. A não utilização

de procedimentos numéricos de integração permite uma avaliação precisa da ortogonalidade entre as funções hierárquicas analisadas e conseqüentemente da esparsidade das matrizes locais. Assim como na análise do condicionamento, as matrizes de rigidez locais também são obtidas a partir da aplicação do operador de *Laplace* sobre o triângulo ou tetraedro de referência. O grau de esparsidade é computado como o inverso da densidade do elemento (Carnevali et al., 1993), i.e., a fração de elementos nulos presente nas matrizes de rigidez e massa locais.

2.4.1 Triângulos

Na Figura 2.9, a porcentagem de elementos nulos das matrizes de rigidez e massa locais é mostrada em função da ordem polinomial p das funções hierárquicas analisadas ($p = 1, \dots, 10$).

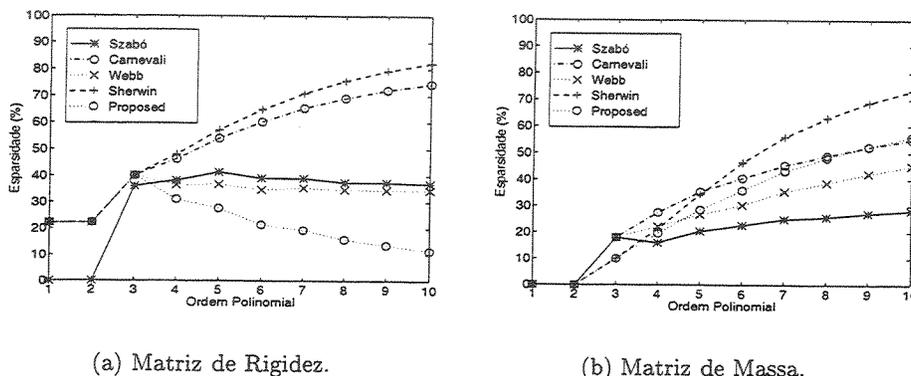
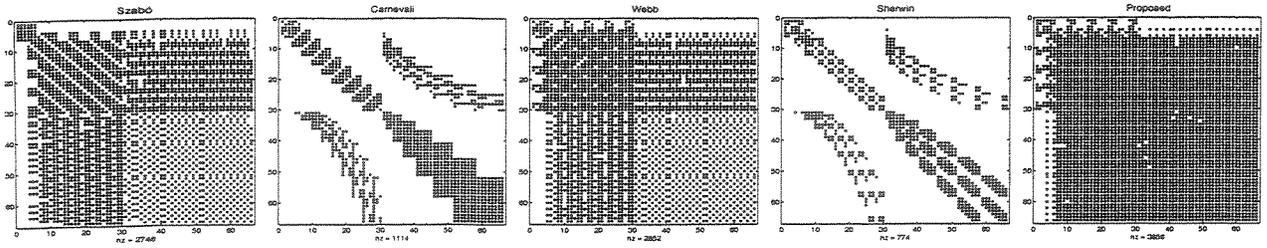


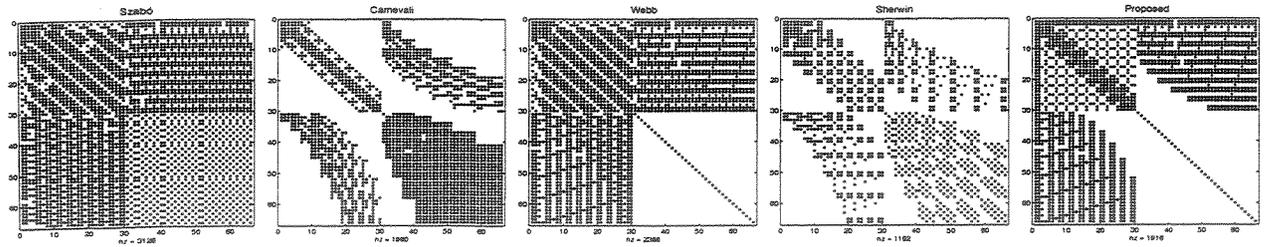
Figura 2.9: Porcentagem de elementos nulos das matrizes de rigidez e massa locais vs. ordem polinomial p (triângulos).

Observa-se que os maiores níveis de esparsidade com o aumento da ordem polinomial p acontecem para o conjunto de funções de forma proposto por (Sherwin e Karniadakis, 1995). Esse fato demonstra que os polinômios de *Jacobi*, tal como empregados nessa formulação, conferem uma acentuada ortogonalidade a essas funções de forma. Da mesma maneira, verifica-se que essa propriedade de ortogonalidade é preservada quando se considera o operador de *Laplace* (matrizes de rigidez) em lugar do operador de interpolação (matrizes de massa). Ou seja, as derivadas dessas funções também possuem um alto grau de ortogonalidade.

A Figura 2.10 mostra o padrão de esparsidade encontrado nas matrizes de rigidez e massa locais associadas aos cinco tipos de funções hierárquicas estudadas.



(a) Matriz de Rigidez.



(b) Matriz de Massa.

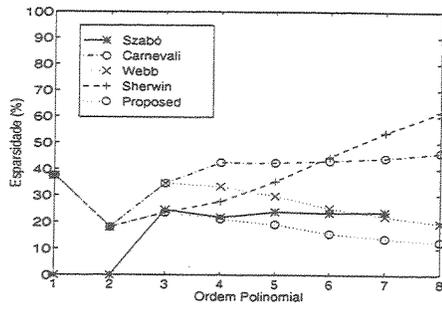
Figura 2.10: Padrão de esparsidade das matrizes de rigidez e massa locais associadas aos cinco tipos de funções hierárquicas estudadas (triângulos).

2.4.2 Tetraedros

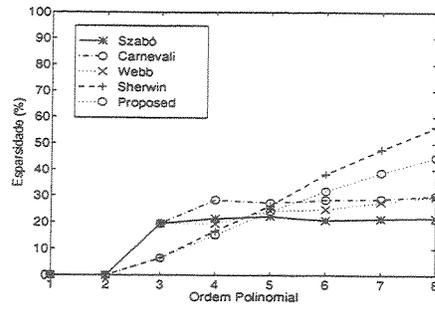
Na Figura 2.11, a porcentagem de elementos nulos das matrizes de rigidez e massa locais é mostrada em função da ordem polinomial p das funções hierárquicas analisadas ($p = 1, \dots, 8$).

Observa-se novamente que em ambos os casos os maiores níveis de esparsidade com o aumento da ordem polinomial p ocorrem para o conjunto de funções proposto em (Sherwin e Karniadakis, 1995).

A Figura 2.12 mostra o padrão de esparsidade encontrado nas matrizes de rigidez e massa locais associadas aos cinco tipos de funções hierárquicas estudadas.

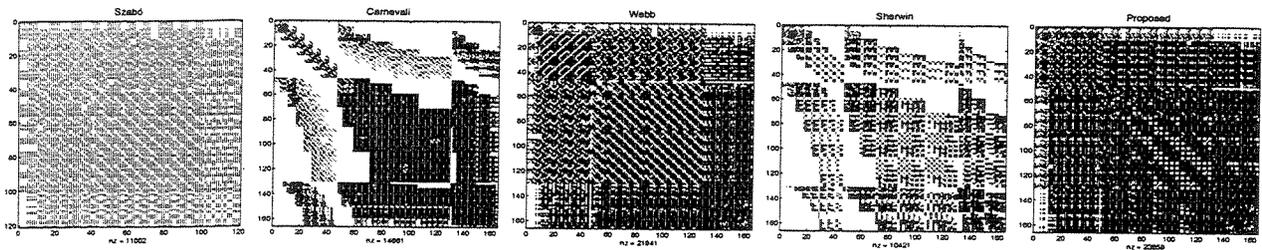


(a) Matriz de Rigidez.

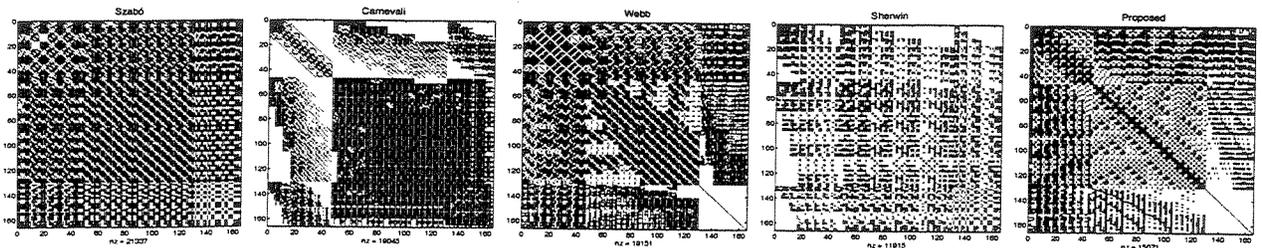


(b) Matriz de Massa.

Figura 2.11: Porcentagem de elementos nulos das matrizes de rigidez e massa locais vs. ordem polinomial p (tetraedros).



(a) Matriz de Rigidez.



(b) Matriz de Massa.

Figura 2.12: Padrão de esparsidade das matrizes de rigidez e massa locais associadas aos cinco tipos de funções hierárquicas estudados (tetraedros).

2.5 Esparsidade e Condicionamento de Matrizes Locais sob o Efeito de Distorção

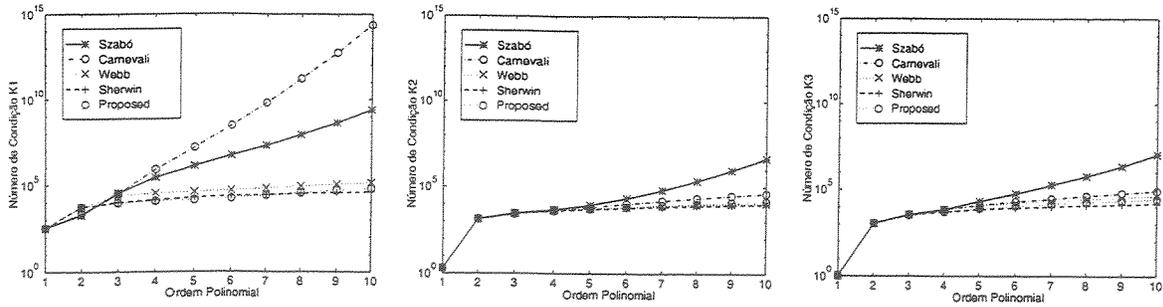
Apresenta-se, a seguir, uma análise sobre o efeito da distorção dos elementos nos níveis de esparsidade e condicionamento das matrizes de rigidez estudadas. A análise se resume às distorções do tipo linear uma vez que na maior parte das situações os elementos da malha são descritos por funções de mapeamento lineares (Carnevali et al., 1993). A mesma análise não se repete para as matrizes de massa já que as distorções lineares não alteram suas características de esparsidade e condicionamento (funções de mapeamento lineares produzem jacobianos de transformação constantes de modo que as matrizes de massa locais, para os elementos distorcidos e não-distorcidos, diferem somente por um fator constante). Emprega-se a mesma metodologia de análise utilizada anteriormente considerando-se, agora, dois casos de distorção: alongamento (dimensões de base e altura na proporção 1 : 16 para triângulos e dimensões de comprimento, largura e altura na proporção 1 : 1 : 16 para tetraedros) e achatamento (dimensões na proporção 16 : 1 para triângulos e 16 : 1 : 1 para tetraedros).

2.5.1 Triângulos

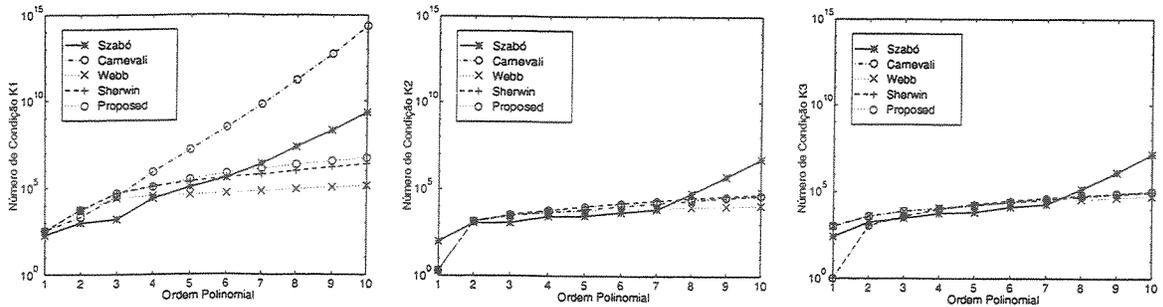
A Figura 2.13 apresenta os números de condição obtidos a partir das matrizes de rigidez locais em função da ordem polinomial p das funções hierárquicas analisadas ($p = 1, \dots, 10$). Consideram-se as três metodologias de cálculo adotadas anteriormente e em cada uma delas o primeiro e o segundo casos de distorção.

Observa-se, a partir dos dois últimos conjuntos de gráficos, uma elevação global dos números de condição das matrizes relativamente ao comportamento demonstrado na Figura 2.7. Nota-se ainda que as menores sensibilidades à distorção linear, para o caso de alongamento, foram apresentadas pelas funções sugeridas em (Sherwin e Karniadakis, 1995) e aquelas propostas neste trabalho. Para o caso de achatamento, as menores sensibilidades à distorção foram demonstradas pelas funções sugeridas em (Webb e Abouchakra, 1995).

A Figura 2.14 apresenta a porcentagem de elementos nulos das matrizes de rigidez locais em função da ordem polinomial p das funções hierárquicas analisadas ($p = 1, \dots, 10$) quando se



(a) Alongamento.



(b) Achatamento.

Figura 2.13: Número de condição das matrizes de rigidez locais vs. ordem polinomial p considerando-se as definições de κ_1 , κ_2 e κ_3 , e ambos os casos de distorção: (a) alongamento e (b) achatamento (triângulos).

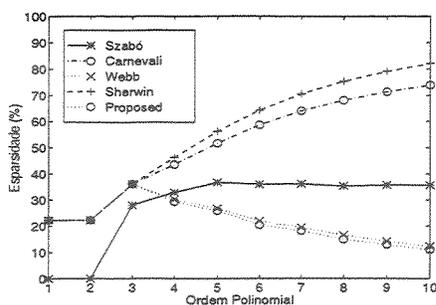
consideram os dois casos de distorção.

Comparativamente à Figura 2.9, é possível notar que apenas o elemento sugerido por (Sherwin e Karniadakis, 1995) não apresentou queda nos níveis de esparsidade sob o efeito das distorções.

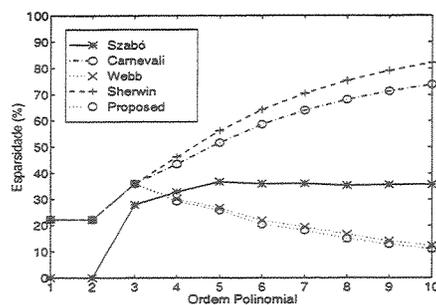
2.5.2 Tetraedros

A Figura 2.15 apresenta os números de condição obtidos a partir das matrizes de rigidez locais em função da ordem polinomial p das funções hierárquicas analisadas ($p = 1, \dots, 8$). Consideram-se as três metodologias de cálculo adotadas anteriormente e em cada uma delas o primeiro e o segundo casos de distorção.

Assim como no caso dos elementos triangulares, observa-se uma elevação global dos números de condição das matrizes de rigidez locais em relação ao comportamento demonstrado na Figura

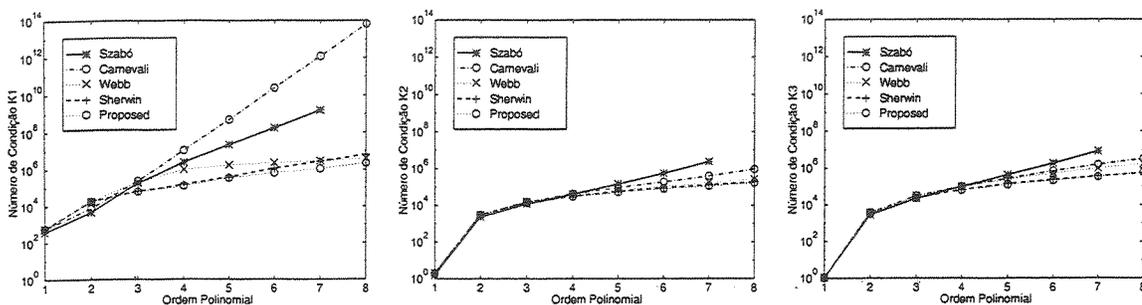


(a) Alongamento.

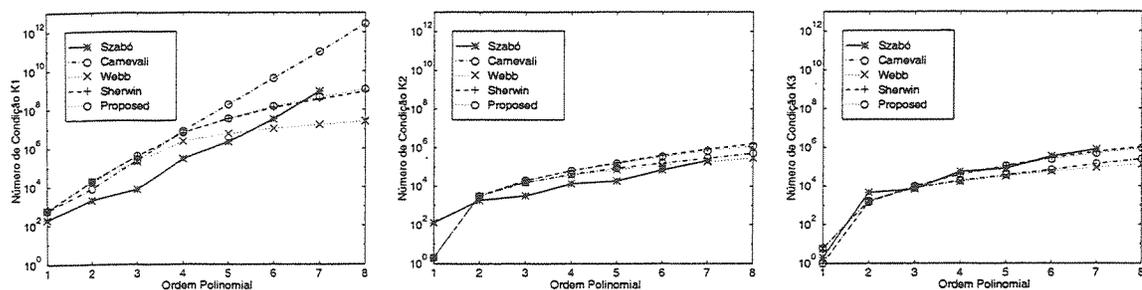


(b) Achatamento.

Figura 2.14: Porcentagem de elementos nulos das matrizes de rigidez locais vs. ordem polinomial p : (a) triângulos padrões alongados; (b) triângulos padrões achatados.



(a) Alongamento.



(b) Achatamento.

Figura 2.15: Número de condição das matrizes de rigidez locais vs. ordem polinomial p considerando-se as definições de κ_1 , κ_2 e κ_3 , e ambos os casos de distorção: (a) alongamento e (b) achatamento (tetraedros).

2.8. Da mesma maneira, nota-se que os efeitos da distorção nos elementos tetraédricos é análogo àquele exibido pelos elementos triangulares.

A Figura 2.16 apresenta a porcentagem de elementos nulos das matrizes de rigidez locais em função da ordem polinomial p das funções hierárquicas analisadas ($p = 1, \dots, 8$) quando se consideram os dois casos de distorção.

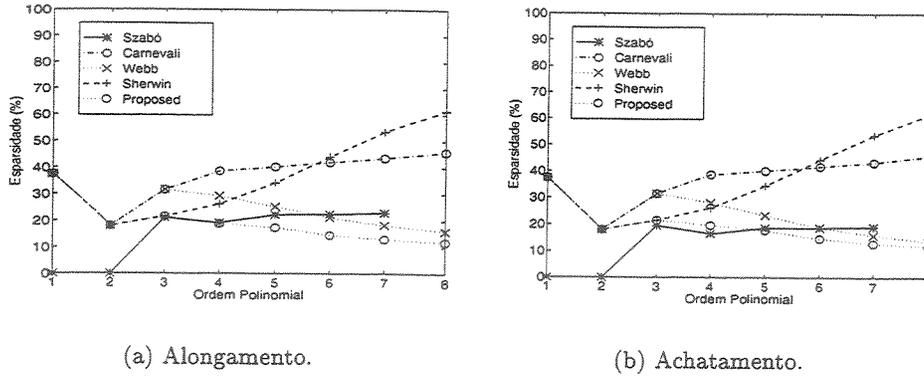


Figura 2.16: Porcentagem de elementos nulos das matrizes de rigidez locais vs. ordem polinomial p : (a) tetraedros padrões alongados; (b) tetraedros padrões achatados.

Comparativamente à Figura 2.11, é possível notar que as distorções provocaram, em geral, uma queda nos níveis de esparsidade das matrizes de rigidez locais, tendo sido o elemento proposto por em (Sherwin e Karniadakis, 1995) o menos afetado.

2.6 Condicionamento de Matrizes Globais para o Problema de *Poisson* 2D

Apresentam-se, a seguir, resultados numéricos sobre o condicionamento de matrizes de rigidez globais obtidas a partir dos diferentes conjuntos de funções de base analisados acima (Nogueira Jr. e Bittencourt, 1999). As matrizes globais foram geradas com base na solução do problema de *Poisson* com condições de contorno de *Dirichlet* homogêneas

$$\begin{aligned} \Delta u + f &= 0 \quad \text{em } \Omega, \\ u &= 0 \quad \text{em } \partial\Omega, \end{aligned} \tag{2.27}$$

sendo Ω um domínio quadrado ou em formato L (vide Figura 2.18) e f , um termo de carga especialmente escolhido para produzir acentuadas variações locais na solução u devido a singularidades ou gradientes intensos (Novotny et al., 1999) (vide seção seguinte).

Os números de condição são obtidos calculando-se $\kappa_1 = \frac{\max \lambda}{\min \lambda}$, sendo $\max \lambda$ e $\min \lambda$ o maior e o menor auto-valor das matrizes de rigidez globais (Zumbusch, 1995). Esse procedimento é realizado com e sem a utilização de um pré-condicionamento diagonal (o pré-condicionamento é equivalente a mudança de escala feita localmente).

A Figura 2.17 mostra os números de condição obtidos a partir das matrizes de rigidez globais em função da ordem polinomial das funções de aproximação ($p = 2, 3, \dots, 8$) considerando-se o problema (2.27) definido sobre um domínio quadrado.

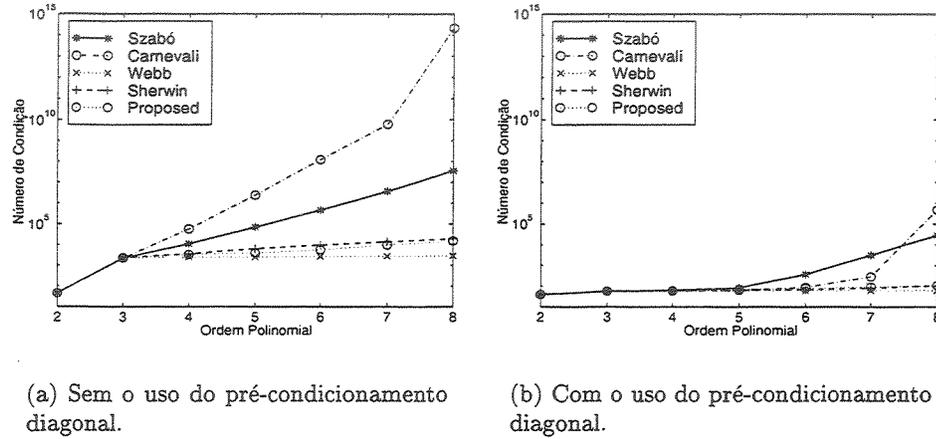


Figura 2.17: Número de condição das matrizes de rigidez globais vs. ordem polinomial p .

Observa-se a partir desses gráficos que houve uma razoável correspondência entre os resultados obtidos globalmente e aqueles obtidos localmente (Nogueira Jr. e Bittencourt, 1999). Essa constatação indica que os resultados teóricos sugeridos em (Zumbusch, 1995) para a obtenção de uma estimativa do número de condição global a partir do número de condição local é bastante aceitável. Esse autor conjectura a seguinte estimativa para valores uniformes de h (diâmetro dos elementos)

$$\kappa(A_{glob}) \leq Ch^{-2} \frac{\max \lambda_{\max}(A_{loc})}{\min \lambda_{\min}(A_{loc})} \quad (2.28)$$

sendo C uma constante que depende: (i) do menor ângulo interno γ_{\min} encontrado entre todos os elementos da malha de discretização e, (ii) da família de funções de base escolhida. Analogamente

ao caso global, $\lambda_{\max}(A_{loc})$ e $\lambda_{\min}(A_{loc})$ representam o maior e o menor autovalores das matrizes locais considerando-se todos os elementos da malha.

Nota-se, finalmente, que as funções propostas em (Webb e Abouchakra, 1995) apresentaram os melhores resultados para esse tipo de análise.

2.7 Solução do Problema de *Poisson* 2D Usando o Método de Gradientes Conjugados

Com o propósito de validar os estudos sobre condicionamento local (Nogueira Jr. e Bittencourt, 1999) no caso da utilização de métodos iterativos de solução de sistemas algébricos, são exibidos adiante resultados numéricos sobre a eficiência do método de gradientes conjugados (CG) quando aplicado a solução do problema (2.27) usando-se as diferentes famílias de funções de base 2D discutidas anteriormente.

O problema (2.27) é analisado em duas situações, a primeira, considerando-se um domínio Ω quadrado e a segunda, considerando-se um domínio Ω em formato L (Figura 2.18).

Os termos de carga foram escolhidos de modo que a solução do problema apresentasse gradientes intensos no primeiro caso e fosse suave no segundo, tal como em (Novotny et al., 1999), já que nesta última situação o problema apresenta uma singularidade geométrica.

Para o caso do domínio Ω quadrado, a solução analítica do problema é dada por (vide Figura 2.19)

$$u(x, y) = xy(1-x)(1-y) \tan^{-1} \left[60 \left(\frac{x+y}{\sqrt{2}} - 0.8 \right) \right]. \quad (2.29)$$

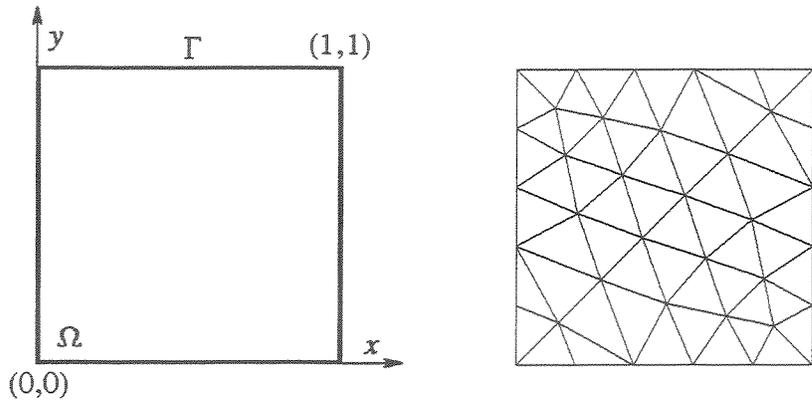
No caso do domínio Ω em formato L , a solução analítica é dada por (vide Figura 2.20)

$$u(x, y) = \frac{3}{2} r^{2/3} \sin \frac{2\theta}{3} (1-x^2)(1-y^2), \quad (2.30)$$

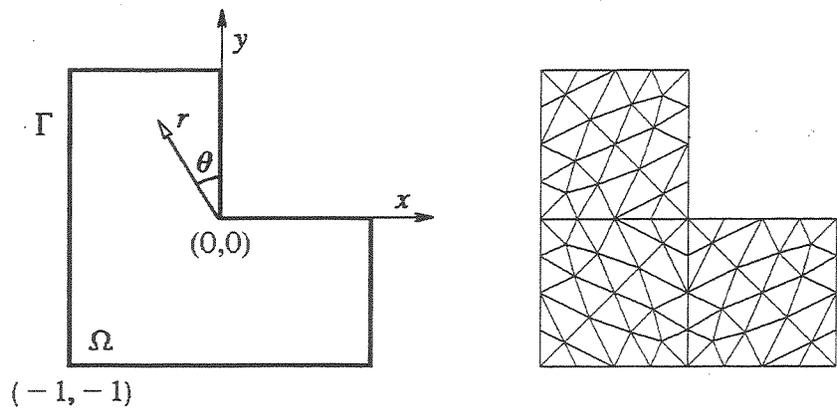
com $r = \sqrt{x^2 + y^2}$ e $\theta = \tan^{-1} \left(-\frac{x}{y} \right)$.

Em ambas as situações, foi considerado o método de gradientes conjugados padrão e com pré-condicionamento diagonal.

A Figura 2.21 mostra o número de iterações necessárias para que o método de gradientes

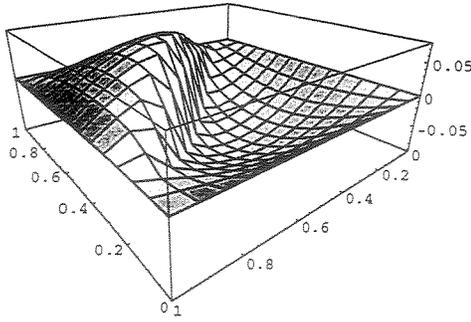


(a) Problema 1: Ω quadrado; malha 54 elementos.

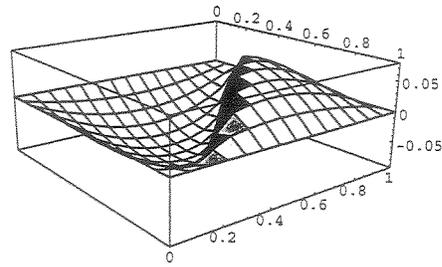


(b) Problema 2: Ω em L ; malha 108 elementos.

Figura 2.18: Domínios e malhas utilizadas nos dois tipos de problemas analisados (Novotny et al., 1999).

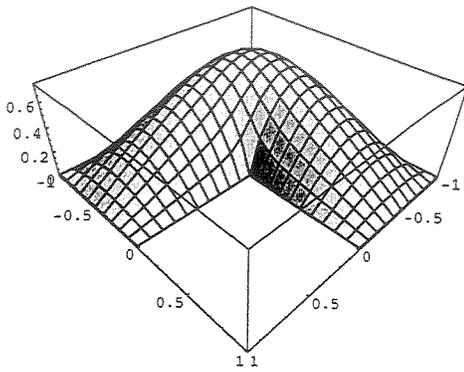


(a) vista (-1.0,0.8,0.5).

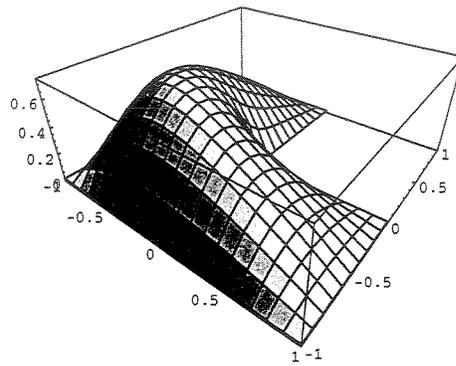


(b) vista (2.0,1.6,0.8).

Figura 2.19: Solução analítica do problema de *Poisson* para Ω quadrado (Novotny et al., 1999).



(a) vista (1.0,1.0,1.0).



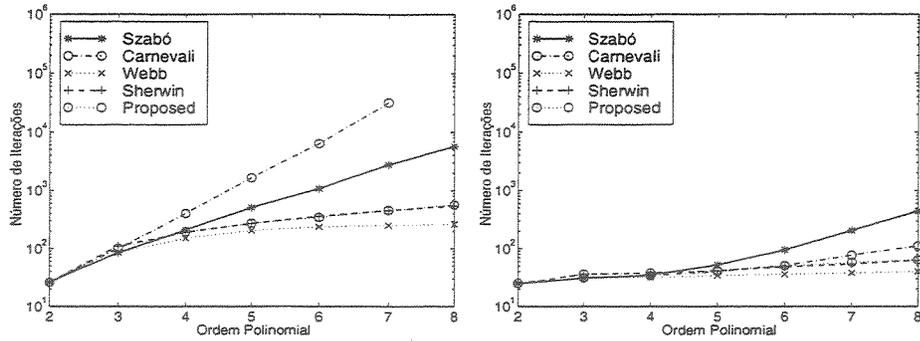
(b) vista (0.75,-1.0,0.75).

Figura 2.20: Solução analítica do problema de *Poisson* para Ω em formato L (Novotny et al., 1999).

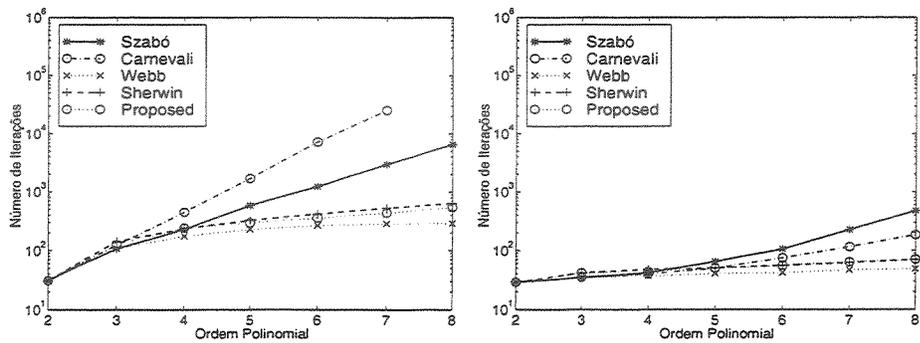
conjugados atinja convergência (com precisão de erro $\varepsilon < 10^{-6}$) em função da ordem polinomial das funções de aproximação ($p = 2, 3, \dots, 8$).

Observa-se que esses gráficos guardam estreita semelhança com aqueles obtidos para o condicionamento global das matrizes de rigidez do item anterior, bem como para os das matrizes locais discutidas anteriormente (Nogueira Jr. e Bittencourt, 1999). Assim, considerando-se a estimativa de condicionamento global apresentada na seção anterior (Zumbusch, 1995) e a proporcionalidade entre o número de iterações gerado pelo método de gradientes conjugados ($NIT(\varepsilon)$) e o número de condição das matrizes de rigidez globais ($\kappa(A_{glob})$) (Axelsson e Barker, 1984)

$$NIT(\varepsilon) \leq \frac{1}{2} \sqrt{\kappa(A_{glob})} \ln(2/\varepsilon) + 1, \quad (2.31)$$



(a) Problema 1: Ω quadrado



(b) Problema 2: Ω em L

Figura 2.21: Número de iterações executadas pelo método GC nas versões padrão e com pré-condicionamento diagonal vs. ordem polinomial p .

sendo ε a tolerância que se deseja alcançar, é possível constatar a legitimidade do estudo sobre o condicionamento local das funções de base hierárquicas. Isso comprova ainda que a análise local fornece informações precisas a respeito do desempenho dessas funções quando se empregam métodos iterativos na solução de sistemas algébricos oriundos da discretização de problemas elípticos lineares 2D.

2.8 Conclusões

Dentro do conjunto de análises conduzido neste capítulo, pode-se dizer que, relativamente ao condicionamento, as funções hierárquicas associadas a triângulos e tetraedros sugeridas por (Webb e Abouchakra, 1995; Abouchakra, 1996) demonstraram, na maioria dos casos, os melhores

resultados. Essas funções apresentaram, em geral, as menores taxas de crescimento do número de condição com o aumento da ordem polinomial p . Ainda dentro desse tipo de análise, observou-se que as funções hierárquicas propostas por (Sherwin e Karniadakis, 1995) e aquelas sugeridas neste trabalho também alcançaram resultados expressivos, com a vantagem sobre as funções encontradas em (Webb e Abouchakra, 1995; Abouchakra, 1996) de não necessitarem de procedimentos especiais de integração numérica. No que diz respeito às análises sobre esparsidade, as funções propostas em (Sherwin e Karniadakis, 1995) foram nitidamente superiores em todos os casos.

Tendo em vista que os padrões de condicionamento e esparsidade das matrizes locais influenciam os das matrizes globais (Carnevali et al., 1993; Edgar e Surana, 1996; Zumbusch, 1995) e que essas propriedades, por sua vez, são de extrema relevância para a solução de sistemas lineares através de métodos iterativos, os resultados obtidos neste capítulo demonstraram-se muito úteis no sentido de orientar a escolha de um conjunto de funções de base adequado ao emprego das versões p e hp do **MEF** associadas ao uso de métodos iterativos e multigrid algébricos.

Capítulo 3

Formulação p em 3 Dimensões Aplicada à Elasticidade Linear e Não-Linear

O capítulo precedente demonstrou as vantagens da utilização de uma base hierárquica construída a partir dos polinômios de *Jacobi* de peso misto ao se considerar aspectos como o condicionamento e a esparsidade das matrizes geradas na formulação p do MEF. Além do tipo de polinômio que constitui a base do espaço de aproximação, outra característica fundamental para se assegurar a eficiência e a flexibilidade da formulação é à capacidade de se executar operações de integração de alta ordem sobre os elementos que compõem a malha de discretização do problema. Nesse sentido, e particularmente no caso de malhas não-estruturadas 3D, as bases tensorizáveis mostram-se superiores àquelas definidas em coordenadas baricêntricas uma vez que se pode aproveitar diretamente as regras de integração numérica usuais construídas sobre intervalos unidimensionais. Isso oferece (a menos da capacidade de processamento do *hardware* utilizado) maior flexibilidade na integração exata de funções polinomiais de qualquer ordem, ao contrário do que ocorre no caso das regras de quadratura simétricas definidas em coordenadas baricêntricas (Jinyun, 1984). O aumento na quantidade de pontos de integração em relação a esse último tipo de regra de quadratura pode ser compensado pela utilização de algoritmos de manipulação algébrica eficientes como os baseados na técnica de *soma fatorada* (Karniadakis e Sherwin, 1999).

Desse modo, a formulação p para malhas **3D** não-estruturadas considerada neste trabalho segue a proposta de (Sherwin e Karniadakis, 1996) originalmente concebida para a aplicação em problemas de mecânica dos fluidos computacional. Este capítulo é inicialmente dedicado à apresentação dos elementos essenciais presentes na formulação p **3D** escolhida. Uma vez tendo sido apresentada a formulação, a segunda parte deste capítulo considera a aplicação dessa formulação a problemas de elasticidade linear e não-linear.

A seção 1 deste capítulo se concentra nos procedimentos relacionados às operações locais tais como integração e diferenciação numéricas além do mapeamento de funções de base. A seção 2 apresenta os procedimentos relacionados às operações globais como a obtenção de conectividade em regiões inter-elementos e a numeração de graus de liberdade. A seção 3 descreve os procedimentos relacionados à aplicação de condições de contorno. A seção 4 traz a aplicação da formulação apresentada em problemas de elasticidade linear. A seção 5 estende a aplicação da seção anterior a problemas de elasticidade não-linear. Finalmente, a seção 6 sumariza os resultados obtidos apresentando as conclusões a respeito do que foi discutido ao longo do capítulo.

3.1 Operações Locais

3.1.1 Integração numérica sobre regiões tetraédricas

Uma maneira eficiente de se executar integrações numéricas de funções polinomiais sobre intervalos unidimensionais é a utilização de regras de quadratura *Gaussianas*. A extensão desse tipo de regra para domínios **3D** estruturados (elementos hexaédricos) pode ser feita de forma trivial através da tensorização dos pontos e pesos de integração. O mesmo não ocorre para domínios **3D** não-estruturados (elementos tetraédricos) uma vez que os limites de integração, nesse caso, não possuem valores constantes. No entanto, a transformação de uma região tetraédrica numa região cujas fronteiras possuem limites constantes é equivalente a se mapear um domínio tetraédrico sobre um domínio hexaédrico, exatamente como foi feito no caso da definição das funções de base associadas a elementos tetraédricos sugeridas em (Sherwin e Karniadakis, 1995) (vide Figura 2.6).

A definição de um sistema de coordenadas colapsadas, nesse caso, torna conveniente o uso de quadraturas do tipo *Gauss*. A integração no sistema de coordenadas colapsadas mantém a precisão

alcançada por esse tipo de quadratura já que a forma como as funções de base foram definidas garante que estas últimas sejam polinômios tanto no sistema cartesiano quanto no colapsado. Embora existam esquemas de integração numérica alternativos baseados em coordenadas baricêntricas (Jinyun, 1984) que necessitam de uma quantidade menor de pontos para uma mesma ordem de integração, esses esquemas, em geral, estão restritos a integrações de ordem moderada ($p = 8$ no máximo) em razão da dificuldade de se obter os pesos de quadratura através de procedimentos numéricos (a obtenção dos pesos e pontos de integração depende da solução de equações algébricas altamente não-lineares).

Assim, o tipo de integração numérica escolhido para o domínio tetraédrico $T^3 = \{(\xi, \eta, \zeta) \mid -1 \leq \xi, \eta, \zeta; \xi + \eta + \zeta \leq -1\}$ adota o sistema de coordenadas colapsado definido pelas seguintes transformações (vide Figura 2.6)

$$a = 2 \frac{(1 + \xi)}{(-\eta - \zeta)} - 1, \quad b = 2 \frac{(1 + \eta)}{(1 - \zeta)} - 1, \quad c = \zeta, \quad (3.1)$$

sendo as respectivas transformações inversas dadas por

$$\xi = \frac{1}{4}(1 + a)(1 - b)(1 - c) - 1, \quad \eta = \frac{1}{2}(1 + b)(1 - c) - 1, \quad \zeta = c. \quad (3.2)$$

No caso da discretização de problemas de elasticidade linear e não-linear, normalmente é necessário se efetuar integrações do seguinte tipo

$$\int_{T^3} u(\xi, \eta, \zeta) d\xi d\eta d\zeta = \int_{-1}^1 \int_{-1}^1 \int_{-1}^1 u(a, b, c) J da db dc, \quad (3.3)$$

sendo J , o Jacobiano de transformação $R^3 \rightarrow T^3$ dado por

$$J = \frac{\partial(\xi, \eta, \zeta)}{\partial(a, b, c)} = \left(\frac{1 - b}{2}\right) \left(\frac{1 - c}{2}\right)^2. \quad (3.4)$$

O segundo termo em (3.3) pode ser aproximado numericamente usando-se regras de quadratura unidimensionais tensorizadas

$$\begin{aligned} & \int_{-1}^1 \int_{-1}^1 \int_{-1}^1 u(a, b, c) \left(\frac{1 - b}{2}\right) \left(\frac{1 - c}{2}\right)^2 da db dc \\ &= \sum_{i=0}^{Q_1-1} w_i \left\{ \sum_{j=0}^{Q_2-1} w_j \left\{ \sum_{k=0}^{Q_3-1} w_k u(a_i, b_j, c_k) \left(\frac{1 - b_j}{2}\right) \left(\frac{1 - c_k}{2}\right)^2 \right\} \right\}, \end{aligned} \quad (3.5)$$

sendo a_i, b_j e c_k as coordenadas tensorizadas dos Q_1, Q_2 e Q_3 pontos de quadratura em cada dimensão espacial. Nesse caso, os pesos w_i, w_j e w_k correspondem aos pesos das regras de *Gauss-Legendre* padrões pois o termo associado ao Jacobiano de transformação é parte do integrando, não

tendo sido incluído nos próprios pesos de integração (vide Apêndice B).

A inclusão da expressão do Jacobiano nos pesos que compõem a regra de quadratura pode ser conseguida utilizando-se uma quadratura do tipo *Gauss-Jacobi*

$$\int_{-1}^1 (1-z)^\alpha (1+z)^\beta f(z) dz = \sum_{i=0}^{Q-1} w_i^{\alpha,\beta} f(z_i^{\alpha,\beta}), \quad (3.6)$$

sendo $w_i^{\alpha,\beta}$ e $z_i^{\alpha,\beta}$ os pesos e pontos correspondentes a escolha de α e β . Nesse caso, se $\alpha = \beta = 0$ a regra padrão de *Gauss-Legendre* é automaticamente recuperada. A quadratura do tipo *Gauss-Jacobi* pode ainda ser gerada a partir da distribuição de pontos de *Lobatto* ou *Radau* de modo a permitir a inclusão total ou parcial, respectivamente, dos pontos pertencentes ao contorno do domínio de integração (Karniadakis e Sherwin, 1999).

Dessa forma, nota-se que é possível simplificar a expressão (3.5) ao se escolher $\alpha = 0, \beta = 0$ para a coordenada a , $\alpha = 1, \beta = 0$ para b e $\alpha = 2, \beta = 0$ para c incluindo, assim, a contribuição do termo associado ao Jacobiano diretamente nos pesos de integração. A integração sobre T^3 em (3.5) pode então ser reescrita como

$$\begin{aligned} & \int_{-1}^1 \int_{-1}^1 \int_{-1}^1 u(a, b, c) \left(\frac{1-b}{2} \right) \left(\frac{1-c}{2} \right)^2 da db dc \\ &= \sum_{i=0}^{Q_1-1} w_i^{0,0} \left\{ \sum_{j=0}^{Q_2-1} \hat{w}_j^{1,0} \left\{ \sum_{k=0}^{Q_3-1} \hat{w}_k^{2,0} u(a_i^{0,0}, b_j^{1,0}, c_k^{2,0}) \right\} \right\}, \end{aligned} \quad (3.7)$$

sendo

$$\hat{w}_j^{1,0} = \frac{w_j^{1,0}}{2}, \quad \hat{w}_k^{2,0} = \frac{w_k^{2,0}}{4} \quad (3.8)$$

e Q_1, Q_2 e Q_3 o número de pontos de quadratura nas direções a, b e c , respectivamente. Observa-se ainda que esse tipo de integração numérica sobre o tetraedro T^3 usa menos pontos de quadratura que a regra padrão de *Gauss-Legendre* para atingir uma precisão equivalente uma vez que a inclusão da expressão do Jacobiano nos pesos da regra de quadratura diminuiu a ordem polinomial do integrando.

Ao se escolher a distribuição dos pontos de integração, nenhuma restrição precisa ser feita quanto ao tipo empregado: *Gauss-Legendre*, *Gauss-Radau* ou *Gauss-Lobatto*. No entanto, neste último caso (*Gauss-Lobatto*), o uso de pontos localizados no bordo do domínio de integração T^3 acarretará o aparecimento de pontos múltiplos sobre os vértices ($\xi = -1, \eta = -1, \zeta = 1$) e ($\xi = -1, \eta = 1, \zeta = -1$) bem como sobre a aresta que liga esses dois vértices. Para se evitar esse tipo

de redundância, pode-se usar uma distribuição mista para os pontos de quadratura como mostrado na Figura 3.1 em que se usa a quadratura de *Gauss-Radau* nas direções η, ζ e de *Gauss-Lobatto* na direção ξ .

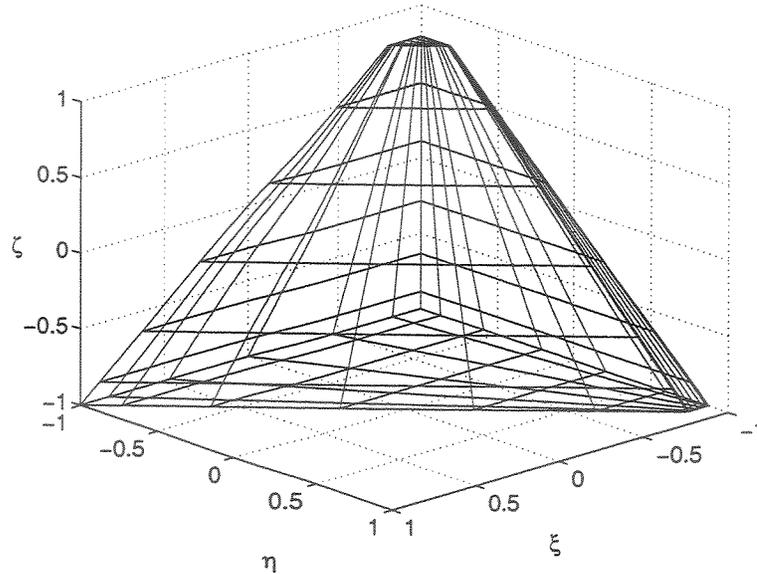


Figura 3.1: Pontos de quadratura sobre o tetraedro de referência T^3 com $Q_1 = Q_2 = Q_3 = 7$. Distribuição de pontos de *Gauss-Lobatto* na direção ξ e *Gauss-Radau* nas direções η e ζ (Karniadakis e Sherwin, 1999).

A obtenção recursiva dos pontos e pesos de integração da regra de *Gauss-Jacobi* para os três tipos de distribuição de pontos mencionados anteriormente (*Gauss-Legendre*, *Gauss-Radau* e *Gauss-Lobatto*) é descrita no Apêndice B. Esse Apêndice foi integralmente construído com base nas expressões e algoritmos fornecidos em (Karniadakis e Sherwin, 1999).

Tendo definido o procedimento de integração sobre o elemento de referência, resta ainda saber como esse tipo de integração pode ser estendido para uma região tetraédrica arbitrária dentro de uma malha de elementos finitos. Nota-se que qualquer elemento numa malha de discretização pode ser mapeado para o espaço padrão T^3 através de uma representação isoparamétrica de coordenadas. Se o elemento mapeado possui arestas retilíneas então o mapeamento possui um Jacobiano constante (pois a transformação é descrita por funções lineares) e a integração sobre esse elemento pode ser efetuada tal qual descrito anteriormente. No caso de elementos que exijam mapeamentos não-lineares (elementos curvos), uma interpolação polinomial das coordenadas curvilíneas permite que se obtenha uma aproximação consistente do Jacobiano (Sherwin e Karniadakis, 1996). Nesse caso, a fim de que se possa executar a integração sobre o domínio deformado, o Jacobiano deve ser avaliado

nos pontos de integração e então multiplicado pelo integrando da Equação (3.7) da seguinte forma:

$$\int_{\Omega_e} u(x, y, z) dx dy dz = \sum_{i=0}^{Q_1-1} \sum_{j=0}^{Q_2-1} \sum_{k=0}^{Q_3-1} J(a_i^{0,0}, b_j^{1,0}, c_k^{2,0}) u(a_i^{0,0}, b_j^{1,0}, c_k^{2,0}) w_i^{0,0} \hat{w}_j^{1,0} \hat{w}_k^{2,0}, \quad (3.9)$$

sendo $J(a_i^{0,0}, b_j^{1,0}, c_k^{2,0})$ a aproximação do Jacobiano da transformação $\Omega_e \rightarrow T^3$.

3.1.2 Diferenciação numérica sobre regiões tetraédricas

A discretização de problemas de elasticidade linear e não-linear necessariamente envolve o cálculo de operadores diferenciais. Esses operadores são aplicados sobre as funções de base que geram o espaço de aproximação e, em geral, são avaliados nos pontos de integração para cada elemento da malha. A obtenção das derivadas das funções de base através de procedimentos de manipulação simbólica, sobretudo no caso **3D**, é uma tarefa bastante custosa podendo inclusive tornar muito poluída a programação de códigos baseados no **MEF**. Dessa forma, a utilização de procedimentos de diferenciação numérica torna-se bastante atraente.

Considera-se nesta seção o tipo de diferenciação numérica denominado *diferenciação por colocação* ou *diferenciação no espaço físico* (Karniadakis e Sherwin, 1999). Esse tipo de diferenciação se baseia na representação de funções polinomiais arbitrárias por polinômios de *Lagrange* definidos sobre um conjunto de pontos tipicamente escolhidos como sendo os próprios pontos de integração. Essa representação pode ser entendida simplesmente como uma mudança de base num espaço de polinômios (i.e., uma mudança de uma base qualquer para a base de *Lagrange*). Desse modo, é possível escrever uma função polinomial de aproximação **1D** em coordenadas locais como

$$u(\xi) = \sum_{p=0}^{P_1} \hat{u}_p \phi_p(\xi) = \sum_{p=0}^{P_1} u_p h_p(\xi) \quad (3.10)$$

sendo $\phi_p(\xi)$ as funções de base de ordem menor ou igual a P_1 associadas ao espaço de aproximação, $h_p(\xi)$, os polinômios de *Lagrange* que passam pelo conjunto de $P_1 + 1$ pontos de integração e \hat{u}_p e u_p , os coeficientes da aproximação nos espaços original e transformado, respectivamente.

Devido às propriedades de colocação da aproximação de *Lagrange* (i.e., $h_i(\xi_j) = \delta_{ij}$), os coeficientes u_p são exatamente os valores da função de aproximação global nos pontos de integração, ou seja,

$$u_p = u(\xi_p). \quad (3.11)$$

Assim, é possível escrever a derivada da função de aproximação como

$$\frac{\partial u}{\partial \xi}(\xi) = \sum_{p=0}^{P_1} u_p \frac{\partial h_p}{\partial \xi}(\xi) = \sum_{p=0}^{P_1} u(\xi_p) \frac{\partial h_p}{\partial \xi}(\xi). \quad (3.12)$$

No caso particular do elemento de referência T^3 mapeado a partir de um espaço de coordenadas retangulares R^3 , pode-se representar qualquer expansão polinomial nesse espaço em termos dos polinômios de *Lagrange* usando-se as coordenadas tensorizadas a, b e c

$$u(a, b, c) = \sum_{p,q,r=0}^{P_1, P_2, P_3} \hat{u}_{pqr} \psi_{pqr}(a, b, c) = \sum_{p=0}^{P_1} \sum_{q=0}^{P_2} \sum_{r=0}^{P_3} u_{pqr} h_p(a) h_q(b) h_r(c), \quad (3.13)$$

sendo $\psi_{pqr}(a, b, c) = \phi_{pqr}(\xi, \eta, \zeta)$ as funções de base representadas no espaço tensorizado (a, b, c) e h_i (com $i = p, q$ ou r), os polinômios de *Lagrange* de ordem P_i (com $i = 1, 2$ ou 3) definidos com valor unitário em um único ponto de integração e zero em todos os outros pontos de integração.

Da mesma forma que na expressão (3.12), escrevem-se as derivadas parciais da função de aproximação com respeito às coordenadas tensorizadas (a, b, c) como

$$\frac{\partial u}{\partial a}(a, b, c) = \sum_{p=0}^{P_1} \sum_{q=0}^{P_2} \sum_{r=0}^{P_3} u_{pqr} \frac{\partial h_p(a)}{\partial a} h_q(b) h_r(c), \quad (3.14)$$

$$\frac{\partial u}{\partial b}(a, b, c) = \sum_{p=0}^{P_1} \sum_{q=0}^{P_2} \sum_{r=0}^{P_3} u_{pqr} h_p(a) \frac{\partial h_q(b)}{\partial b} h_r(c), \quad (3.15)$$

$$\frac{\partial u}{\partial c}(a, b, c) = \sum_{p=0}^{P_1} \sum_{q=0}^{P_2} \sum_{r=0}^{P_3} u_{pqr} h_p(a) h_q(b) \frac{\partial h_r(c)}{\partial c}. \quad (3.16)$$

Ao se avaliar essas expressões nos pontos de integração, usando-se a propriedade de colocação dos polinômios de *Lagrange*, chega-se a

$$\frac{\partial u}{\partial a}(a_i, b_j, c_k) = \sum_{p=0}^{P_1} u_{pjk} \frac{\partial h_p(a)}{\partial a} \Big|_{a_i}, \quad (3.17)$$

$$\frac{\partial u}{\partial b}(a_i, b_j, c_k) = \sum_{q=0}^{P_2} u_{iqk} \frac{\partial h_q(b)}{\partial b} \Big|_{b_j}, \quad (3.18)$$

$$\frac{\partial u}{\partial c}(a_i, b_j, c_k) = \sum_{r=0}^{P_3} u_{ijr} \frac{\partial h_r(c)}{\partial c} \Big|_{c_k}. \quad (3.19)$$

Os procedimentos de derivação por colocação que resultam no cálculo das derivadas dos polinômios de *Lagrange* nas expressões acima podem ser encontrados em (Karniadakis e Sherwin, 1999). Nessa referência, os algoritmos que avaliam essas derivadas nos pontos de integração levam

em conta os três tipos de distribuição de pontos considerados na seção anterior (i.e., *Gauss-Legendre*, *Gauss-Radau* e *Gauss-Lobatto*).

A partir das expressões (3.17), (3.18) e (3.19), pode-se utilizar a regra de diferenciação em cadeia para se obter as derivadas parciais locais com respeito às coordenadas cartesianas (ξ, η, ζ) do elemento de referência T^3

$$\nabla = \begin{pmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \\ \frac{\partial}{\partial \zeta} \end{pmatrix} = \begin{pmatrix} \frac{4}{(1-b)(1-c)} \frac{\partial}{\partial a} \\ \frac{2}{(1-b)(1-c)} \frac{\partial}{\partial a} + \frac{2}{(1-c)} \frac{\partial}{\partial b} \\ \frac{2}{(1-b)(1-c)} \frac{\partial}{\partial a} + \frac{2}{(1-c)} \frac{\partial}{\partial b} + \frac{\partial}{\partial c} \end{pmatrix}. \quad (3.20)$$

Observa-se em (3.20) que existe um problema iminente no cálculo do operador ∇ caso $b = 1$ ou $c = 1$ pois os fatores $1/(1-b)(1-c)$ e $1/(1-c)$ tornam-se infinitos. Uma maneira óbvia de se contornar esse inconveniente é evitar a avaliação das derivadas nesses pontos pelo uso da quadratura de *Gauss-Radau* nas direções b e c . Outras possíveis saídas para esse problema seriam a derivação explícita dos modos da função de aproximação ou a interpolação das derivadas nesses pontos a partir da avaliação das derivadas nos pontos correspondentes a distribuição de *Gauss-Radau*. Do ponto de vista da implementação, a primeira alternativa parece ser a mais simples e econômica.

3.1.3 Mapeamento de elementos

Como descrito no capítulo introdutório, este trabalho se restringe ao uso de mapeamentos do tipo afim. Isso significa que apenas malhas de elementos tetraédricos com arestas retilíneas e faces planas são admitidas. Nessa situação, um mapeamento simples pode ser construído a partir das funções de base associadas aos vértices do elemento de referência T^3 .

Para tratar esse caso de mapeamento, introduz-se a seguinte notação para uma transformação bijetiva do elemento de referência para um elemento qualquer da malha de discretização

$$x = \mathcal{X}^e(\xi, \eta, \zeta), \quad y = \mathcal{Y}^e(\xi, \eta, \zeta), \quad z = \mathcal{Z}^e(\xi, \eta, \zeta). \quad (3.21)$$

Assumindo que as coordenadas globais $\{(x^A, y^A, z^A), (x^B, y^B, z^B), (x^C, y^C, z^C), (x^D, y^D, z^D)\}$ dos vértices dos tetraedros da malha de discretização sejam conhecidas, constrói-se a seguinte trans-

formação, escrita em coordenadas tensorizadas:

$$\begin{aligned}
x &= \mathcal{X}^e(a, b, c) = x^A \left(\frac{1-a}{2} \right) \left(\frac{1-b}{2} \right) \left(\frac{1-c}{2} \right) \\
&+ x^B \left(\frac{1+a}{2} \right) \left(\frac{1-b}{2} \right) \left(\frac{1-c}{2} \right) \\
&+ x^C \left(\frac{1+b}{2} \right) \left(\frac{1-c}{2} \right) \\
&+ x^D \left(\frac{1+c}{2} \right),
\end{aligned} \tag{3.22}$$

sendo a, b e c dados pelas relações (3.1). Expressões análogas podem ser derivadas para as coordenadas globais y e z .

Esse mapeamento linear faz com que as derivadas das coordenadas globais em relação às coordenadas locais sejam constantes o que implica que o Jacobiano da transformação de um elemento arbitrário em (x, y, z) para o elemento de referência em (ξ, η, ζ) seja constante também. Como visto anteriormente (Equação (3.4)), o Jacobiano da transformação do elemento de referência (coordenadas ξ, η, ζ) para o espaço tensorizado (coordenadas a, b, c) não é constante mas pode ser acomodado nos pesos da regra de quadratura.

A construção de um mapeamento baseado nas próprias funções de base que interpolam o espaço de aproximação pode ser estendida para incluir regiões com curvatura. Nesse caso a geometria curva é representada por meio de uma expansão polinomial de mesma ordem que aquela que aproxima a solução do problema estudado (mapeamento isoparamétrico). Maiores detalhes sobre esse tipo de técnica podem ser encontrados em (Sherwin e Karniadakis, 1996; Karniadakis e Sherwin, 1999).

3.2 Operações Globais

3.2.1 Conectividade entre elementos tetraédricos

Para uma vasta classe de problemas elípticos, incluindo os de elasticidade linear e não-linear, a exigência de continuidade da solução aproximada entre os elementos da malha de discretização é fundamental para que se garanta a convergência do **MEF**. Nesse sentido, é necessário compreender como a orientação dos sistemas de coordenadas locais pode ser manipulada de modo a gerar uma

expansão polinomial global contínua.

Para se manter a continuidade da expansão polinomial global é preciso conectar os modos de formato similar. Como os modos associados ao interior dos elementos possuem suporte compacto sobre o próprio elemento (i.e., se anulam em todos os seus bordos), considera-se apenas a conectividade dos modos associados ao contorno dos tetraedros (funções associadas a arestas e faces). Observando a Figura 3.2, nota-se que o uso de coordenadas colapsadas faz com que as faces triangulares do elemento tetraédrico não apresentem uma simetria rotacional. Essa constatação torna-se clara ao se verificar a concentração de linhas de coordenadas em torno dos vértices do tetraedro que foram colapsados a partir do domínio hexaédrico ($R^3 \rightarrow T^3$). Esses vértices recebem as seguintes denominações: *vértice degenerado da base* ($\xi = -1, \eta = 1, \zeta = -1$) e *vértice degenerado do topo* ($\xi = -1, \eta = -1, \zeta = 1$).

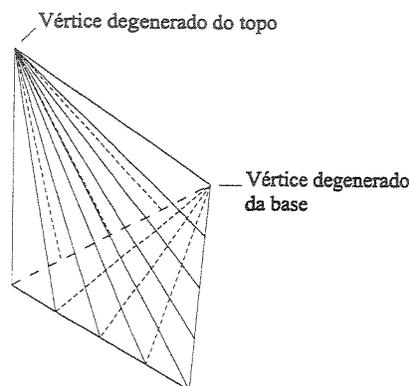
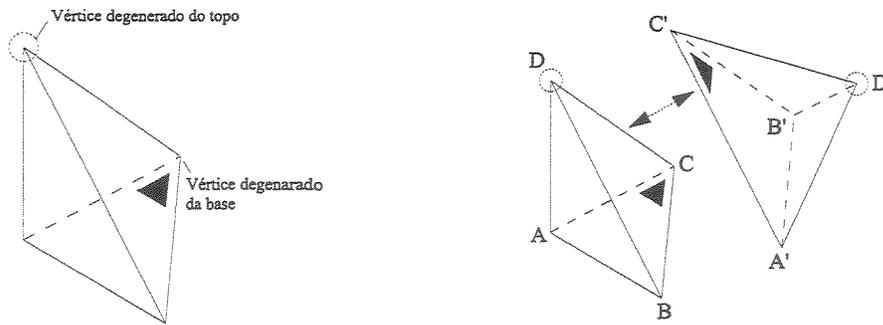


Figura 3.2: Linhas de coordenadas sobre as faces do elemento tetraédrico (Sherwin e Karniadakis, 1996).

Dessa forma, a restrição de conectividade entre os modos associados às faces dos elementos tetraédricos implica que as linhas de coordenadas de diferentes elementos devem estar orientadas numa mesma direção. Portanto, um vértice degenerado deve estar sempre conectado a outro vértice degenerado (Figura 3.3). Nesse sentido, é possível imaginar três tipos de conexão admissíveis: (i) um *vértice degenerado do topo* se cola a outro *vértice degenerado do topo*; (ii) um *vértice degenerado do topo* se cola a um *vértice degenerado da base* ($D = C'$, Figura 3.3(b)); (iii) um *vértice degenerado da base* se cola a outro *vértice degenerado da base*.

Como a orientação dos sistemas de coordenadas dos elementos está associada a incidência



(a) Representação diagramática dos *vértices degenerados da base e do topo*.

(b) Conexão admissível entre um *vértice degenerado da base* e um *vértice degenerado do topo*.

Figura 3.3: Conectividade entre faces de elementos tetraédricos definida segundo (Sherwin e Karniadakis, 1996).

da malha de discretização, é possível manipular essa informação (proveniente de um gerador de malhas) de modo a se obter uma orientação consistente para todos os sistemas locais. Isso, por sua vez, produz uma aproximação conforme sobre uma malha genérica de elementos tetraédricos.

Assumindo que todo vértice global seja identificado por um único número e que esses vértices estejam organizados quatro a quatro (cada elemento tetraédrico é composto de quatro vértices) numa matriz de incidência, pode-se estabelecer um algoritmo de orientação para os sistemas de coordenadas locais da seguinte forma (Warburton, 1998):

Algoritmo para a orientação dos sistemas de coordenadas locais

1. Coloque (em cada linha da matriz de incidência) o vértice com o menor número global na posição do *vértice degenerado do topo* (última posição);
2. Coloque o segundo vértice com o menor número global na posição do *vértice degenerado da base* (penúltima posição);
3. Oriente os dois vértices restantes (primeira e segunda posições) de forma consistente com a rotação do sistema local (normamente anti-horária).

Esse algoritmo possui um caráter local podendo, no entanto, ser implementado numa etapa de pré-processamento. Esse critério de orientação dos sistemas locais garante que qualquer eixo (local) apontando na direção de um vértice degenerado esteja posicionado de forma consistente dentro

da malha de elementos finitos. Apesar disso, ainda existe a possibilidade de algumas funções de interpolação (e.g., as que variam ao longo da base da face triangular compartilhada) necessitarem de uma inversão de sinal para que se assegure a continuidade global da aproximação. Isso ocorre porque as funções de base são definidas com respeito às coordenadas tensorizadas (a, b, c) e, em certas configurações dos elementos na malha, a direção dos sistemas de coordenadas locais fica invertida. Esse fato pode ser facilmente observado se for considerada a colagem entre as faces de número 1 de dois elementos distintos (a Figura 3.4 fornece a numeração das entidades topológicas sobre o elemento tetraédrico). Nesse caso, a coordenada a de cada elemento aponta em sentidos opostos pois a face compartilhada é percorrida no sentido horário para um elemento e anti-horário para o outro. Assim, é necessário trocar o sinal dos modos ímpares associados à aresta 1 de um dos elementos que compartilham essa face. Além disso, os modos associados à face de um dos elementos, que são ímpares na variável tensorizada a , também devem ter o seu sinal trocado.

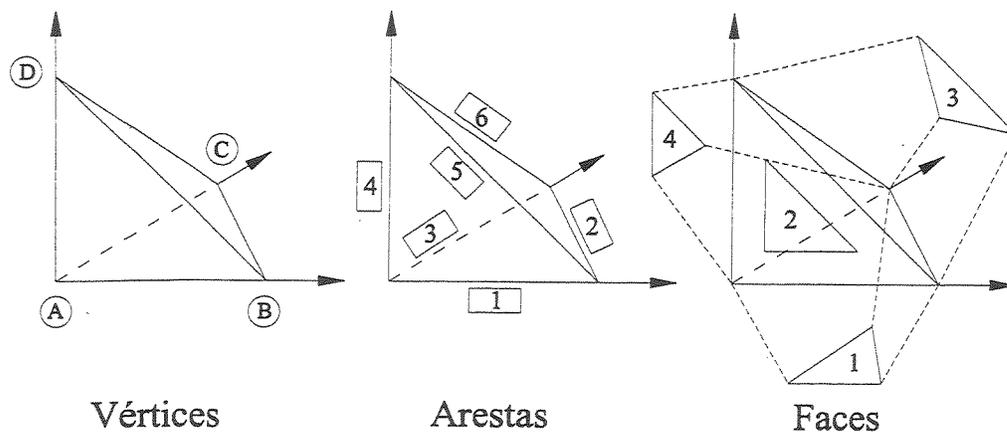
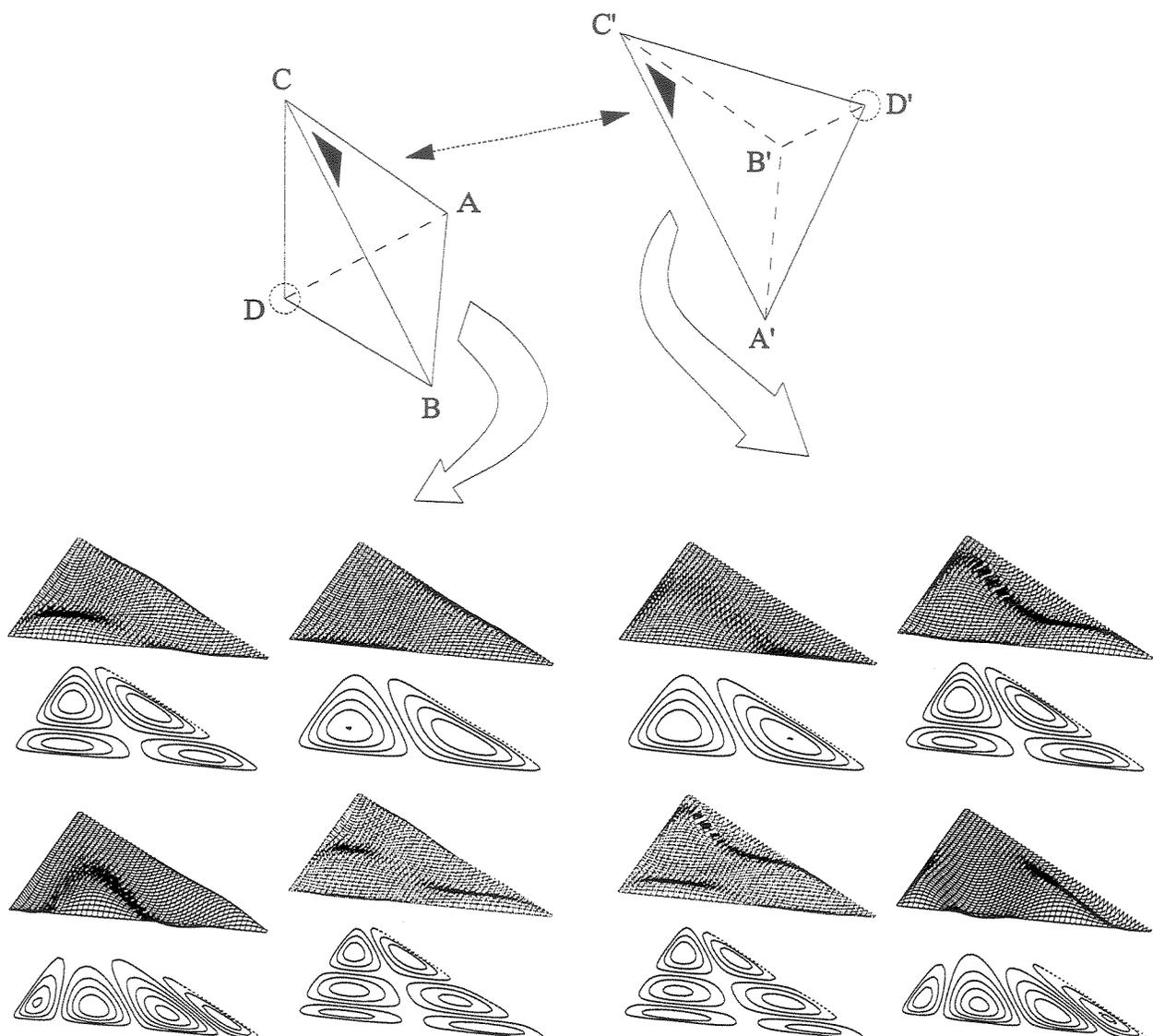


Figura 3.4: Identificação das entidades topológicas sobre o elemento tetraédrico (Sherwin e Karniadakis, 1996).

A Figura 3.5 mostra uma representação 2D do comportamento dos modos de ordem menor ou igual a 6 associados às faces de número 1 de dois elementos distintos quando não há inversão de sinal. Apenas os modos que são ímpares na variável tensorizada a estão ilustrados.

Observa-se que, embora as curvas de nível dessas funções sejam idênticas, elas próprias diferem exatamente por um sinal negativo.

O procedimento de inversão de sinal dos modos ímpares associados às arestas e às faces deve



(a) Modos de ordens $p = 4, 5$ e 6 associados a face 1 do primeiro elemento.

(b) Modos de ordens $p = 4, 5$ e 6 associados a face 1 do segundo elemento.

Figura 3.5: Representação 2D do comportamento das funções de base associadas as faces de número 1 compartilhadas por dois elementos tetraédricos distintos quando não há inversão de sinal dos modos ímpares.

ocorrer sempre que:

- uma face i (na numeração local) se conecta a outra face i ($i = 1, \dots, 4$);
- a face 1 se conecta à face 4;
- a face 2 se conecta à face 3;

3.2.2 Numeração de graus de liberdade

A propriedade hierárquica das funções de base permite que a expansão polinomial de aproximação do MEF não precise ser totalmente reconstruída ao se elevar a ordem da aproximação. Do ponto de vista computacional, essa propriedade pode ser vantajosa ao evitar cálculos de integração e diferenciação desnecessários. Além disso, uma ordenação hierárquica dos graus de liberdade globais é essencial para a aplicação de técnicas iterativas multi-níveis na solução dos sistemas de equações resultantes do processo de discretização.

Por outro lado, uma ordenação dos graus de liberdade que leve em conta a estrutura topológica da malha de discretização (vértices, arestas, faces e elementos) pode se tornar bastante atraente quando se considera a drástica redução no tempo de solução dos sistemas lineares ao se utilizar o procedimento de condensação estática dos modos internos. Essa vantagem decorre tanto da redução do tamanho dos sistemas globais quanto do número de condição associado às matrizes de rigidez desses sistemas. Como os graus de liberdade associados aos modos internos são puramente locais em relação aos elementos (i.e, não sofrem superposição na matriz de rigidez global), o procedimento de condensação pode ser conduzido elemento a elemento facilitando assim sua paralelização.

Para que se possa efetivamente aproveitar as vantagens de ambos os tipos de ordenação é necessário pensar num tipo de numeração de graus de liberdade híbrido, ou seja, que de algum modo seja propício à aplicação de técnicas de condensação e ao mesmo tempo preserve uma parte da estrutura hierárquica do espaço de aproximação. Nesse sentido, o presente trabalho considera uma numeração local dos graus de liberdade baseada na estrutura topológica do elemento e uma numeração global que tenha uma ordenação polinomial hierárquica a menos dos graus de liberdade associados aos modos internos.

Desse modo, a numeração dos graus de liberdade locais é ordenada de maneira que os vértices

apareçam primeiro seguidos das arestas, faces e por último os interiores. Em termos computacionais, isso pode ser facilmente conseguido através de um arranjo correto das funções de base dentro do código de programação. Globalmente, os graus de liberdade associados a vértices, arestas e faces são organizados pela ordem polinomial de aproximação, o que significa dizer que nem sempre graus de liberdade associados a arestas ocuparão posições contíguas, o mesmo ocorrendo para o caso das faces. Os graus de liberdade associados ao interior dos elementos serão sempre posicionados por fim (i.e., aparecem após a numeração dos demais graus de liberdade) ignorando portanto a sua disposição hierárquica.

Algoritmos que mapeiam a numeração local dos graus de liberdade para a numeração global serão discutidos posteriormente num capítulo dedicado aos aspectos de implementação. Neste ponto, é suficiente descrever algumas estruturas básicas necessárias à determinação e ao controle da numeração dos graus de liberdade.

A princípio, a numeração dos graus de liberdade numa estrutura baseada na versão- p do MEF começa com a numeração dos vértices (nós físicos) da malha fornecida por um gerador de malhas. A partir daí, é possível definir nós não-físicos associados às entidades topológicas da malha como arestas, faces e interiores de elementos. Cada uma dessas entidades recebe uma numeração global. A etapa seguinte à definição de nós não-físicos está relacionada a identificação de arestas e faces que sejam comuns a diferentes elementos. Esse procedimento é necessário já que os graus de liberdade locais que estão compartilhados possuem uma única numeração global. A seguir, é preciso definir a ordem polinomial sobre cada entidade topológica global uma vez que isso está diretamente relacionado ao tamanho do espaço de aproximação e portanto ao número de graus de liberdade envolvidos. Uma distribuição da ordem polinomial para cada elemento da malha deve ser fornecida juntamente com o arquivo de coordenadas e incidências obtido do gerador de malha. Após essa etapa, é preciso considerar a identificação das entidades topológicas da malha que estão sob condições de contorno de *Dirichlet* homogêneas. Esse passo permite eliminar graus de liberdade que seriam redundantes para a solução do sistema algébrico linear final. De posse de todas essas informações, é possível estabelecer um mapeamento (descrito por vetores indexados) que forneça a numeração global dos graus de liberdade a partir da numeração local restrita ao elemento de referência (computacionalmente chamado de elemento de trabalho).

3.3 Descrição de Fronteiras em Problemas de Valor de Contorno

3.3.1 Tipos de condições de contorno associados a problemas de elasticidade

Ao se considerar as equações da elasticidade linear e não-linear através da formulação de problemas de valor de contorno (PVC), é imprescindível representar matematicamente as condições de fronteira que definem univocamente esses problemas. Dentro dessa classe de equações, é comum o tratamento de condições de contorno do tipo *Dirichlet* e/ou *Neumann*. Essas condições de contorno podem ainda ser classificadas como homogêneas e não-homogêneas.

As condições de contorno do tipo *Neumann* estão relacionadas a aplicação de esforços sobre a fronteira do domínio estudado e as do tipo *Dirichlet* estão associadas a imposição de deslocamentos sobre essas fronteiras. As condições do tipo *Neumann* homogêneas não requerem nenhum tipo especial de atenção uma vez que não adicionam informações à descrição do PVC, tanto do ponto de vista da formulação quanto da implementação. As condições do tipo *Neumann* não-homogêneas envolvem a integração de termos descritos sobre a fronteira do domínio considerado e em geral necessitam do cálculo do *Jacobiano* de superfície associado a transformação entre as faces do elemento de referência e as faces dos elementos que tocam a fronteira.

No que concerne à formulação do PVC, a aplicação de condições do tipo *Dirichlet* está intimamente ligada à escolha do espaço de aproximação. Do ponto de vista computacional, a aplicação desse tipo de condição de contorno depende da manipulação dos graus de liberdade do problema (caso homogêneo) bem como do sistema de equações lineares resultante (caso não-homogêneo). No caso homogêneo, o tratamento de condições de contorno de *Dirichlet* se dá pela eliminação de graus de liberdade com valor nulo (a inclusão desses graus de liberdade no sistema de equações globais é redundante e aumenta o custo do processo de solução). Esse procedimento é normalmente incluído na etapa de numeração dos graus de liberdade do problema como descrito na seção anterior. No caso não-homogêneo, a aplicação desse tipo de condição de contorno é bem menos imediata necessitando de técnicas que descrevam o contorno geométrico do domínio além de procedimentos locais que permitam a interpolação dos coeficientes associados aos graus de liberdade não-físicos por meio dos valores impostos àqueles associados aos graus de liberdade físicos (esse procedimento pode ser entendido como uma projeção, no sentido de *Galerkin*) (Sherwin e Karniadakis, 1996; Karniadakis e Sherwin, 1999).

Este trabalho considera apenas a aplicação de condições de contorno dos tipos *Dirichlet* homogêneo e *Neumann* não-homogêneo. Os procedimentos ligados à descrição das condições de contorno do primeiro tipo foram mencionados na seção anterior e os do segundo tipo são discutidos a seguir.

3.3.2 Aplicação de condições de contorno do tipo Neumann não-homogêneas

Essencialmente, a formulação variacional de um PVC com condições de contorno de *Neumann* não-homogêneas associado às equações de elasticidade 3D envolve o cálculo de funcionais lineares do seguinte tipo:

$$\mathcal{F}(v) = \int \int \int_{\Omega} (b_x v_x + b_y v_y + b_z v_z) dx dy dz + \int \int_{\partial\Omega^N} (\bar{t}_x v_x + \bar{t}_y v_y + \bar{t}_z v_z) dS \quad (3.23)$$

sendo b_x, b_y, b_z as componentes do vetor \mathbf{b} que descreve as forças de corpo sobre um domínio Ω ; $\bar{t}_x, \bar{t}_y, \bar{t}_z$, as componentes do vetor $\bar{\mathbf{t}}$ que descreve os esforços aplicados ao contorno $\partial\Omega^N$ (denominado vetor de tração) e v_x, v_y, v_z , as funções testes pertencentes ao espaço de aproximação.

No caso mais geral, o vetor de carregamento $\bar{\mathbf{t}}$ é descrito em cada uma de suas componentes por uma função que depende das coordenadas espaciais (x, y, z) . Em problemas de caráter prático, no entanto, é bastante comum representar esses carregamentos ao longo do contorno através de funções constantes por partes. Esse tratamento simplifica acentuadamente a avaliação da integral de superfície presente no segundo membro de (3.23). Do ponto de vista local, efetua-se o cálculo dessa integral mapeando-se as faces dos elementos que estão em contato com a fronteira $\partial\Omega^N$ para as respectivas faces do elemento de referência T^3 (a fim de que se possa utilizar as regras de quadratura discutidas previamente).

Na discretização dos operadores de elasticidade pelo método de *Galerkin*, a integral de superfície em (3.23) descrita localmente sobre uma das faces do elemento de referência T^3 é calculada para cada função de base associada a essa face. Dessa forma, tomando-se como exemplo a obtenção do i -ésimo coeficiente do vetor de carga correspondente às forças de superfície (supondo que a face de interesse tenha sido mapeada para a face 1 de T^3), tem-se

$$\begin{aligned} & \int \int_{\partial\Omega^e \cap \partial\Omega^N} (\bar{t}_x|_{\partial\Omega^e} \phi_x^i(x, y, z) + \bar{t}_y|_{\partial\Omega^e} \phi_y^i(x, y, z) + \bar{t}_z|_{\partial\Omega^e} \phi_z^i(x, y, z)) dS \\ &= \int \int_{F_{ace1}} (\bar{t}_x|_{\partial\Omega^e} \hat{\phi}_x^i(\xi, \eta, -1) + \bar{t}_y|_{\partial\Omega^e} \hat{\phi}_y^i(\xi, \eta, -1) + \bar{t}_z|_{\partial\Omega^e} \hat{\phi}_z^i(\xi, \eta, -1)) \left\| \frac{\partial \mathbf{r}}{\partial \xi} \times \frac{\partial \mathbf{r}}{\partial \eta} \right\| d\xi d\eta \end{aligned}$$

(3.24)

sendo $\hat{\phi}_x^i(\xi, \eta, -1)$, $\hat{\phi}_y^i(\xi, \eta, -1)$, $\hat{\phi}_z^i(\xi, \eta, -1)$ as funções de base associadas à face 1 do tetraedro T^3 (com $i = 1, \dots, n_p^1$) e $\left\| \frac{\partial \mathbf{r}}{\partial \xi} \times \frac{\partial \mathbf{r}}{\partial \eta} \right\|$, o Jacobiano de superfície que transforma a face 1 de T^3 na face triangular do elemento Ω^e que toca o bordo $\partial\Omega^N$.

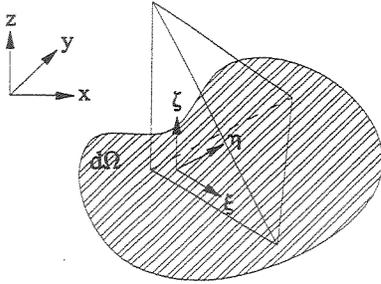
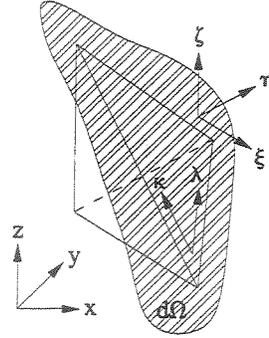
(a) Parametrização correspondente à face número 1 de T^3 (ξ, η)(b) Parametrização correspondente à face número 3 de T^3 (λ, κ)

Figura 3.6: Representação paramétrica de superfícies com condição de contorno de *Neumann* não-homogênea.

Nessa situação, a face triangular do elemento Ω^e submetida ao carregamento $\bar{\mathbf{t}}$ foi parametrizada pelas próprias coordenadas ξ e η do tetraedro de referência T^3 (Figura 3.6(a)). Essa parametrização é completamente descrita pelas seguintes transformações:

$$x = \mathcal{X}(\xi, \eta, -1), \quad y = \mathcal{Y}(\xi, \eta, -1), \quad z = \mathcal{Z}(\xi, \eta, -1). \quad (3.25)$$

Dessa forma, é possível definir um vetor posição $\mathbf{r}(x, y, z)$ que representa a face triangular de Ω^e associada à face 1 de T^3 a partir dos parâmetros ξ e η . Observa-se que uma variação desses parâmetros provoca uma variação do vetor posição \mathbf{r} dada por

$$d\mathbf{r} = \frac{\partial \mathbf{r}}{\partial \xi} d\xi + \frac{\partial \mathbf{r}}{\partial \eta} d\eta, \quad (3.26)$$

¹ n_p é o número total de funções de base de ordem menor ou igual a p associadas à face 1 de T^3 .

sendo $\frac{\partial \mathbf{r}}{\partial \xi}$, $\frac{\partial \mathbf{r}}{\partial \eta}$ os vetores tangentes à superfície descrita pelas linhas que conservam η e ξ constantes, respectivamente, i.e.

$$\frac{\partial \mathbf{r}}{\partial \xi} \Big|_{\eta=cte} = \begin{bmatrix} \frac{\partial x}{\partial \xi} \\ \frac{\partial y}{\partial \xi} \\ \frac{\partial z}{\partial \xi} \end{bmatrix}, \quad \frac{\partial \mathbf{r}}{\partial \eta} \Big|_{\xi=cte} = \begin{bmatrix} \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \eta} \\ \frac{\partial z}{\partial \eta} \end{bmatrix}. \quad (3.27)$$

Assim, o diferencial de área dS em (3.24), escrito em termos das coordenadas paramétricas ξ e η , se torna

$$dS = \left(\left\| \frac{\partial \mathbf{r}}{\partial \xi} \right\| \left\| \frac{\partial \mathbf{r}}{\partial \eta} \right\| \sin \theta \right) d\xi d\eta, \quad (3.28)$$

sendo θ o ângulo formado entre os vetores tangentes $\frac{\partial \mathbf{r}}{\partial \xi}$ e $\frac{\partial \mathbf{r}}{\partial \eta}$. Esse diferencial de área pode ainda ser escrito de forma equivalente como um produto vetorial, ou seja,

$$dS = \left\| \frac{\partial \mathbf{r}}{\partial \xi} \times \frac{\partial \mathbf{r}}{\partial \eta} \right\| d\xi d\eta. \quad (3.29)$$

Essa expressão representa exatamente o Jacobiano de superfície definido em (3.24).

A avaliação numérica da integral dupla em (3.24) é normalmente feita levando-se em conta o sistema de coordenadas cartesianas colapsado, assim

$$\begin{aligned} & \int \int_{Face1} (\bar{t}_x|_{\partial\Omega^e} \hat{\phi}_x^i(\xi, \eta, -1) + \bar{t}_y|_{\partial\Omega^e} \hat{\phi}_y^i(\xi, \eta, -1) + \bar{t}_z|_{\partial\Omega^e} \hat{\phi}_z^i(\xi, \eta, -1)) \left\| \frac{\partial \mathbf{r}}{\partial \xi} \times \frac{\partial \mathbf{r}}{\partial \eta} \right\| d\xi d\eta \\ &= \int_{-1}^1 \int_{-1}^1 (\bar{t}_x|_{\partial\Omega^e} \hat{\phi}_x^i(a, b, -1) + \bar{t}_y|_{\partial\Omega^e} \hat{\phi}_y^i(a, b, -1) + \bar{t}_z|_{\partial\Omega^e} \hat{\phi}_z^i(a, b, -1)) \left\| \frac{\partial \mathbf{r}}{\partial \xi} \times \frac{\partial \mathbf{r}}{\partial \eta} \right\| J da db, \end{aligned} \quad (3.30)$$

sendo $J = \left(\frac{1-b}{2}\right)$ o Jacobiano da transformação da face 1 do sistema colapsado T^3 para a face correspondente do sistema cartesiano hexaédrico R^3 . Observa-se que as derivadas $\frac{\partial \mathbf{r}}{\partial \xi}$, $\frac{\partial \mathbf{r}}{\partial \eta}$, na integral do segundo membro, devem ser avaliadas em $(\xi(a, b), \eta(a, b))$.

Com base no teorema da invariância de integrais de superfície com respeito a representações paramétricas suavemente equivalentes (**Teorema 12.2** de (Apostol, 1969)) e lembrando-se que as transformações dadas por (3.25) também podem ser escritas em função das coordenadas (a, b, c) (vide Equação (3.22)), é possível ainda reescrever a integral do segundo membro em (3.30) como

$$\begin{aligned}
& \int_{-1}^1 \int_{-1}^1 (\bar{t}_x|_{\partial\Omega^e} \hat{\phi}_x^i(a, b, -1) + \bar{t}_y|_{\partial\Omega^e} \hat{\phi}_y^i(a, b, -1) + \bar{t}_z|_{\partial\Omega^e} \hat{\phi}_z^i(a, b, -1)) \left\| \frac{\partial \mathbf{r}}{\partial \xi} \times \frac{\partial \mathbf{r}}{\partial \eta} \right\| J dadb \\
= & \int_{-1}^1 \int_{-1}^1 (\bar{t}_x|_{\partial\Omega^e} \hat{\phi}_x^i(a, b, -1) + \bar{t}_y|_{\partial\Omega^e} \hat{\phi}_y^i(a, b, -1) + \bar{t}_z|_{\partial\Omega^e} \hat{\phi}_z^i(a, b, -1)) \left\| \frac{\partial \mathbf{r}}{\partial a} \times \frac{\partial \mathbf{r}}{\partial b} \right\| dadb.
\end{aligned} \tag{3.31}$$

O procedimento de parametrização das faces de um elemento genérico Ω^e que correspondem às faces 2 e 4 de T^3 é análogo ao que foi feito para a face 1 de T^3 . No caso da face 3, no entanto, a parametrização óbvia, através da combinação de duas das variáveis locais ξ, η e ζ não é aplicável. Nessa situação, pode-se parametrizar a face 3 através das seguintes Equações (Figura 3.6(b)):

$$\begin{aligned}
\xi &= -\lambda - \kappa + 1, \\
\eta &= \lambda - 1, \\
\zeta &= \kappa - 1,
\end{aligned} \tag{3.32}$$

ou, em formato vetorial,

$$\{\xi, \eta, \zeta\} = \{-1, 1, 0\}\lambda + \{-1, 0, 1\}\kappa + \{1, -1, -1\}. \tag{3.33}$$

O vetor posição \mathbf{r} é dado em termos dos parâmetros λ, κ e a variação $d\mathbf{r}$ pode ser escrita como

$$d\mathbf{r} = \frac{\partial \mathbf{r}}{\partial \lambda} d\lambda + \frac{\partial \mathbf{r}}{\partial \kappa} d\kappa. \tag{3.34}$$

Com base na regra de derivação em cadeia, obtém-se os vetores tangentes $\frac{\partial \mathbf{r}}{\partial \lambda}$ e $\frac{\partial \mathbf{r}}{\partial \kappa}$

$$\frac{\partial \mathbf{r}}{\partial \lambda} \Big|_{\kappa=cte} = \begin{bmatrix} \frac{\partial x}{\partial \xi} \frac{\partial \xi}{\partial \lambda} + \frac{\partial x}{\partial \eta} \frac{\partial \eta}{\partial \lambda} + \frac{\partial x}{\partial \zeta} \frac{\partial \zeta}{\partial \lambda} \\ \frac{\partial y}{\partial \xi} \frac{\partial \xi}{\partial \lambda} + \frac{\partial y}{\partial \eta} \frac{\partial \eta}{\partial \lambda} + \frac{\partial y}{\partial \zeta} \frac{\partial \zeta}{\partial \lambda} \\ \frac{\partial z}{\partial \xi} \frac{\partial \xi}{\partial \lambda} + \frac{\partial z}{\partial \eta} \frac{\partial \eta}{\partial \lambda} + \frac{\partial z}{\partial \zeta} \frac{\partial \zeta}{\partial \lambda} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial \eta} - \frac{\partial x}{\partial \xi} \\ \frac{\partial y}{\partial \eta} - \frac{\partial y}{\partial \xi} \\ \frac{\partial z}{\partial \eta} - \frac{\partial z}{\partial \xi} \end{bmatrix} = \left(\frac{\partial \mathbf{r}}{\partial \eta} - \frac{\partial \mathbf{r}}{\partial \xi} \right), \tag{3.35}$$

$$\frac{\partial \mathbf{r}}{\partial \kappa} \Big|_{\lambda=cte} = \begin{bmatrix} \frac{\partial x}{\partial \xi} \frac{\partial \xi}{\partial \kappa} + \frac{\partial x}{\partial \eta} \frac{\partial \eta}{\partial \kappa} + \frac{\partial x}{\partial \zeta} \frac{\partial \zeta}{\partial \kappa} \\ \frac{\partial y}{\partial \xi} \frac{\partial \xi}{\partial \kappa} + \frac{\partial y}{\partial \eta} \frac{\partial \eta}{\partial \kappa} + \frac{\partial y}{\partial \zeta} \frac{\partial \zeta}{\partial \kappa} \\ \frac{\partial z}{\partial \xi} \frac{\partial \xi}{\partial \kappa} + \frac{\partial z}{\partial \eta} \frac{\partial \eta}{\partial \kappa} + \frac{\partial z}{\partial \zeta} \frac{\partial \zeta}{\partial \kappa} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial \zeta} - \frac{\partial x}{\partial \xi} \\ \frac{\partial y}{\partial \zeta} - \frac{\partial y}{\partial \xi} \\ \frac{\partial z}{\partial \zeta} - \frac{\partial z}{\partial \xi} \end{bmatrix} = \left(\frac{\partial \mathbf{r}}{\partial \zeta} - \frac{\partial \mathbf{r}}{\partial \xi} \right). \tag{3.36}$$

Assim, o cálculo da integral dada por (3.24), para o caso de uma face triangular descrita pelos parâmetros λ, κ associados à face 3 do elemento de referência T^3 , pode ser feito da seguinte maneira:

$$\begin{aligned} & \int \int_{Face3} (\bar{t}_x|_{\partial\Omega^e} \hat{\phi}_x^i(1, \eta, \zeta) + \bar{t}_y|_{\partial\Omega^e} \hat{\phi}_y^i(1, \eta, \zeta) + \bar{t}_z|_{\partial\Omega^e} \hat{\phi}_z^i(1, \eta, \zeta)) \left\| \frac{\partial \mathbf{r}}{\partial \lambda} \times \frac{\partial \mathbf{r}}{\partial \kappa} \right\| d\eta d\zeta \\ &= \int_{-1}^1 \int_{-1}^1 (\bar{t}_x|_{\partial\Omega^e} \hat{\phi}_x^i(1, b, c) + \bar{t}_y|_{\partial\Omega^e} \hat{\phi}_y^i(1, b, c) + \bar{t}_z|_{\partial\Omega^e} \hat{\phi}_z^i(1, b, c)) \left\| \frac{\partial \mathbf{r}}{\partial \lambda} \times \frac{\partial \mathbf{r}}{\partial \kappa} \right\| J dbdc, \end{aligned} \quad (3.37)$$

sendo $J = \left(\frac{1-c}{2}\right)$ o Jacobiano da transformação da face 3 do sistema colapsado T^3 para a face correspondente do sistema cartesiano hexaédrico R^3 . Ressalta-se que a integral do segundo membro de (3.37) deve ter as derivadas $\frac{\partial \mathbf{r}}{\partial \lambda}, \frac{\partial \mathbf{r}}{\partial \kappa}$ calculadas nos pontos $(\lambda(b, c), \kappa(b, c))$.

A avaliação numérica de integrais de superfície do tipo descrito em (3.24) caracteriza de forma completa a aplicação de condições de contorno do tipo *Neumann* não-homogêneas sobre os elementos de uma malha de discretização não-estruturada.

3.4 Aplicações em Elasticidade Linear

3.4.1 Formulação do problema de elasticidade infinitesimal

Nesta seção, pretende-se aplicar o desenvolvimento das seções precedentes em problemas de elasticidade linear sobre malhas **3D** não-estruturadas. Para tanto, descreve-se inicialmente o **PVC** tipicamente associado aos modelos de elasticidade linear bem como a formulação variacional empregada na discretização desse tipo de problema.

O **PVC** comumente associado a teoria de elasticidade linear é descrito pelas seguintes equações:

$$\begin{aligned} \operatorname{div} \boldsymbol{\sigma} + \mathbf{b} &= \mathbf{0} \quad \text{em } \Omega, \\ \mathbf{u} &= \bar{\mathbf{u}} \quad \text{em } \partial\Omega^D, \\ \boldsymbol{\sigma} \mathbf{n} &= \bar{\mathbf{t}} \quad \text{em } \partial\Omega^N. \end{aligned} \quad (3.38)$$

sendo $\boldsymbol{\sigma}$ o tensor de tensões de *Cauchy*, \mathbf{b} , o vetor de forças de corpo sobre o domínio Ω , $\bar{\mathbf{u}}$, o vetor de deslocamentos imposto na fronteira de *Dirichlet* $\partial\Omega^D$, $\bar{\mathbf{t}}$, o carregamento de superfície sobre a fronteira de *Neumann* $\partial\Omega^N$ e \mathbf{n} , o vetor unitário normal a superfície encerrada por $\partial\Omega^N$. Nos casos

considerados nesta seção, $\bar{\mathbf{u}}$ será sempre nulo.

A caracterização completa de um problema de elasticidade infinitesimal ainda depende da definição de uma relação constitutiva que associe tensões e deformações além de uma relação cinemática que descreva os deslocamentos compatíveis à formulação do problema:

$$\boldsymbol{\sigma}(\mathbf{u}) = \mathbf{C} : \boldsymbol{\epsilon}, \quad (3.39)$$

$$\boldsymbol{\epsilon}(\mathbf{u}) = \frac{1}{2} (\nabla \mathbf{u} + \nabla \mathbf{u}^T). \quad (3.40)$$

Nessas duas equações, \mathbf{C} é o tensor de constantes elásticas do material que compõe o domínio Ω , $\boldsymbol{\epsilon}$ é o tensor de deformações infinitesimais e \mathbf{u} é o vetor que fornece o deslocamento dos pontos materiais sobre o domínio Ω .

O vetor \mathbf{u} pode ser escrito como

$$\mathbf{u} = \mathbf{x} - \mathbf{X}, \quad (3.41)$$

sendo \mathbf{x} o ponto que descreve as coordenadas espaciais do domínio deformado (i.e., *ponto espacial*) e \mathbf{X} , o ponto que descreve as coordenadas materiais da configuração de referência do domínio Ω (*ponto material*).

A formulação variacional (ou fraca) equivalente ao PVC descrito em (3.38) pode ser escrita como:

Encontre o campo de deslocamentos $\mathbf{u} \in \mathbf{V} = \{\mathbf{v}(\mathbf{x}) \in [H^1(\Omega)]^3 : \mathbf{v} = \mathbf{0} \text{ em } \partial\Omega^D\}$ tal que

$$\int_{\Omega} \boldsymbol{\sigma}(\mathbf{u}) : \boldsymbol{\epsilon}(\mathbf{v}) dV = \int_{\Omega} \mathbf{b} \cdot \mathbf{v} dV + \int_{\partial\Omega^N} \bar{\mathbf{t}} \cdot \mathbf{v} dS, \quad \forall \mathbf{v} \in \mathbf{V}, \quad (3.42)$$

Nesse caso, $H^1(\Omega)$ é o espaço de *Sobolev* usualmente empregado na formulação variacional de operadores de segunda ordem. Observa-se que, para problemas de elasticidade infinitesimal, a formulação variacional apresentada acima não faz distinção entre as configurações original e deformada do domínio de integração (i.e., Ω é o domínio original não-deformado do problema e $\boldsymbol{\sigma}(\mathbf{u})$ é medido na configuração deformada).

Do ponto de vista da discretização pelo **MEF**, é comum definir o problema variacional (3.42) de forma compacta como

$$\mathcal{B}(u, v) = \mathcal{F}(v), \quad \forall \{v\} \in \mathbf{V}, \quad (3.43)$$

sendo $\mathcal{B}(u, v)$ uma forma bilinear e $\mathcal{F}(v)$ um funcional linear dados por

$$\mathcal{B}(u, v) = \int \int \int_{\Omega} ([D]\{v\})^T [C]([D]\{u\}) dx dy dz, \quad (3.44)$$

$$\mathcal{F}(v) = \int \int \int_{\Omega} (b_x v_x + b_y v_y + b_z v_z) dx dy dz + \int \int_{\partial\Omega^N} (\bar{t}_x v_x + \bar{t}_y v_y + \bar{t}_z v_z) dS. \quad (3.45)$$

Na expressão da forma bilinear, o integrando é obtido a partir da operação tensorial $\sigma(\mathbf{u}) : \epsilon(\mathbf{v})$ e da combinação de (3.39) e (3.40). Nesse caso ainda, $[C]$ é a matriz de constantes para materiais elásticos isotrópicos (vide (Szabó e Babuška, 1991)) correspondente ao tensor C em (3.39) e $[D]$ é o operador diferencial definido matricialmente por

$$[D] = \begin{bmatrix} \frac{\partial}{\partial x} & 0 & 0 \\ 0 & \frac{\partial}{\partial y} & 0 \\ 0 & 0 & \frac{\partial}{\partial z} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial z} & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} & 0 & \frac{\partial}{\partial x} \end{bmatrix} \quad (3.46)$$

Partindo-se do subespaço de dimensão finita $\mathbf{S}^p \subset \mathbf{V}^2$ associado a versão- p do **MEF** (Szabó e Babuška, 1991) e de uma aproximação para a solução $\{u\}$ do problema (3.43) dada por

$$\{u\} = \sum_i^{n_p} [\Phi_i] \{\hat{u}_i\}, \quad (3.47)$$

sendo

$$[\Phi_i] = \begin{bmatrix} \phi_x^i & 0 & 0 \\ 0 & \phi_y^i & 0 \\ 0 & 0 & \phi_z^i \end{bmatrix} \quad (3.48)$$

a matriz de funções de base globais obtida a partir da base local para tetraedros proposta por (Karniadakis e Sherwin, 1999), $\{\hat{u}_i\} = \{\hat{u}_x^i, \hat{u}_y^i, \hat{u}_z^i\}$, o vetor de graus de liberdade do problema e n_p , a dimensão do espaço \mathbf{S}^p , chega-se ao seguinte sistema de equações algébricas:

$$[\mathbf{K}] \{u\} = \{\mathbf{f}\}. \quad (3.49)$$

² \mathbf{S}^p é escrito de forma matematicamente rigorosa como $\mathbf{S}^p(\Omega, \Delta, \mathbf{Q})$, sendo os parâmetros $\Omega, \Delta, \mathbf{Q}$ associados ao domínio do problema, à malha de discretização e ao mapeamento entre o elemento de referência e os elementos da malha, respectivamente. O índice p em \mathbf{S} indica a ordem polinomial dos elementos que constituem a malha do problema.

Nesse sistema, $[K]$ é a matriz de rigidez e $\{f\}$, o vetor de carga globais associados ao problema discretizado.

3.4.2 Resultados numéricos para elasticidade linear

Uma vez descrito o tipo de problema que se pretende analisar, parte-se inicialmente para a validação da formulação desenvolvida anteriormente através da solução de um problema simples cuja solução analítica é conhecida. Nesse caso, faz-se uma breve avaliação do comportamento do erro cometido pela aproximação, na norma energia, ao se enriquecer o espaço polinomial de aproximação. A seguir, são considerados dois exemplos mais elaborados que não dispõem de uma solução analítica conhecida. Em ambas as situações, estima-se o erro cometido pela aproximação, na norma energia, com base no estimador de erros *a posteriori* sugerido em (Szabó e Babuška, 1991). Com o objetivo de se avaliar a qualidade dos resultados obtidos, compara-se uma solução pontual, em cada um dos casos, com a formulação adotada pelo programa computacional de uso comercial ANSYS 6.0.

Viga submetida ao seu peso próprio

O problema modelo adotado para a validação da formulação escolhida consiste da deformação elástica de uma viga prismática de seção transversal constante submetida ao seu peso próprio e engastada em uma de suas extremidades. A Figura 3.7 (obtida a partir do pré-processador do programa ANSYS) mostra a disposição dos elementos finitos sobre a viga, um sistema de coordenadas global posicionado em um de seus vértices superiores, além de uma representação esquemática das condições de contorno na sua face superior. As dimensões da largura e da espessura dessa viga são fixadas, respectivamente, em $l = 1,0\text{ m}$ e $h = 1,0\text{ m}$ enquanto que seu comprimento é tomado em três valores distintos: $c = 3,0\text{ m}$, $9,0\text{ m}$ e $18,0\text{ m}$. As constantes físicas do material que constitui a viga são descritas por: densidade $\rho = 7885\text{ kg/m}^3$, módulo de elasticidade longitudinal $E = 210 \times 10^9\text{ N/m}^2$ e coeficiente de Poisson $\nu = 0,3$. Esses dados correspondem às propriedades físicas do aço. Adota-se ainda o valor de $g = 9,81\text{ m/s}^2$ para a aceleração da gravidade. A malha do problema é composta por 18 elementos tetraédricos qualquer que seja seu comprimento.

A título de simplificação, a condição de contorno na extremidade superior da viga é imposta de modo a somente evitar os deslocamentos de corpo rígido. Desse modo, a extremidade superior

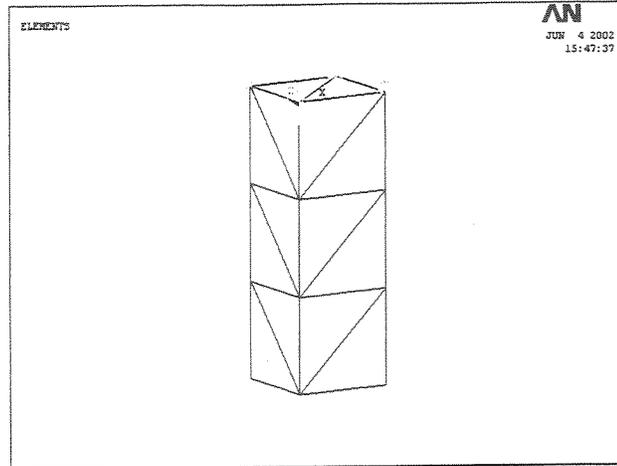


Figura 3.7: Malha de elementos tetraédricos sobre a viga submetida ao seu peso próprio e engastada em sua extremidade superior.

tem um de seus vértices com todas as coordenadas globais x, y, z restritas (i.e., um dos vértices está engastado), outro vértice tem as coordenadas y e z restritas e os dois restantes estão restritos apenas na coordenada y (i.e., 2 dos vértices superiores estão simplesmente apoiados). A solução analítica desse problema $\mathbf{u}_{ex} = (u_x, u_y, u_z)$, obtida com base em (Timoshenko e Goodier, 1970), é dada por

$$\begin{aligned}
 u_x &= \frac{\nu \rho g}{2E} (2xy - 2cx - y), \\
 u_y &= \frac{\rho g}{2E} \{2cy - y^2 - \nu(x^2 + z^2 - x - z)\}, \\
 u_z &= \frac{\nu \rho g}{2E} (2yz - 2cz - y),
 \end{aligned} \tag{3.50}$$

sendo c o comprimento da viga.

Nesse caso, assume-se que as tensões e as deformações específicas ao longo da viga possam ser calculadas através das seguintes expressões:

$$\begin{aligned}
 \sigma_{xx} &= \sigma_{zz} = \tau_{xy} = \tau_{xz} = \tau_{yz} = 0, \\
 \sigma_{yy} &= \rho g(c - y), \\
 \gamma_{xy} &= \gamma_{xz} = \gamma_{yz} = 0, \\
 \varepsilon_{xx} &= \varepsilon_{zz} = -\frac{\nu \rho g}{E} (c - y), \\
 \varepsilon_{yy} &= \frac{\rho g}{E} (c - y).
 \end{aligned} \tag{3.51}$$

A partir dessas relações, é possível calcular a norma energia da solução exata do problema representado na Figura 3.7. Observando que σ_{yy} é a única componente de tensão que contribui para a energia interna armazenada, pode-se escrever

$$\|\mathbf{u}_{ex}\|_{E(\Omega)}^2 = \frac{1}{2}\mathcal{B}(u_{ex}, u_{ex}) = \frac{1}{2} \int_{\Omega} \sigma_{yy} \varepsilon_{yy} dV. \quad (3.52)$$

Substituindo-se os termos σ_{yy} e ε_{yy} das expressões (3.51) na integral de (3.52), chega-se a

$$\int_{\Omega} \sigma_{yy} \varepsilon_{yy} dV = \frac{(\rho g)^2}{E} \int_0^h \int_0^l \int_0^c (c-y)^2 dx dy dz. \quad (3.53)$$

Aplicando-se os valores adequados às constantes ρ, g, E e aos limites de integração c, l, h nessa última expressão, obtêm-se os seguintes valores para a energia da solução exata (usando-se uma precisão de 8 dígitos) quando se consideram os três comprimentos da viga:

$$\begin{aligned} \|\mathbf{u}_{ex}\|_{E(\Omega)}^2 &\approx 0,12821376 \quad (c = 3,0 \text{ m}), \\ \|\mathbf{u}_{ex}\|_{E(\Omega)}^2 &\approx 3,46177146 \quad (c = 9,0 \text{ m}), \\ \|\mathbf{u}_{ex}\|_{E(\Omega)}^2 &\approx 27,69417169 \quad (c = 18,0 \text{ m}). \end{aligned} \quad (3.54)$$

Uma vez tendo sido calculada a energia da solução exata³, é possível se chegar ao erro cometido pela aproximação dessa solução obtida através da versão- p do **MEF**. Notando que a discretização em elementos finitos associada ao método de *Galerkin* implica na ortogonalidade do erro em relação à solução aproximada, pode-se escrever

$$\|\mathbf{u}_{ex}\|_{E(\Omega)}^2 = \|\mathbf{u}\|_{E(\Omega)}^2 + \|\mathbf{e}\|_{E(\Omega)}^2. \quad (3.55)$$

Assim, é suficiente calcular o valor da norma da solução aproximada \mathbf{u} para se obter o erro. Tendo sido montado o sistema de equações lineares para o problema considerado (Equação (3.49)), o cálculo da norma energia de \mathbf{u} torna-se imediato pois

$$\|\mathbf{u}\|_{E(\Omega)} = \sqrt{\frac{1}{2}\mathcal{B}(u, u)}, \quad (3.56)$$

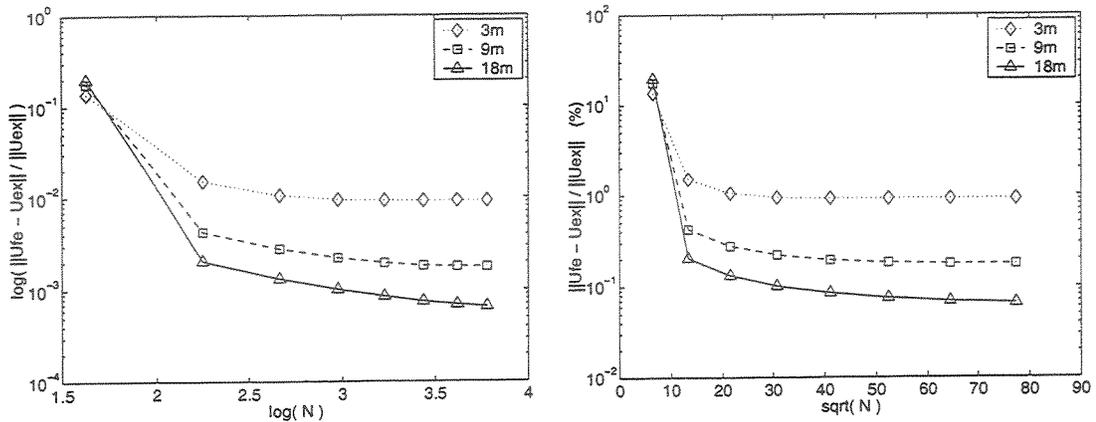
$$\mathcal{B}(u, u) = \{\hat{u}\}^T \left[\int \int \int_{\Omega} ([B]^T [C] [B]) dx dy dz \right] \{\hat{u}\} = \{\hat{u}\}^T [\mathbf{K}] \{\hat{u}\} = \{\hat{u}\}^T \{\mathbf{f}\}. \quad (3.57)$$

Na expressão (3.57), a matriz $[B]$ representa a aplicação do operador $[D]$ (Equação (3.46)) sobre todas as funções de base globais que interpolam a solução do problema e $\{\hat{u}\}$, a solução do

³A energia da solução é definida como o quadrado da norma energia de u , i.e.: $\Pi(u) = \|\mathbf{u}\|_{E(\Omega)}^2 = \frac{1}{2}\mathcal{B}(u, u)$.

sistema linear obtida a partir da Equação (3.49).

O erro relativo, na norma energia, da solução do problema da viga submetida ao peso próprio (Figura 3.7) em função do número de graus de liberdade associados aos refinamentos p de ordem 1 a 8 é mostrado nos gráficos 3.8(a) e 3.8(b). Adotando o padrão de análise apresentado em (Szabó e Babuška, 1991), o primeiro gráfico traz o comportamento do erro medido em escala log-log e o segundo, traz o erro percentual em função da raiz quadrada do número de graus de liberdade.



(a) Erro relativo vs. graus de liberdade.

(b) Erro percentual vs. $\sqrt{\text{graus de liberdade}}$.

Figura 3.8: Decaimento do erro da solução do problema da viga submetida ao peso próprio em função do número de graus de liberdade associados aos refinamentos p de ordem 1 a 8. Considere-se a viga com comprimentos $c = 3, 9$ e 18 m.

Observa-se, a partir desses gráficos, que a norma energia da solução obtida através do MEF não coincide exatamente com aquela fornecida pela solução analítica (expressões (3.54)), como era de se esperar para as aproximações de ordem maior ou igual a 2. Esse resultado, aparentemente errôneo, se deve basicamente à influência das condições de contorno pontuais (e portanto singulares) aplicadas nos vértices superiores da viga. A aplicação de condições de contorno feita dessa forma, como é comum em simulações numéricas de tensão-deformação, provoca perturbações locais na solução aproximada do problema (Princípio de *Saint-Venant* (Timoshenko e Goodier, 1970)). É possível verificar essa afirmativa notando-se que a norma energia da solução aproximada é reduzida apenas com o aumento do comprimento da viga sem que se altere o número de elementos da malha (vide gráficos 3.8(a) e 3.8(b)).

A Figura 3.9 confirma a hipótese apresentada acima ao mostrar o erro cometido pela solução

aproximada do deslocamento vertical de um nó (coordenadas $(x, y, z) = (1, c, 1)$) na extremidade livre da viga em função da ordem polinomial p . Nesse gráfico, observa-se claramente o caráter quadrático da solução do problema.

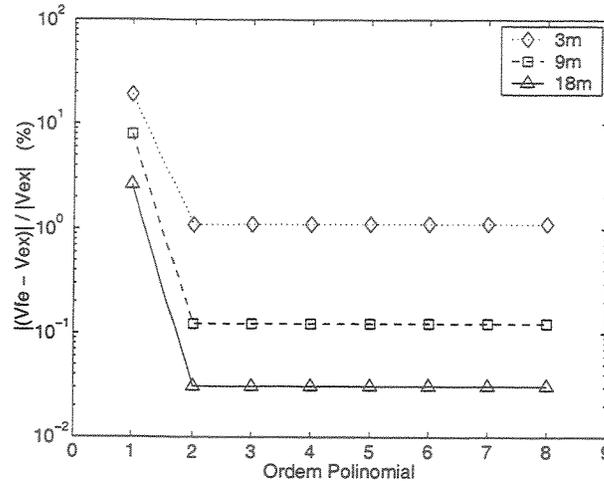
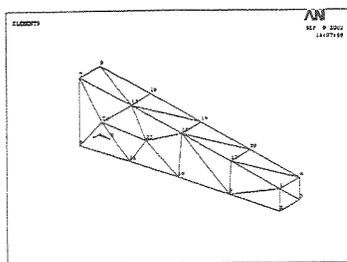


Figura 3.9: Erro percentual cometido na aproximação do deslocamento vertical u_y do ponto $(x, y, z) = (1, c, 1)$ pertencente à extremidade livre da viga submetida ao peso próprio em função da ordem polinomial p .

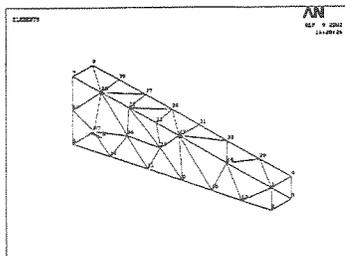
Viga em balanço submetida ao peso próprio e a uma carga de superfície

Este exemplo considera uma viga em balanço (i.e., engastada em uma das extremidades) submetida a um carregamento produzido pelo seu peso próprio e por uma carga de superfície de valor constante. Utilizando a formulação p do MEF desenvolvida anteriormente, obtém-se a solução desse problema para três malhas distintas com 34, 65 e 102 elementos tetraédricos. A ordem polinomial dos espaços de aproximação varia entre $p = 1$ e $p = 8$. As Figuras 3.10(a), 3.10(b) e 3.10(c) ilustram a disposição dos elementos tetraédricos e dos nós físicos para as três malhas consideradas. As Figuras 3.11(a), 3.11(b) e 3.11(c) trazem uma representação esquemática das condições de contorno associadas a essas três malhas. Nessas três últimas figuras, as linhas duplicadas sobre os elementos da face superior da viga representam uma pressão constante igual a 10^4 N/m^2 na direção contrária ao eixo y . As dimensões do comprimento, das larguras da extremidade engastada e da extremidade livre e da espessura da viga são dadas respectivamente por: $c = 2,0 \text{ m}$, $l_1 = 0,6 \text{ m}$, $l_2 = 0,2 \text{ m}$, $h = 0,2 \text{ m}$. As constantes físicas desse problema são idênticas às do problema anterior.

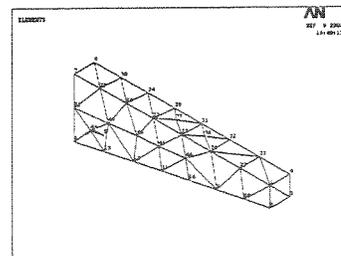
A Figura 3.12 mostra, para o caso da malha de 34 elementos, a deformação sofrida pela viga



(a) 34 elementos.

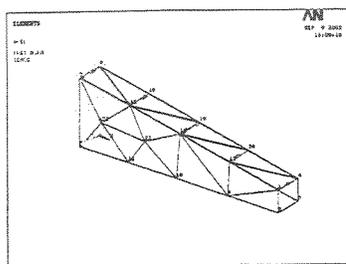


(b) 65 elementos.

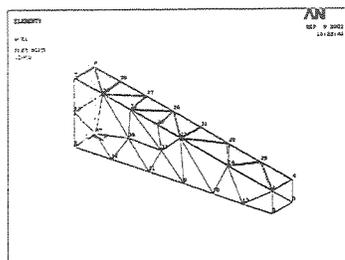


(c) 102 elementos.

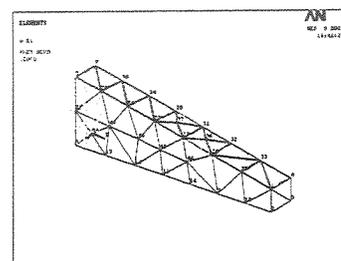
Figura 3.10: Malhas empregadas na análise da viga em balanço submetida ao peso próprio e a uma carga de superfície de valor constante.



(a) Malha de 34 elementos.



(b) Malha de 65 elementos.



(c) Malha de 102 elementos.

Figura 3.11: Condições de contorno aplicadas sobre a viga em balanço.

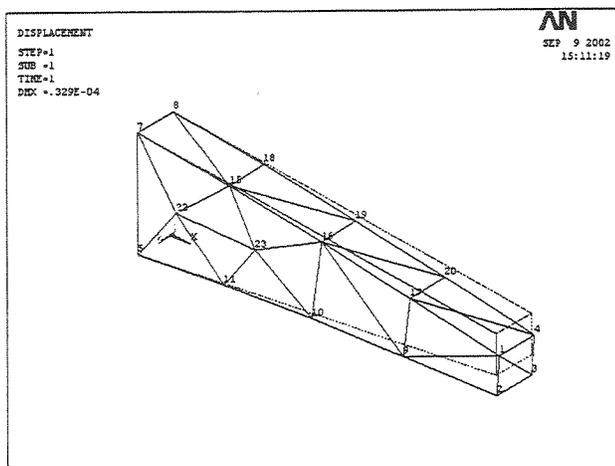


Figura 3.12: Configurações inicial (linha tracejada) e final (linha cheia) da viga em balanço (malha de 34 elementos).

após a obtenção da solução do problema.

Como não se dispõe da solução analítica desse problema, é necessário estimar o erro cometido pela solução aproximada para se avaliar a convergência da formulação p empregada. Dessa forma, adota-se a estimativa *a posteriori* para o erro na norma energia sugerida em (Szabó e Babuška, 1991). Basicamente, essa estimativa baseia-se na obtenção da energia da solução aproximada em três diferentes níveis de refinamento: $p, p-1$ e $p-2$. A partir desses valores e do número de graus de liberdade em cada um desses níveis, é possível estimar o valor da energia da solução exata.

Formalmente, com base na Equação (3.55), na definição da energia da solução e na estimativa de erro *a priori* sugerida por (Babuška et al., 1981), escreve-se

$$\|e\|_{E(\Omega)}^2 = \|\mathbf{u}_{ex} - \mathbf{u}\|_{E(\Omega)}^2 = \Pi(\mathbf{u}_{ex}) - \Pi(\mathbf{u}) \approx \frac{k^2}{N^{2\beta}}, \quad (3.58)$$

sendo k e β constantes positivas (β está associada à regularidade da solução exata do problema) e N , o número de graus de liberdade envolvidos na aproximação. Observa-se, a partir de (3.58), que existem três variáveis desconhecidas na última relação: $\Pi(\mathbf{u}_{ex}), k$ e β . Calculando-se as soluções aproximadas para três diferentes valores de p , obtém-se três equações para se computar as variáveis desconhecidas. Assim, pode-se escrever

$$\frac{\log \frac{\Pi - \Pi_p}{\Pi - \Pi_{p-1}}}{\log \frac{\Pi - \Pi_{p-1}}{\Pi - \Pi_{p-2}}} \approx \frac{\log \frac{N_{p-1}}{N_p}}{\log \frac{N_{p-2}}{N_{p-1}}}, \quad (3.59)$$

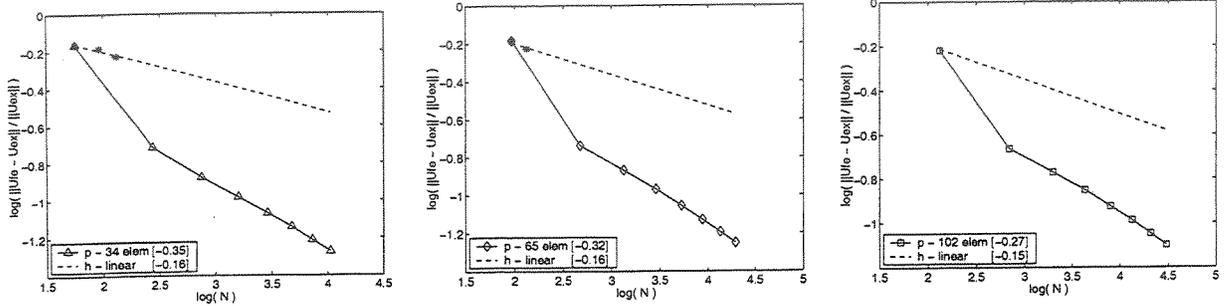
sendo Π a energia da solução exata; $\Pi_p, \Pi_{p-1}, \Pi_{p-2}$ e N_p, N_{p-1}, N_{p-2} , a energia das soluções e o número de graus de liberdade, respectivamente, referentes aos espaços de aproximação de ordem $p, p-1$ e $p-2$. Essa última relação pode ser reescrita de forma mais direta se o seu lado direito for denotado por uma única variável (e.g. Q), ou seja,

$$\frac{\Pi - \Pi_p}{\Pi - \Pi_{p-1}} \approx \left(\frac{\Pi - \Pi_{p-1}}{\Pi - \Pi_{p-2}} \right)^Q. \quad (3.60)$$

A estimativa *a posteriori* descrita acima assume a convergência monotônica de Π_p o que implica que os espaços de aproximação de diferentes ordens polinomiais devam ser aninhados, i.e., $S^{p-2} \subset S^{p-1} \subset S^p$. Isso é sempre verdadeiro para as formulações do tipo p uma vez que as bases desses espaços são hierárquicas. O mesmo não ocorre para as formulações do tipo h .

Os gráficos 3.13(a), 3.13(b) e 3.13(c) comparam os padrões de convergência das soluções

aproximadas obtidas a partir das formulações h e p do MEF para as três malhas consideradas. A convergência das soluções é medida com base na evolução do erro relativo na norma energia em função do número de graus de liberdade envolvidos (usa-se uma escala log-log). Os refinamentos do tipo p são representados pelas linhas cheias com a ordem polinomial de aproximação variando uniformemente entre 1 e 8. Os refinamentos do tipo h são representados pelas linhas tracejadas. Essas linhas foram extrapoladas a partir das soluções obtidas para as malhas de 34, 65 e 102 elementos. Nesse caso, foram empregados os mesmos elementos tetraédricos lineares de quatro nós que iniciam o processo de refinamento p . Observa-se, por fim, que a legenda de cada gráfico traz o coeficiente angular associado ao decaimento do erro em ambos os casos: h e p . O cálculo do coeficiente angular, no caso particular do refinamento p , despreza o decaimento pré-assintótico localizado entre as aproximações de ordem $p = 1$ e $p = 2$.



(a) Malha de 34 elementos.

(b) Malha de 65 elementos.

(c) Malha de 102 elementos.

Figura 3.13: Decaimento do erro relativo na norma energia em função do número de graus de liberdade envolvidos no problema da viga em balanço.

Observa-se a partir desses gráficos que a taxa de decaimento do erro relativo no caso do refinamento p é aproximadamente o dobro daquela apresentada pelo refinamento h . Esse resultado demonstra pleno acordo com as estimativas teóricas para a taxa de convergência exponencial associada aos refinamentos do tipo p (Szabó e Babuška, 1991).

As estimativas obtidas (com precisão de 8 dígitos) para a energia da solução exata em cada uma das malhas consideradas foram

$$\begin{aligned}
 \Pi(u) &\approx 0,08464549 \quad (34 \text{ elem.}), \\
 \Pi(u) &\approx 0,08483894 \quad (65 \text{ elem.}), \\
 \Pi(u) &\approx 0,08634225 \quad (102 \text{ elem.}).
 \end{aligned}
 \tag{3.61}$$

O gráfico 3.14 mostra a evolução do erro percentual das soluções aproximadas obtidas a partir da formulação p desenvolvida anteriormente em função da raiz quadrada do número de graus de liberdade associados aos três tipos de malhas físicas consideradas. Da mesma forma que na análise precedente, o refinamento polinomial varia entre $p = 1$ e $p = 8$.

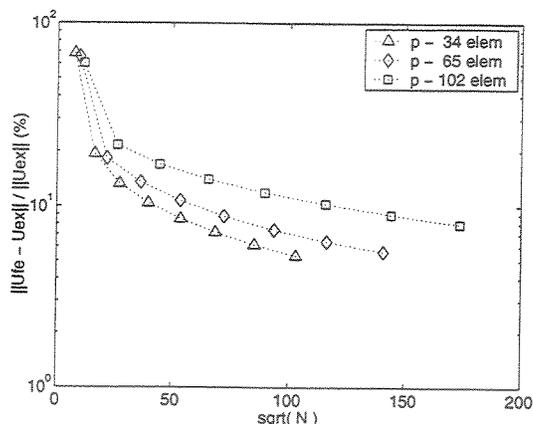


Figura 3.14: Decaimento do erro percentual em função do número de graus de liberdade associados às três malhas consideradas no problema da viga em balanço.

Um resultado notável observado nesse gráfico é a diminuição da precisão de resposta com o aumento do número de elementos na malha de discretização. Esse fato poderia parecer discrepante à luz da teoria de aproximação para métodos do tipo hp , no entanto, deve-se salientar que a estimativa de erro na norma energia utilizada nesses casos é construída exclusivamente para refinamentos do tipo p . Isso suscita a hipótese de que o estimador *a posteriori* sugerido em (Szabó e Babuška, 1991) torna-se cada vez mais exigente a medida que se combina o refinamento do tipo h ao do tipo p . Outra observação relevante contida nesse gráfico diz respeito a baixa precisão alcançada pelas soluções lineares de partida. Essa limitação, largamente conhecida pelos usuários de programas comerciais de simulação estrutural, está associada a rigidez excessiva introduzida por elementos tetraédricos lineares em modelos descritos através do operador de elasticidade.

A Tabela 3.1 compara soluções pontuais do problema da viga em balanço obtidas a partir de duas formulações do tipo p distintas: aquela desenvolvida neste capítulo e outra obtida a partir do programa comercial ANSYS 6.0. Para cada malha considerada, apresenta-se a solução do problema referente aos deslocamentos associados a um nó físico da malha. Observa-se que o programa ANSYS possui um estimador de erros para a versão- p que é usado localmente (i.e., em cada elemento da malha) o que implica num refinamento polinomial não-uniforme. O critério de parada do refinamento

é, em geral, estipulado pelo usuário sendo que nas situações consideradas esse critério foi fixado com o valor de 1% para a diferença entre as energias de deformação de dois refinamentos sucessivos. Dessa forma, a ordem polinomial atribuída às soluções geradas por esse programa, na tabela a seguir, representam a ordem p de aproximação predominante. Esse valor de p ainda se refere ao valor máximo encontrado na malha. No caso da formulação proposta, p representa a ordem polinomial uniformemente distribuída sobre a malha.

Tabela 3.1: Comparação pontual entre as soluções produzidas pela formulação p proposta e aquela empregada no programa ANSYS 6.0 para o caso da viga em balanço.

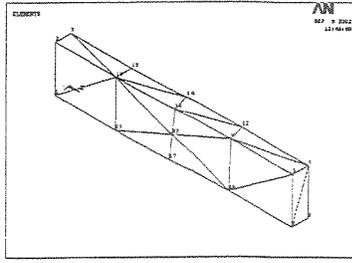
Malha	Nó ⁴	Formulação	Deslocamento u [m]		
			u_x	u_y	u_z
34 elem.	1	ANSYS	0,54665e-05	-0,32492e-04	-0,29401e-08
$p = 4$		Proposta	0,54970e-05	-0,32792e-04	0,60912e-06
65 elem.	1	ANSYS	0,54679e-05	-0,32501e-04	-0,80350e-08
$p = 4$		Proposta	0,55637e-05	-0,32834e-04	0,10157e-06
102 elem.	1	ANSYS	0,54621e-05	-0,32470e-04	0,14558e-09
$p = 3$		Proposta	0,54983e-05	-0,32734e-04	0,31620e-06

Os resultados obtidos para as duas formulações apresentam estreita coerência entre si o que demonstra um comportamento pontual bastante razoável para as soluções geradas a partir da formulação proposta. Observa-se que na solução obtida pelo programa ANSYS para o segundo caso, dos 65 elementos da malha, 19 possuíam ordem $p = 2$ e 2, ordem $p = 3$. No terceiro caso, dos 105 elementos da malha, 41 possuíam ordem $p = 2$.

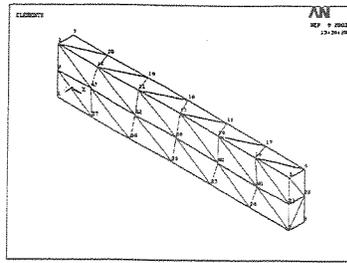
Viga bi-engastada submetida ao peso próprio e a uma carga de superfície

Este exemplo considera uma viga engastada em ambas as extremidades submetida a um carregamento produzido pelo seu peso próprio e por uma carga de superfície de valor constante. A partir da formulação p do MEF desenvolvida anteriormente, obtém-se a solução desse problema para três malhas distintas com 35, 84 e 145 elementos tetraédricos. A ordem polinomial dos espaços de aproximação variam entre $p = 1$ e $p = 8$. As Figuras 3.15(a), 3.15(b) e 3.15(c) ilustram a disposição dos elementos tetraédricos e dos nós físicos para as três malhas consideradas. As Figuras 3.16(a), 3.16(b) e 3.16(c) trazem uma representação esquemática das condições de contorno associadas às três malhas do problema. As dimensões do comprimento, da largura e da espessura da viga são dadas respectivamente por: $c = 3,0 m$, $l = 0,6 m$, $h = 0,2 m$. As constantes físicas desse problema

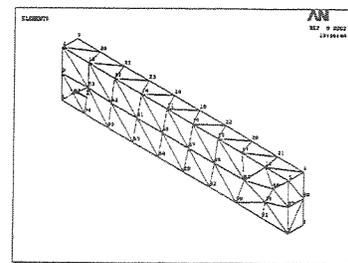
⁴Vide Figura 3.12.



(a) 35 elementos.

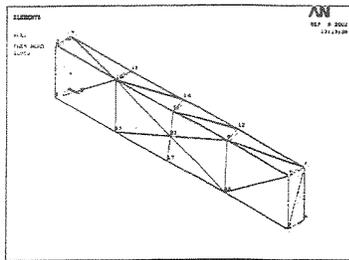


(b) 84 elementos.

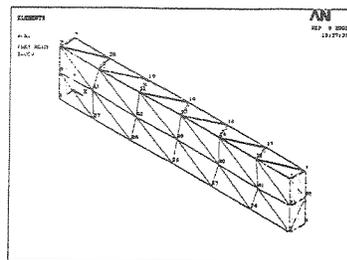


(c) 145 elementos.

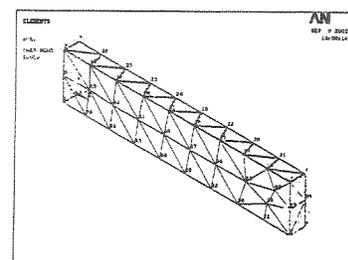
Figura 3.15: Malhas empregadas na análise da viga bi-engastada submetida ao peso próprio e a uma carga de superfície de valor constante.



(a) Malha de 35 elementos.



(b) Malha de 84 elementos.



(c) Malha de 145 elementos.

Figura 3.16: Condições de contorno aplicadas sobre a viga bi-engastada.

são idênticas às do problema anterior e a carga de superfície, nesse caso, possui o valor constante de $2 \times 10^4 \text{ N/m}^2$ na direção oposta ao eixo y .

A Figura 3.17 mostra, para o caso da malha de 35 elementos, a deformação sofrida pela viga após a obtenção da solução do problema.

Os gráficos 3.18(a), 3.18(b) e 3.18(c) comparam os padrões de convergência das soluções aproximadas obtidas a partir das formulações h e p do MEF para as três malhas consideradas. A convergência das soluções é medida com base no comportamento do erro relativo na norma energia em função do número de graus de liberdade envolvidos. Os refinamentos do tipo p são representados pelas linhas cheias com a ordem polinomial de aproximação variando uniformemente entre 1 e 8. Os refinamentos do tipo h são representados pelas linhas tracejadas. Essas linhas foram extrapoladas a partir das soluções obtidas para as malhas de 35, 84 e 145 elementos. Nesse caso, foram empregados os mesmos elementos tetraédricos lineares de quatro nós que iniciam o processo de refinamento p .

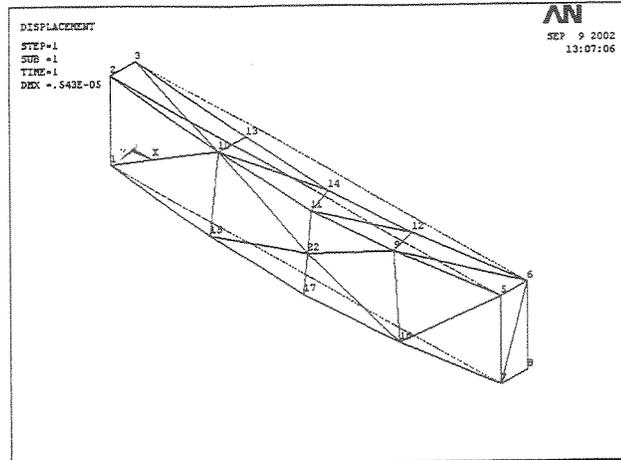
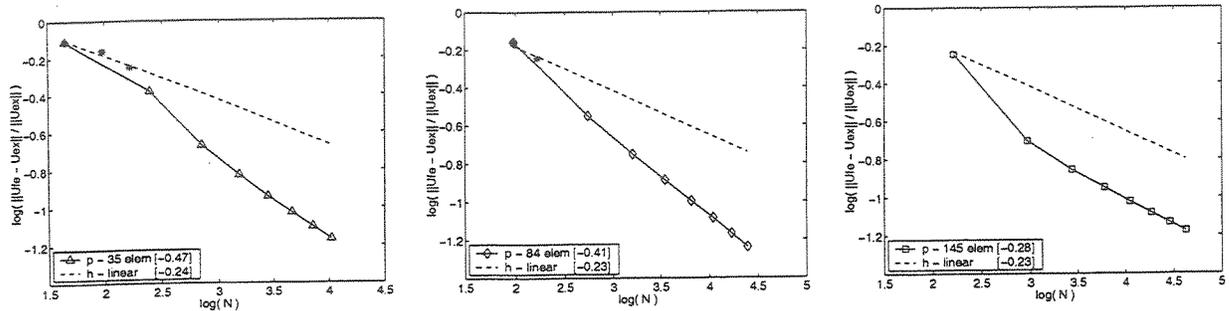


Figura 3.17: Configurações inicial (linha tracejada) e final (linha cheia) da viga bi-engastada (malha de 35 elementos).



(a) Malha de 35 elementos.

(b) Malha de 84 elementos.

(c) Malha de 145 elementos.

Figura 3.18: Decaimento do erro relativo na norma energia em função do número de graus de liberdade envolvidos no problema da viga bi-engastada.

Como no caso anterior, esses gráficos revelam que a taxa de decaimento do erro relativo no caso do refinamento p é aproximadamente o dobro daquela apresentada pelo refinamento h , salvo para a malha com 145 elementos. O resultado para as duas primeiras malhas é coerente com o que se poderia esperar do ponto de vista teórico. No caso da malha com 145 elementos, o mesmo não ocorre. Uma razão para isso poderia estar no fato do comportamento associado ao refinamento h não estar bem representado pela extrapolação das malhas precedentes (35 e 84 elementos) as quais possuem um número relativamente pequeno de graus de liberdade. Outra explicação estaria no comportamento do próprio estimador *a posteriori* utilizado. Nesse caso, o aumento do número de elementos em relação às malhas precedentes poderia produzir uma estimativa de erro extremamente

exigente uma vez que se está combinando duas estratégias de refinamento.

As estimativas obtidas (com precisão de 8 dígitos) para a energia da solução exata em cada uma das malhas consideradas foram

$$\begin{aligned}
 \Pi(u) &\approx 0,06288922 \quad (35 \text{ elem.}), \\
 \Pi(u) &\approx 0,06356032 \quad (84 \text{ elem.}), \\
 \Pi(u) &\approx 0,06344598 \quad (145 \text{ elem.}).
 \end{aligned}
 \tag{3.62}$$

O gráfico 3.19 mostra a evolução do erro percentual das soluções aproximadas obtidas a partir da formulação p desenvolvida ao longo deste capítulo em função da raiz quadrada do número de graus de liberdade associados aos três tipos de malhas físicas consideradas. Da mesma forma que na análise precedente, o refinamento polinomial varia entre $p = 1$ e $p = 8$.

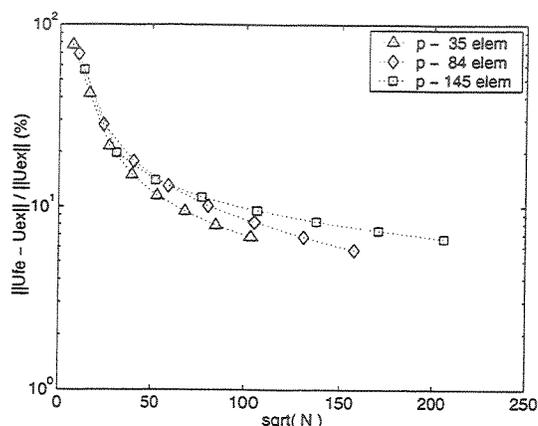


Figura 3.19: Decaimento do erro percentual em função do número de graus de liberdade associados às três malhas consideradas no problema da viga bi-engastada.

Os mesmos comentários feitos para a análise do erro percentual no caso da viga em balanço são pertinentes para o exemplo da viga bi-engastada.

Como no exemplo anterior, a Tabela 3.2 compara as soluções pontuais do problema da viga bi-engastada através das formulações p desse trabalho e do programa comercial ANSYS 6.0. Para cada malha considerada, apresenta-se a solução do problema referente aos deslocamentos associados a um nó físico da malha. O critério de parada do refinamento, no caso do programa ANSYS, foi estipulado com o valor de 1% da diferença entre as energias de deformação de dois refinamentos sucessivos. A ordem polinomial atribuída às soluções geradas por esse programa, na tabela a seguir,

Tabela 3.2: Comparação pontual entre as soluções produzidas pela formulação p proposta e aquela empregada no programa ANSYS 6.0 para o caso da viga bi-engastada.

Malha	Nó ⁵	Formulação	Deslocamento \mathbf{u} [m]		
			u_x	u_y	u_z
35 elem. $p = 5$	17	ANSYS	-0,64665e-07	-0,53358e-05	-0,62001e-07
		Proposta	-0,58054e-07	-0,53388e-05	0,27786e-07
84 elem. $p = 4$	25	ANSYS	-0,10004e-06	-0,53317e-05	-0,66914e-07
		Proposta	-0,11735e-06	-0,53647e-05	-0,16543e-07
145 elem. $p = 4$	34	ANSYS	-0,43737e-06	-0,51671e-05	-0,54641e-07
		Proposta	-0,44096e-06	-0,52243e-05	-0,12485e-06

não é necessariamente uniforme. No caso da formulação proposta, p representa a ordem polinomial uniformemente distribuída sobre a malha.

A partir dessa tabela, nota-se que os resultados obtidos para ambas as formulações são bastante próximos demonstrando que a abordagem seguida neste trabalho pode ser aplicada de forma confiável mesmo no que se refere a análises pontuais. Observa-se que na solução obtida pelo programa ANSYS para o segundo caso, dos 84 elementos da malha, 2 possuíam ordem $p = 3$. No terceiro caso, dos 145 elementos da malha, 9 possuíam ordem $p = 3$.

3.5 Aplicação em Elasticidade Não-Linear

3.5.1 Formulação do problema de elasticidade finita

O intuito desta seção é estender a aplicação da formulação p desenvolvida ao longo deste capítulo a problemas de elasticidade **3D** com não-linearidades geométricas. Da mesma forma que na seção anterior, parte-se de um **PVC** modelo associado ao problema de elasticidade não-linear e deriva-se daí a formulação variacional que precede a discretização por elementos finitos.

As equações envolvidas nesse tipo de problema são essencialmente as mesmas descritas na seção precedente com a diferença de que, no caso não-linear, o campo de deformações linear $\epsilon(\mathbf{u})$ é substituído pelo tensor Lagrangeano de deformação finita $\mathbf{E}(\mathbf{u})$ (ou tensor de *Green*) dado por

$$\mathbf{E}(\mathbf{u}) = \frac{1}{2} (\nabla \mathbf{u} + \nabla \mathbf{u}^T) + \frac{1}{2} \nabla \mathbf{u}^T \nabla \mathbf{u}. \quad (3.63)$$

A descrição geral do **PVC** associado a problemas de elasticidade infinitesimal é mantida nesse

⁵Vide Figura 3.17.

caso (Equações (3.38)) bem como a equação constitutiva que fornece a relação tensão-deformação (Equação (3.39)).

Em geral, a avaliação do tensor de tensões em problemas de elasticidade finita é feita na configuração original do domínio Ω uma vez que a configuração final deformada não é conhecida *a priori*. Nesse sentido, a formulação variacional apresentada em (3.42) é reescrita como:

Encontre o campo de deslocamentos $\mathbf{u} \in \mathbf{V} = \{\mathbf{v}(\mathbf{x}) \in [H^1(\Omega)]^3 : \mathbf{v} = \mathbf{0} \text{ em } \partial\Omega^D\}$ tal que

$$\int_{\Omega} \mathbf{S} : \delta \mathbf{E} dV = \int_{\Omega} \mathbf{b} \cdot \delta \mathbf{v} dV + \int_{\partial\Omega^N} \bar{\mathbf{t}} \cdot \delta \mathbf{v} dS, \quad (3.64)$$

sendo \mathbf{S} o segundo tensor de tensões de *Piola-Kirchhoff* e $\delta \mathbf{E} = D\mathbf{E}(\mathbf{v})[\delta \mathbf{v}]$, a variação de *Gateaux* de \mathbf{E} na direção $\delta \mathbf{v}$.

A discretização do problema variacional de elasticidade finita (Equação (3.64)) pela versão- p do MEF segue as mesmas premissas do caso linear (i.e., utilização do espaço \mathbf{S}^p e das Equações (3.47) e (3.48)) de modo que o sistema de equações algébricas final pode ser escrito da seguinte forma:

$$[\mathbf{K}(u)] \{u\} = \{\mathbf{f}\}, \quad (3.65)$$

sendo $\mathbf{K}(u)$ a matriz de rigidez global não-linear associada ao problema discretizado.

O procedimento completo de discretização do problema variacional (3.64) bem como a linearização de $\mathbf{K}(u)$ pela aplicação do método de *Newton-Raphson* encontram-se detalhados no Apêndice C. A formulação matricial decorrente do procedimento de linearização também é apresentada no mesmo Apêndice. Essa formulação, baseada integralmente no trabalho de (Silva, 2002), revela os aspectos essenciais da implementação de um código de elementos finitos para problemas com não-linearidades geométricas.

3.5.2 Resultados Numéricos para elasticidade não-linear

Depois de se descrever o tipo de problema não-linear que se pretende analisar, considera-se um exemplo para o qual não se dispõe de uma solução analítica conhecida. Nesse caso, estima-se o erro cometido pela aproximação, na norma energia, com base no estimador *a posteriori* sugerido em (Szabó e Babuška, 1991). Ressalta-se que esse tipo de estimador é indiferente ao tratamento

linear ou não-linear do problema de elasticidade examinado.

Para uma malha de discretização com ordem polinomial fixa, apresentam-se gráficos que mostram a evolução das iterações de *Newton-Raphson* ao longo do processo de convergência da solução não-linear segundo três critérios diferentes: o de deslocamento, o de carregamento e o de energia (Bathe, 1996). Para essa mesma malha, compara-se o comportamento das soluções obtidas segundo as abordagens linear e não-linear.

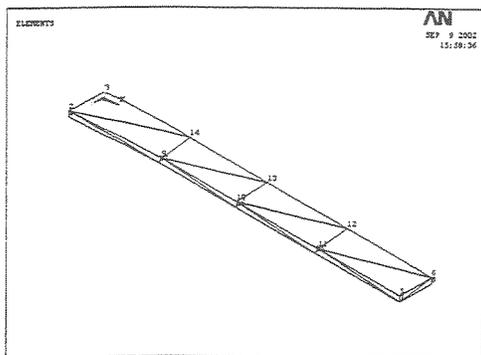
Com o objetivo de se avaliar a qualidade do resultado obtido pela formulação p não-linear desenvolvida, compara-se a solução do problema estudado em um ponto da malha de discretização com a formulação h não-linear para grandes deslocamentos utilizada pelo programa comercial ANSYS 6.0. Nessa situação, em que as formulações são distintas, adota-se um número de graus de liberdade equivalente como parâmetro de comparação.

Viga em balanço submetida ao peso próprio e a uma carga de superfície

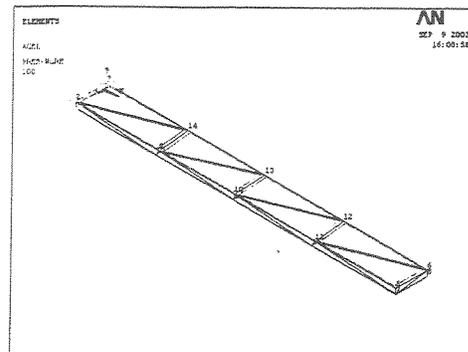
O exemplo não-linear escolhido consiste de uma viga em balanço submetida a um carregamento causado pelo seu peso próprio e por uma carga de superfície de valor constante. Utilizando a formulação p descrita anteriormente, obtém-se a solução desse problema para uma malha de 23 elementos. Nesse caso, a ordem polinomial dos espaços de aproximação pode variar entre $p = 1$ e $p = 8$. A Figura 3.20(a) ilustra a disposição dos elementos tetraédricos e dos nós físicos sobre a malha considerada e a Figura 3.20(b) representa de forma esquemática as condições de contorno envolvidas no problema. As dimensões do comprimento, da largura e da espessura da viga são dadas respectivamente por: $c = 3,0\text{ m}$, $l = 0,3\text{ m}$, $h = 0,05\text{ m}$. As constantes físicas do material que compõe essa viga foram escolhidas de modo que a diferença entre as soluções linear e não-linear para o mesmo problema fosse minimamente observável. Assim, foram adotados os seguintes valores: densidade $\rho = 250\text{ kg/m}^3$, módulo de elasticidade $E = 4 \times 10^8\text{ N/m}^2$ e coeficiente de *Poisson* $\nu = 0,3$. A pressão aplicada sobre a face superior da viga na direção contrária ao eixo y recebe o valor de 100 N/m^2 .

A Figura 3.21 mostra a deformação sofrida pela viga após a obtenção da solução do problema não-linear.

A solução de um problema de elasticidade não-linear normalmente exige que o carregamento



(a) Malha de 23 elementos empregada na análise.



(b) Condições de contorno associadas ao problema.

Figura 3.20: Viga em balanço não-linear submetida ao peso próprio e a uma carga de superfície de valor constante.

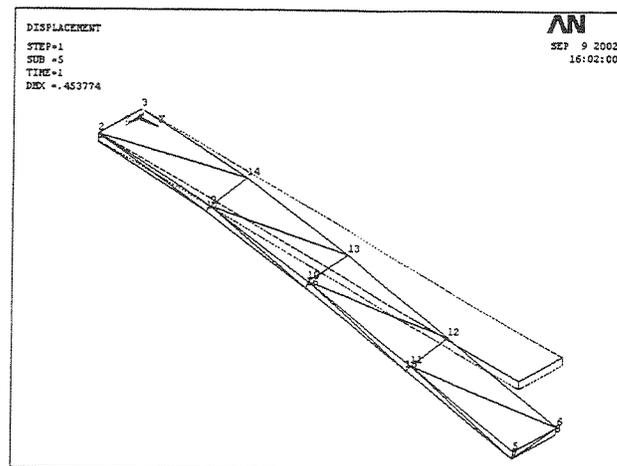


Figura 3.21: Configurações inicial (linha tracejada) e final (linha cheia) da viga em balanço não-linear (malha de 23 elementos).

aplicado seja subdividido em passos de carregamento. Esse procedimento visa melhorar o desempenho global do método de *Newton-Raphson* uma vez a convergência desse método é tanto melhor quanto menos abrupta é a distorção do domínio analisado. No caso particular do exemplo desta seção, a subdivisão do carregamento demonstrou-se desnecessária ao se verificar que as soluções obtidas permaneciam praticamente inalteradas quando a análise realizada utilizava apenas um passo de carregamento. Nesse sentido, o estudo de convergência da formulação p , no caso do exemplo não-linear considerado, leva em conta apenas um passo de carregamento. O critério de parada para as iterações de *Newton-Raphson* foi estabelecido com base na norma euclidiana do vetor gerado pelo produto interno entre o resíduo das iterações (i.e., a diferença entre os carregamentos externo e interno) e o incremento de deslocamento produzido a cada passo do método (esta é a definição do critério de energia⁶). A tolerância estipulada para essa norma foi de 10^{-4} .

Os gráficos 3.22(a) e 3.22(b) mostram o padrão de convergência das soluções do problema da viga em balanço não-linear ao se refinar a ordem polinomial p dos espaços de aproximação. A convergência, nesse caso, é medida com base no comportamento logarítmico ou percentual do erro relativo, na norma energia, em função do número de graus de liberdade envolvidos. O primeiro gráfico traz ainda a comparação entre as formulações h e p do MEF em relação ao decaimento do erro relativo. Nesse gráfico, o refinamento do tipo p é representado pela linha cheia sendo que a ordem polinomial de aproximação varia uniformemente entre 1 e 8. O refinamento do tipo h é representado pela linha tracejada a qual foi extrapolada a partir das soluções obtidas por malhas de elementos tetraédricos lineares com 23, 33, 64 e 108 elementos.

Assim como no caso linear, a análise do primeiro gráfico revela que a taxa de convergência da versão- p em problemas de elasticidade não-linear 3D também possui um decaimento exponencial. Nota-se, nesse caso, que a taxa de convergência associada ao refinamento p é muito superior àquela associada ao refinamento h . Esse resultado demonstra claramente o inexpressivo desempenho dos elementos de *Lagrange* lineares usados na versão- h quando aplicados a problemas desse tipo. Em comparação com o caso linear, o excesso de rigidez introduzido por esse tipo de elemento é ainda mais drástico. O segundo gráfico apenas confirma o bom desempenho da formulação p para malhas 3D não-estruturadas descrita neste capítulo quando aplicada a problemas de elasticidade não-linear.

⁶O critério de carregamento é baseado na norma euclidiana do próprio resíduo das iterações de *Newton-Raphson* enquanto que o critério de deslocamento é baseado na norma euclidiana dos incrementos de deslocamento.

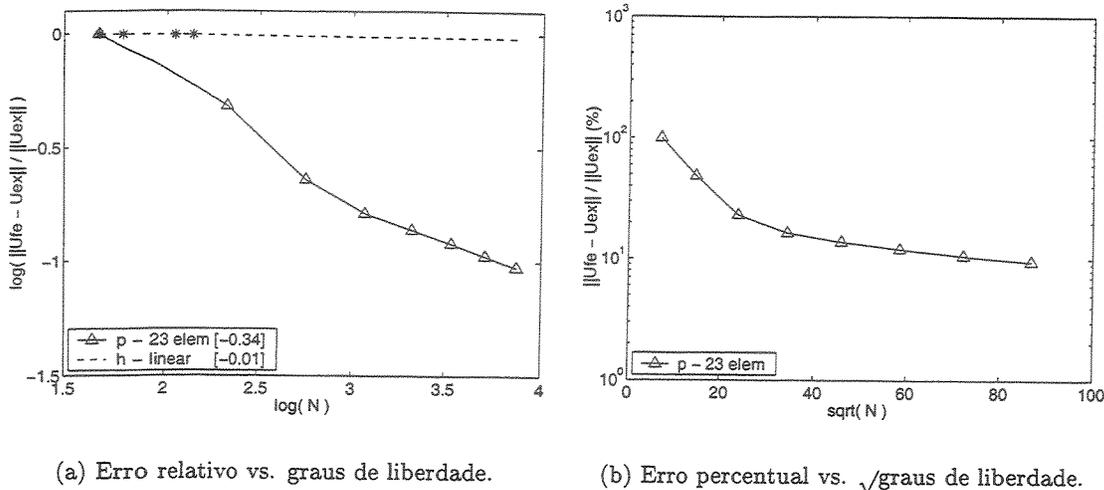


Figura 3.22: Decaimento do erro da solução do problema da viga em balanço não-linear em função do número de graus de liberdade associados aos refinamentos p de ordem 1 a 8.

A estimativa obtida para a energia da solução exata no caso do problema não-linear com malha de 23 elementos foi

$$\Pi(u) \approx 21,19735351. \tag{3.66}$$

Para o caso da malha de 23 elementos descrita no início desta subseção (Figura 3.20(a)), com uma ordem polinomial $p = 5$ uniformemente distribuída e o carregamento subdividido em 5 passos, os gráficos 3.23(a), 3.23(b) e 3.23(c) exibem o processo de solução não-linear através das iterações de *Newton-Raphson* para três critérios distintos de convergência da solução usando a norma euclidiana (norma-2). A avaliação das normas do resíduo (critério de carregamento), do deslocamento (critério de deslocamento) ou da combinação dos dois (critério de energia) fornece um meio bastante preciso de se controlar o processo iterativo de solução.

Observa-se a partir desses gráficos que os critérios de convergência adotados não são absolutamente equivalentes em termos das ordens de grandeza obtidas para as normas. Nota-se que o critério de carregamento apresenta as maiores ordens de grandeza enquanto que o de deslocamento apresenta as menores. As ordens de grandeza associadas ao critério de energia situam-se entre os limites dos outros dois critérios. Por essa razão este último critério tem sido preferencialmente adotado (e.g., o estudo de convergência da formulação p não-linear conduzido anteriormente).

Para a mesma situação descrita na análise dos critérios de convergência acima, o gráfico 3.24

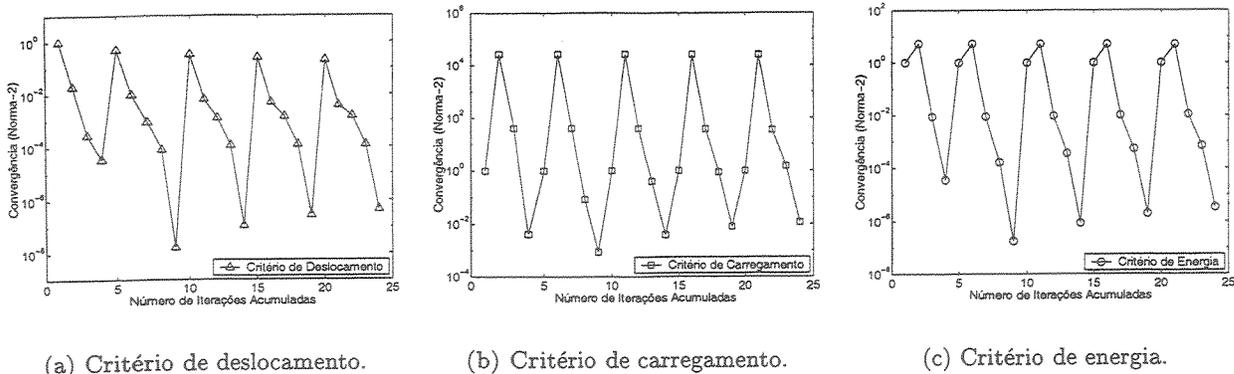


Figura 3.23: Convergência das iterações de *Newton-Raphson* medida na norma-2 (euclidiana) segundo três critérios diferentes em função do número total de iterações necessárias para se chegar à solução do problema da viga em balanço não-linear. Considera-se uma malha com ordem polinomial uniforme $p = 5$.

mostra a diferença produzida entre os deslocamentos (na direção y) de um ponto da malha (nó 5 da Figura 3.21) ao se considerar as abordagens linear e não-linear para esse problema da viga em balanço.

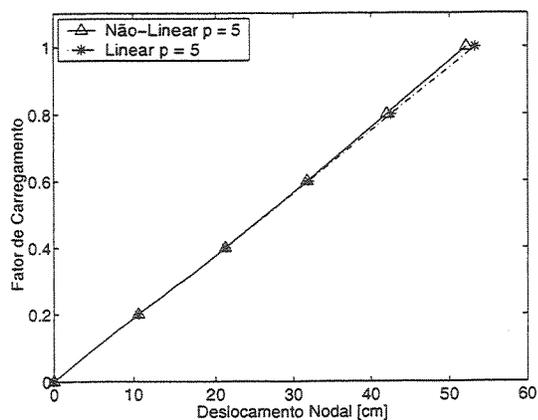


Figura 3.24: Passos de carregamento em função do deslocamento na direção y de um nó sobre a viga em balanço considerada.

Para esse exemplo em específico, como era de se esperar, o deslocamento final no caso da abordagem não-linear é inferior ao da abordagem linear uma vez que esta primeira introduz mais termos no cálculo dos elementos da matriz de rigidez tornando a estrutura da viga menos flexível.

Analogamente ao que foi feito nos exemplos de elasticidade linear, a Tabela 3.3 compara as soluções pontuais do problema da viga em balanço não-linear obtidas através da formulação p

descrita neste trabalho e da formulação h para grandes deslocamentos implementada no programa comercial ANSYS 6.0. Consideram-se duas malhas para o problema: uma do tipo p contendo 23 elementos tetraédricos e ordem polinomial uniforme $p = 5$ e outra do tipo h contendo 303 elementos tetraédricos quadráticos (elemento SOLID92 do ANSYS). Em ambas as malhas, o número total de graus de liberdade (NGL) livres é de 2100. O valor das soluções em termos dos deslocamentos é comparado sobre um nó físico cujas coordenadas espaciais são idênticas em ambas as malhas. A escolha do critério de convergência no caso do programa ANSYS foi feita para os tipos carregamento e deslocamento simultaneamente uma vez que esse programa não dispõe de um critério que combine ambos. Os valores de tolerância adotados para a norma-2 foram como 10^{-3} para carregamento e 10^{-6} para deslocamento (esses valores são coerentes com aqueles apresentados nos gráficos 3.23(a), 3.23(b) e 3.23(c)). No caso da formulação p desenvolvida, adotou-se a tolerância de 10^{-4} para o critério de energia.

Tabela 3.3: Comparação pontual entre as soluções produzidas pela formulação p não-linear proposta e pela formulação h para grandes deslocamentos utilizada pelo programa ANSYS 6.0 para o caso da viga em balanço considerada.

Malha	NGL	Nó ⁷	Formulação	Deslocamento u [m]		
				u_x	u_y	u_z
h	2100	5	ANSYS	-0,46644e-01	-0,52177	0,50587e-04
$p = 5$	2100	5	Proposta	-0,46630e-01	-0,52143	-0,13399e-03

A partir dessa tabela, nota-se que os resultados obtidos para ambas as formulações apresentam uma semelhança razoável demonstrando a eficiência e confiabilidade da abordagem seguida neste trabalho.

3.6 Conclusões

A partir do que foi exposto neste capítulo, é possível concluir que a formulação p proposta por (Karniadakis e Sherwin, 1999) originalmente concebida para a aplicação em problemas de mecânica dos fluidos computacional em malhas **3D** não-estruturadas é perfeitamente aplicável a problemas de elasticidade linear e não-linear com os mesmos tipos de malha. Ao longo do capítulo, foi possível notar uma série de vantagens associadas a esse tipo de formulação como por exemplo a maleabilidade para se tratar os procedimentos de integração e diferenciação numéricas de alta ordem do ponto de

⁷Vide Figura 3.21.

vista local. Em termos globais, pôde-se verificar que a construção de uma aproximação de classe C^0 pode ser obtida de forma consistente e direta (uma vez que esses procedimentos podem ser eficientemente implementados numa etapa de pré-processamento). Em se tratando especificamente da precisão das soluções obtidas para problemas de elasticidade linear e não-linear, pode-se sugerir uma leve modificação nessa formulação de modo que ela incorpore um elemento quadrático em lugar do elemento linear de partida evitando assim aproximações muito pobres na primeira etapa do refinamento p .

Capítulo 4

Métodos Multigrid Algébricos para a Versão- p do MEF

Como visto no capítulo introdutório, o método multigrid clássico é um dos exemplos mais bem sucedidos de técnica iterativa empregada na solução de sistemas algébricos lineares oriundos da discretização de equações diferenciais pela versão- h do MEF. Esse tipo de método se utiliza das vantagens das técnicas de relaxação tradicionais como as de *Gauss-Seidel* e *Jordan* para reduzir rapidamente as altas frequências do erro cometido a cada iteração. A associação dessas técnicas de relaxação ao uso de um conjunto de malhas com diferentes níveis de refinamento produz um decaimento global do erro envolvido no processo iterativo de solução. Apesar das excelentes performances obtidas pelos métodos multigrid, a sua aplicação a problemas com complexas geometrias **3D** apresenta uma série de limitações construtivas uma vez que a geração de uma sequência de malhas sobre esse tipo de geometria nem sempre é uma tarefa trivial, tanto do ponto de vista matemático quanto computacional. Nesses casos, a adoção de técnicas multigrid algébricas é uma alternativa bastante viável.

Algumas das técnicas multigrid algébricas utilizam a versão- p do MEF em lugar da versão- h clássica. Nessa abordagem, o processo iterativo se baseia na relaxação sucessiva dos graus de liberdade hierárquicos a medida que as ordens polinomiais de aproximação variam em diferentes níveis p (em analogia com as diferentes malhas na versão- h). Neste capítulo, discute-se o emprego dos métodos multigrid algébricos associados as estratégias de refino p (casos **2D** e **3D**) e hp (somente casos **2D**) do MEF em problemas elípticos lineares. A eficiência dessa metodologia com respeito ao

custo computacional é aferida através de exemplos **2D** baseados na equação de *Poisson* e exemplos **3D** baseados no operador de elasticidade. Adotam-se malhas de discretização não-estruturadas em ambos os tipos de domínio. As funções de base sugeridas por (Webb e Abouchakra, 1995) e (Sherwin e Karniadakis, 1995) discutidas no Capítulo 2 são empregadas no caso do uso de elementos triangulares e tetraédricos, respectivamente. Na situação particular do problema de *Poisson 2D*, avalia-se ainda o desempenho do pré-condicionador modelo para a versão- p desenvolvido por (Babuška et al., 1991). Finalmente, no caso do refino hp **2D**, a idéia original dos métodos multi- p é estendida para esquemas multi- hp adaptáveis.

Este Capítulo está organizado da seguinte forma: a primeira seção trata dos aspectos teóricos relacionados aos métodos multi- p ; a segunda seção considera as técnicas de pré-condicionamento para a versão- p baseadas nos métodos multi- p e no algoritmo modelo para problemas **2D** proposto por (Babuška et al., 1991); a terceira seção discute brevemente as formas de abordagem associadas às estratégias multi- hp adaptáveis; a quarta seção discute a performance computacional dos vários métodos apresentados nas seções precedentes quando aplicados ao problema de *Poisson 2D*; a quinta seção discute a performance computacional dos métodos multi- p quando aplicados a problemas de elasticidade linear **3D**; e, finalmente, a sexta seção sintetiza os resultados obtidos apresentando as conclusões a respeito do que foi discutido no presente Capítulo.

4.1 Métodos Multi- p

A análise dos métodos multi- p para problemas elípticos tem como ponto de partida a escolha de um problema de valor de contorno modelo definido sobre uma região limitada Ω cuja fronteira $\partial\Omega$ pode ser descrita por uma função suave por partes. Nesse sentido, considera-se a equação de *Poisson* definida sobre o domínio $\Omega \subset \mathbb{R}^2 = \{\mathbf{x} = (x_1, x_2) : x_i \in \mathbb{R}\}$ com condições de contorno de *Dirichlet* homogêneas sobre o bordo $\partial\Omega$

$$\begin{aligned} -\Delta u &= g \quad \text{em } \Omega, \\ u &= 0 \quad \text{em } \partial\Omega, \end{aligned} \tag{4.1}$$

tal que $g \in L^2(\Omega)$ e Δ é o operador de *Laplace* definido por

$$\Delta = \frac{\partial^2}{\partial x_1^2} + \frac{\partial^2}{\partial x_2^2}. \tag{4.2}$$

A forma variacional desse problema, sobre a qual a discretização pelo **MEF** é construída, pode ser escrita como:

Encontrar $u \in H_0^1(\Omega)$ tal que

$$a(u, v) = f(v) \quad \forall v \in H_0^1(\Omega), \quad (4.3)$$

sendo

$$\begin{aligned} a(u, v) &= \int_{\Omega} \nabla u \cdot \nabla v dx, \\ f(v) &= \int_{\Omega} g v dx \end{aligned} \quad (4.4)$$

e $H_0^1(\Omega)$, o espaço de *Sobolev* das funções admissíveis que se anulam no bordo $\partial\Omega$. A existência e a unicidade da solução desse problema podem ser garantidas sob a aplicação de hipóteses comumente associadas a g e $\partial\Omega$.

A aproximação pelo **MEF** é obtida substituindo-se a formulação variacional contínua (4.3) pela seguinte formulação discreta:

Encontrar $u \in V_p$ tal que

$$a(u, v) = f(v) \quad \forall v \in V_p, \quad (4.5)$$

sendo $V_p \subset H_0^1(\Omega)$ o espaço de dimensão finita associado a versão- p do **MEF**. A base $\{\varphi_i\}$ do espaço V_p pode ser construída a partir de qualquer conjunto de funções hierárquicas descritas no Capítulo 2. Nos exemplos analisados mais adiante, são adotadas as bases de (Webb e Abouchakra, 1995) para elementos triangulares e de (Sherwin e Karniadakis, 1995) para elementos tetraédricos.

A solução aproximada u do problema (4.5) pode ser escrita como combinação linear das funções de base φ_i

$$u = \sum_{i=1}^{N_p} u_i \varphi_i, \quad (4.6)$$

sendo N_p a dimensão do espaço V_p .

Como consequência do processo de discretização, o problema (4.5) pode ser reescrito na seguinte forma algébrica equivalente:

$$Au = f,$$

sendo \mathbf{A} a matriz de rigidez global do problema, \mathbf{f} , o vetor de carga global e \mathbf{u} , o vetor solução que fornece os coeficientes u_i de (4.6).

De forma similar ao que ocorre com os espaços de aproximação p (espaços aninhados), as matrizes de rigidez e vetores de carga resultantes do processo de discretização também podem exibir uma estrutura hierárquica. Assim, se \mathbf{A}^p e \mathbf{f}^p são, respectivamente, a matriz de rigidez e o vetor de carga globais correspondentes às funções de base de ordem menor ou igual p , o sistema de equações algébricas resultante $\mathbf{A}^p \mathbf{u}^p = \mathbf{f}^p$ pode ser escrito em blocos como

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{12}^T & A_{22} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}, \quad (4.7)$$

sendo que A_{11} e f_1 correspondem às funções de base de ordem menor ou igual $p-1$, i.e., $A_{11} = A_{p-1}$ e $f_1 = f_{p-1}$.

A idéia de um esquema multigrid para a versão- p do MEF deriva da propriedade hierárquica do sistema algébrico resultante e das características fundamentais estabelecidas para os métodos multigrid, tais como: técnicas de relaxação, iterações aninhadas, correção de malha grossa e operadores de transferência (Hackbush, 1985; Briggs, 1987; Bittencourt, 1996).

Os operadores de transferência (prolongamento e restrição) nos métodos multi- p são naturalmente definidos pelo mapeamento injetivo

$$\mathbf{I}_{p-1}^p = \begin{bmatrix} I_{p-1} \\ 0 \end{bmatrix}, \mathbf{I}_p^{p-1} = \begin{bmatrix} I_{p-1} & 0 \end{bmatrix}, \quad (4.8)$$

sendo I_{p-1} a matriz identidade de ordem $N_{p-1} \times N_{p-1}$ e N_p , o número de graus de liberdade utilizados na discretização p . Observa-se que, de acordo com essa definição, os operadores de transferência estão relacionados pela seguinte expressão:

$$\mathbf{I}_p^{p-1} = \left(\mathbf{I}_{p-1}^p \right)^T. \quad (4.9)$$

Comparativamente aos métodos multigrid do tipo h , os operadores de prolongamento e restrição dos métodos multi- p são bastante simples e independentes da geometria do problema estudado. É possível verificar ainda que esses operadores satisfazem a seguinte relação:

$$\mathbf{A}^{p-1} = \mathbf{I}_p^{p-1} \mathbf{A}^p \mathbf{I}_{p-1}^p. \quad (4.10)$$

As estratégias de ciclos multi- p são construídas em estreita analogia às respectivas estratégias multigrid. Nesse sentido, seguindo a notação proposta por (Briggs, 1987), os algoritmos multi- p de ciclo \mathbf{V} , \mathbf{W} , \mathbf{FMV} e suas derivações são essencialmente idênticos àqueles empregados nas técnicas multigrid com a diferença de que os vários níveis do processo representam diferentes ordens polinomiais p de aproximação e não diferentes malhas (vide Figura 4.1). No caso dos ciclos \mathbf{FMV} , a noção de iteração aninhada é igualmente aplicada sendo que cada ciclo \mathbf{V} é precedido por um ciclo \mathbf{V} menor idealizado para fornecer a melhor aproximação inicial para o nível mais fino seguinte.

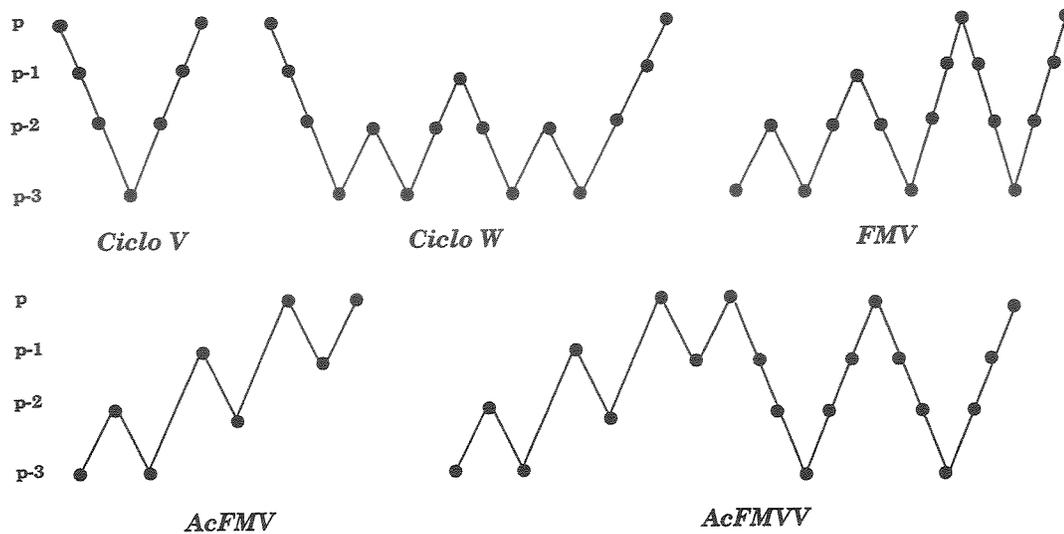


Figura 4.1: Estratégias multi- p .

Uma característica peculiar dos métodos multi- p , derivada da estrutura hierárquica das discretizações do tipo p , que os distingue dos métodos multigrid clássicos é a possibilidade de se decompor o espaço de aproximação V_p numa soma direta

$$V_p = V_{p-1} \oplus V_{-p}, \quad (4.11)$$

sendo V_{-p} o conjunto gerado pelas funções de base de ordem polinomial p exclusivamente. Neste trabalho, o espaço V_{-p} é denominado espaço *complementar*. Essa notável característica permite que se incremente a performance dos ciclos multi- p apenas acrescentando relaxações ao espaço complementar em cada nível do ciclo. Para ilustrar essa possibilidade de refinamento numérico, descreve-se, a seguir, dois algoritmos multi- p de ciclo \mathbf{V} em dois níveis propostos originalmente por (Hu e Katz, 1995).

ALGORITMO MULTI- P de CICLO-V em DOIS NÍVEIS

Seja $P(\mathbf{u}, \mathbf{f})$ um esquema estacionário de relaxação (e.g., Gauss-Seidel) na forma matricial e \mathbf{A}^P e \mathbf{f}^P definidos como em (4.7). Seja $[u_1^0 u_2^0]^T$ uma aproximação inicial para o problema $\mathbf{A}^P \mathbf{u}^P = \mathbf{f}^P$.

1. Execute μ iterações de $P(\mathbf{u}, \mathbf{f})$ em (4.7) de modo a obter a solução $[u_1^\mu u_2^\mu]^T$;

2. Relaxe em

$$A_{11}u_1 = f_1 - A_{12}u_2^\mu \quad (\text{restrição})$$

com aproximação inicial u_1^μ de modo a obter a solução $u_1^{\mu+1}$;

3. Execute μ iterações em (4.7) com aproximação inicial $[u_1^{\mu+1} u_2^\mu]^T$ (prolongamento) para obter $[u_1^{2\mu+1} u_2^{2\mu}]^T$;

4. Se o resíduo for menor que uma dada tolerância, pare; senão faça $[u_1^0 u_2^0]^T \leftarrow [u_1^{2\mu+1} u_2^{2\mu}]^T$ e retorne ao passo 1.

Como consequência do uso do operador de transferência injetivo, é possível observar que os cálculos no nível mais baixo desse ciclo melhoram apenas as componentes em u_1 . Uma pequena modificação nesse algoritmo conduz imediatamente a uma variante que também refina as componentes em u_2 .

ALGORITMO MULTI- P de CICLO-V em DOIS NÍVEIS MODIFICADO

1. idem ao algoritmo anterior;

2. idem ao algoritmo anterior;

3. Execute σ iterações de $P(\mathbf{u}, \mathbf{f})$ nas seguintes equações:

$$A_{22}u_2 = f_2 - A_{12}^T u_1^{\mu+1}$$

com aproximação inicial u_2^μ de modo a obter a solução $u_2^{\mu+1}$;

4. Execute μ iterações em (4.7) com aproximação inicial $[u_1^{\mu+1} u_2^{\mu+1}]^T$ (prolongamento) para obter $[u_1^{2\mu+1} u_2^{2\mu+1}]^T$;

5. Se o resíduo for menor que uma dada tolerância, pare; senão faça $[u_1^0 u_2^0]^T \leftarrow [u_1^{2\mu+1} u_2^{2\mu+1}]^T$ e retorne ao passo 1.

O algoritmo de dois níveis modificado mostra que é possível superar a limitação imposta pelo operador injetivo. Os algoritmos de dois níveis podem ser generalizados de forma a incorporar vários níveis p . Antes de se prosseguir à etapa de generalização, é necessário definir os operadores

de transferência dos espaços complementares. Esses operadores funcionam como recíprocos dos operadores injetivos apresentados anteriormente, i.e.,

$$\mathbf{I}_{-p}^p = \begin{bmatrix} 0 \\ I_{-p} \end{bmatrix}, \mathbf{I}_p^{-p} = \begin{bmatrix} 0 & I_{-p} \end{bmatrix}, \quad (4.12)$$

sendo I_{-p} a matriz identidade de ordem $N_{-p} \times N_{-p}$ e N_{-p} , o número de graus de liberdade correspondentes à ordem polinomial p exclusivamente. Da mesma forma que em (4.9) e (4.10), os operadores de transferência complementares também são dotados das propriedades

$$\begin{aligned} \mathbf{I}_p^{-p} &= \left(\mathbf{I}_{-p}^p \right)^T, \\ \mathbf{A}^{-p} &= \mathbf{I}_p^{-p} \mathbf{A}^p \mathbf{I}_{-p}^p. \end{aligned} \quad (4.13)$$

A partir da definição dos operadores complementares e com base na notação utilizada em (Briggs, 1987), propõe-se o seguinte algoritmo multi- p de ciclo V generalizado em formato matricial:

ALGORITMO MULTI- P de CICLO V GENERALIZADO (MV^p)

$\mathbf{v}^p \leftarrow MV^p(\mathbf{v}^p, \mathbf{f}^p)$

1. Relaxe ν_1 vezes em $\mathbf{A}^p \mathbf{v}^p = \mathbf{f}^p$ com aproximação inicial \mathbf{v}^p ;
2. Se \mathbf{A}^p corresponde a matriz de rigidez do nível mais fino, então vá para 3. Senão,

$$\mathbf{v}_*^p \leftarrow \mathbf{I}_{p-1}^p \mathbf{v}^{p-1}$$

$$\mathbf{f}^{-p} \leftarrow \mathbf{I}_p^{-p} (\mathbf{f}^p - \mathbf{A}^p \mathbf{v}_*^p);$$

$$\mathbf{v}^{-p} \leftarrow \mathbf{I}_p^{-p} \mathbf{v}^p;$$
 Relaxe μ vezes em $\mathbf{A}^{-p} \mathbf{v}^{-p} = \mathbf{f}^{-p}$ com aproximação inicial \mathbf{v}^{-p} ;
3. Se \mathbf{A}^p corresponde a matriz de rigidez do nível mais grosso, então vá para 5. Senão,

$$\mathbf{f}^{p-1} \leftarrow \mathbf{I}_p^{p-1} (\mathbf{f}^p - \mathbf{A}^p \mathbf{v}^p);$$

$$\mathbf{v}^{p-1} \leftarrow \mathbf{0};$$

$$\mathbf{v}^{p-1} \leftarrow MV^{p-1}(\mathbf{v}^{p-1}, \mathbf{f}^{p-1});$$
4. Corrija $\mathbf{v}^p \leftarrow \mathbf{v}^p + \mathbf{I}_{p-1}^p \mathbf{v}^{p-1} + \mathbf{I}_{-p}^p \mathbf{v}^{-p}$;
5. Relaxe ν_2 vezes em $\mathbf{A}^p \mathbf{v}^p = \mathbf{f}^p$ com aproximação inicial \mathbf{v}^p .

Nesse algoritmo, o vetor \mathbf{v}_*^p corresponde ao vetor de resíduo atualizado até a ordem polinomial $p - 1$ (vide vetor $u_1^{\mu+1}$ no passo 2 do algoritmo multi- p de cicloV em dois níveis).

Da mesma maneira, constrói-se o algoritmo para o ciclo FMV analogamente àquele desenvolvido em (Briggs, 1987)

ALGORITMO MULTI- P de CICLO FMV GENERALIZADO (FMV^p)

$\mathbf{v}^p \leftarrow FMV^p(\mathbf{v}^p, \mathbf{f}^p)$

1. Se \mathbf{A}^p corresponde a matriz de rigidez do nível mais grosso, então vá para 3.
Senão,

$$\mathbf{f}^{p-1} \leftarrow \mathbf{I}_p^{p-1}(\mathbf{f}^p - \mathbf{A}^p \mathbf{v}^p);$$

$$\mathbf{v}^{p-1} \leftarrow \mathbf{0};$$

$$\mathbf{v}^{p-1} \leftarrow FMV^{p-1}(\mathbf{v}^{p-1}, \mathbf{f}^{p-1});$$
2. Corrija $\mathbf{v}^p \leftarrow \mathbf{v}^p + \mathbf{I}_{p-1}^p \mathbf{v}^{p-1};$
3. $\mathbf{v}^p \leftarrow MV^p(\mathbf{v}^p, \mathbf{f}^p)$ ν_0 vezes.

Pode-se ainda construir uma versão simplificada desse último algoritmo que é numericamente menos custosa e tão eficiente quanto a do ciclo FMV. Essa versão simplificada, denominada originalmente *método multi- p de ciclo V acelerado* (Hu e Katz, 1995), consiste na execução de um ciclo V de apenas dois níveis, a cada nível do processo, em lugar de um ciclo V completo (vide Figura 4.1).

ALGORITMO MULTI- P de CICLO V ACELERADO ($AcFMVV^p$)

$\mathbf{v}^p \leftarrow FMV^p(\mathbf{v}^p, \mathbf{f}^p)$

1. Se \mathbf{A}^p corresponde a matriz de rigidez do nível mais grosso, então vá para 3.
Senão,

$$\mathbf{f}^{p-1} \leftarrow \mathbf{I}_p^{p-1}(\mathbf{f}^p - \mathbf{A}^p \mathbf{v}^p);$$

$$\mathbf{v}^{p-1} \leftarrow \mathbf{0};$$

$$\mathbf{v}^{p-1} \leftarrow FMV^{p-1}(\mathbf{v}^{p-1}, \mathbf{f}^{p-1});$$
2. Corrija $\mathbf{v}^p \leftarrow \mathbf{v}^p + \mathbf{I}_{p-1}^p \mathbf{v}^{p-1};$
3. Execute $\mathbf{v}^p \leftarrow MV^p(\mathbf{v}^p, \mathbf{f}^p)$ com apenas dois níveis: p e $p - 1$;
4. Se \mathbf{A}^p corresponde a matriz de rigidez do nível mais fino,

$$\mathbf{v}^p \leftarrow MV^p(\mathbf{v}^p, \mathbf{f}^p)$$
 ν_0 vezes.

É importante notar que, em qualquer um dos ciclos aninhados anteriores, o nível mais grosso ou mais fino depende da subrotina que está sendo considerada. Assim, todo ciclo V interno aos ciclos

aninhados sempre possui níveis mais grosso e mais fino locais que não necessariamente correspondem aos níveis mais grosso e mais fino globais. No caso em que o nível global mais grosso é atingido, os níveis locais mais grosso e mais fino são considerados idênticos.

4.2 Pré-Condicionadores para a Versão- p

4.2.1 Pré-condicionadores multi- p

Assim como no caso dos métodos multigrid clássicos, as capacidades dos métodos multi- p podem ser estendidas para incorporar o pré-condicionamento do método de gradientes conjugados (GC). O caráter algébrico dos esquemas multi- p , tanto na aplicação direta quanto no pré-condicionamento, permite que esse método seja usado com espaços de elementos finitos que não possuem uma estrutura hierárquica. Em outras palavras, os métodos multi- p podem ser efetivamente aplicados mesmo quando a matriz de rigidez correspondente às funções de base de ordem menor ou igual p não é uma submatriz da matriz de rigidez correspondente às funções de base de ordem menor ou igual a $p+1$. A única restrição imposta nesse caso é que os graus de liberdade associados às funções de base sejam ordenados de forma hierárquica como no caso do vetor \mathbf{u}^p (Equação (4.7)). Em particular, essa característica permite que se aplique os esquemas multi- p na solução de sistemas de equações lineares baseados na condensação estática de elementos finitos oriundos da versão- p .

O processo de condensação estática de variáveis hierárquicas é um recurso extremamente vantajoso uma vez que reduz sensivelmente o tamanho do sistema linear resultante e conseqüentemente o custo na obtenção de soluções aproximadas. Como na versão- p do MEF as funções de base associadas aos modos internos são de uso obrigatório, a condensação estática das variáveis internas torna-se um processo natural já que esse tipo de procedimento é independente da geometria do problema. Para se condensar as variáveis internas, é necessário arranjar o sistema linear resultante em uma estrutura topológica por blocos como ocorre na classificação das funções de base hierárquicas, ou seja,

$$\mathbf{A}^p = \begin{bmatrix} A_{bb} & A_{bi} \\ A_{bi}^T & A_{ii} \end{bmatrix}, \mathbf{f}^p = \begin{bmatrix} f_b \\ f_i \end{bmatrix}, \quad (4.14)$$

sendo A_{bb} o bloco correspondente às variáveis associadas ao contorno dos elementos (vértices, arestas e faces), A_{ii} , o bloco correspondente às variáveis associadas ao interior dos elementos e A_{bi} , o bloco correspondente ao acoplamento entre as variáveis associadas ao contorno e aquelas associadas ao interior dos elementos.

Nessa estrutura, a condensação é efetuada calculando-se o complemento de *Schur* de \mathbf{A}^p

$$S^p = A_{bb} - A_{bi}A_{ii}^{-1}A_{bi}^T. \quad (4.15)$$

O complemento de *Schur*, na verdade, elimina todas as variáveis associadas ao interior dos elementos. Essa eliminação parcial de graus de liberdade pode ser feita simultaneamente em todos os elementos da malha facilitando assim a paralelização do processo.

É notoriamente sabido que a eliminação parcial de variáveis pode ser entendida como uma técnica particular de pré-condicionamento uma vez que o número de condição $\kappa(S^p)$ é, na pior das hipóteses, igual a $\kappa(\mathbf{A}^p)$ (Mandel, 1990b). Formalmente, tem-se

$$\kappa(S^p) \leq \kappa(\mathbf{A}^p). \quad (4.16)$$

A seguir, discute-se brevemente a aplicação dos esquemas multi- p aos sistemas de equações reduzidos pelo processo de condensação estática. Primeiramente, considera-se o sistema de equações original

$$\mathbf{A}^p \mathbf{u}^p = \mathbf{f}^p$$

descrito na forma matricial por blocos como

$$\begin{bmatrix} A_{bb} & A_{bi} \\ A_{bi}^T & A_{ii} \end{bmatrix} \begin{bmatrix} u_b \\ u_i \end{bmatrix} = \begin{bmatrix} f_b \\ f_i \end{bmatrix}. \quad (4.17)$$

Sobre este sistema de equações, executa-se a transformação

$$\mathbf{Q}\mathbf{A}^p\mathbf{Q}^T(\mathbf{Q}^T)^{-1}\mathbf{u}^p = \mathbf{Q}\mathbf{f}^p, \quad (4.18)$$

sendo

$$\mathbf{Q} = \begin{bmatrix} I & -A_{bi}A_{ii}^{-1} \\ 0 & I \end{bmatrix}, \quad (4.19)$$

que resulta em

$$\mathbf{Q}\mathbf{A}^p\mathbf{Q}^T\mathbf{x}^p = \tilde{\mathbf{f}}^p, \quad (4.20)$$

com $\mathbf{x}^p = (\mathbf{Q}^T)^{-1}\mathbf{u}^p$ e $\tilde{\mathbf{f}}^p = \mathbf{Q}\mathbf{f}^p$.

Em forma matricial, tem-se

$$\begin{bmatrix} S^p & 0 \\ 0 & A_{ii} \end{bmatrix} \begin{bmatrix} x_b \\ x_i \end{bmatrix} = \begin{bmatrix} \tilde{f}_b \\ f_i \end{bmatrix}, \quad (4.21)$$

sendo S^p o complemento de *Schur* obtido em (4.15), \tilde{f}_b , o vetor de carga reduzido dado por

$$\tilde{f}_b = f_b - A_{bi}A_{ii}^{-1}f_i \quad (4.22)$$

e \mathbf{x}^p , o vetor solução transformado que se relaciona ao vetor solução original através da relação

$$\begin{bmatrix} u_b \\ u_i \end{bmatrix} = \begin{bmatrix} I & 0 \\ -A_{ii}^{-1}A_{bi}^T & I \end{bmatrix} \begin{bmatrix} x_b \\ x_i \end{bmatrix}. \quad (4.23)$$

Dessa forma, a solução do sistema algébrico original se reduz à obtenção dos coeficientes associados ao contorno dos elementos pois

$$\begin{aligned} S^p x_b &= \tilde{f}_b, \\ x_b &= u_b. \end{aligned} \quad (4.24)$$

Novamente, o procedimento de construção da matriz S^p e do vetor \tilde{f}_b pode ser feito elemento a elemento já que não há superposição dos coeficientes associados ao interior dos elementos no processo de montagem da matriz global. A ausência de superposição ocorre em razão do suporte das funções de base internas se restringir às fronteiras do elemento.

Uma vez conhecida a solução u_b , é possível retornar à Equação (4.17) para se recuperar a solução associada às variáveis internas, i.e.,

$$A_{ii}u_i = f_i - A_{bi}^T u_b. \quad (4.25)$$

Desse modo, a solução do sistema algébrico dado pelo complemento de *Schur* constitui-se na principal tarefa do processo de condensação. Nesse sentido, os esquemas multi- p desenvolvidos na seção anterior podem ser aplicados diretamente ou como pré-condicionadores na solução das

Equações (4.24). No caso do uso do pré-condicionador multi- p , o algoritmo padrão para o método GC pré-condicionado (GCP) pode ser obtido em (Axelsson e Barker, 1984). Nesse caso, substitui-se a matriz de pré-condicionamento por um único ciclo V como sugerido em (Hu et al., 1997).

Um importante aspecto a se considerar no uso do pré-condicionamento multi- p é a propriedade de simetria dos ciclos utilizados. Essa propriedade assegura a convergência do algoritmo GCP (Hu et al., 1997). Assim, empregando-se um esquema de relaxação simétrico (e.g. *Gauss-Seidel* simetrizado) aos ciclos V , a simetria do método é preservada. Para os ciclos do tipo FMV (aninhados), a propriedade de simetria não pode ser preservada mesmo se um esquema de relaxação simétrico for usado. Portanto, nesses casos, a convergência do método não pode ser garantida.

4.2.2 Pré-condicionador 2D de *Babuska*¹

Assim como vários tipos de pré-condicionador para a versão- p baseados na decomposição de domínios, o algoritmo de *Babuska* pode ser visto como um pré-condicionador de *Jacobi* por blocos após uma mudança de base apropriada. Essa mudança de base é realizada através do cálculo do complemento de *Schur* em relação aos graus de liberdade internos resultando na ortogonalização das funções de base associadas às interfaces (vértices e arestas) com respeito àquelas associadas ao interior dos elementos (exatamente como nas Equações (4.21)). Uma vez efetuada a mudança de base, constrói-se o pré-condicionador a partir de um bloco, relativamente pequeno, associado aos graus de liberdade físicos (vértices apenas), um bloco para cada aresta dos elementos da malha e um bloco para o interior de cada elemento. Em forma matricial esse pré-condicionador é escrito da seguinte forma:

$$\mathbf{B}^{-1} = \begin{bmatrix} [B_{vv}]^{-1} & 0 & 0 \\ 0 & [B_{ee}]^{-1} & 0 \\ 0 & 0 & [B_{ii}]^{-1} \end{bmatrix} \quad (4.26)$$

sendo B_{vv} o bloco associados aos vértices, B_{ee} , o bloco associado às arestas e B_{ii} , o bloco associado aos interiores. Os dois primeiros blocos são fatorados por métodos diretos (e.g., eliminação de *Gauss*) enquanto que o último bloco é obtido por uma equação de recuperação análoga à expressão (4.25).

¹(Babuška et al., 1991)

Por razões computacionais, os blocos associados às arestas B_{ee} são aproximados pelas suas diagonais, ou seja, usa-se um método inexato para se desacoplar os graus de liberdade associados às arestas. Isso reduz substancialmente o esforço computacional na construção do pré-condicionador uma vez que se elimina a necessidade de se montar e fatorar os blocos associados às arestas dos elementos. Apesar desse procedimento imbutir uma aproximação numérica adicional no processo de pré-condicionamento, verificou-se que o uso de estruturas bloco diagonais associadas às arestas dos elementos não aumentam significativamente o número de condição do algoritmo iterativo como um todo (Casarin, 1997).

Neste trabalho, propõe-se uma versão ligeiramente modificada do algoritmo de pré-condicionamento bloco diagonal por arestas apresentado acima. Nesse algoritmo, substitui-se a fatoração direta do bloco associado aos vértices pelo método iterativo padrão de *Gauss-Seidel* usando-se um número fixo de iterações. Esse tipo de abordagem enfraquece as habilidades de aproximação do pré-condicionador original de *Babuska*, no entanto, isso pode ser visto como uma alternativa extremamente barata de se evitar os indesejados efeitos de preenchimento no processo de fatoração do bloco associado às funções de vértice pertencentes às matrizes de rigidez resultantes da discretização de problemas **2D** com muitos graus de liberdade físicos (aproximações *hp* sobretudo).

4.3 Métodos Multi-*hp* Adaptáveis

A combinação das estratégias multi-*p* com os métodos multigrid clássicos sugere à idéia de um método que contemple as discretizações do tipo *h* e *p* simultaneamente. O emprego de um estimador de erros associado a essa combinação leva ao conceito do método multi-*hp* adaptável. Do ponto de vista matemático, a aplicação simultânea desses dois tipos de refinamento num método iterativo multi-*hp* não constitui uma abordagem bem definida uma vez que os graus de liberdade hierárquicos não possuem nenhum significado físico diferentemente dos graus de liberdade nodais. Assim, o método multi-*hp* adaptável pode, basicamente, ser concebido de duas maneiras distintas:

1. empregar o estimador como desencadeador de novas discretizações *hp* e em cada nova malha física aplicar de forma direta os esquemas multi-*p* com aproximação inicial nula;
2. empregar o estimador como desencadeador de novas discretizações *hp* aplicando os operadores de transferência *h* entre malhas para se obter uma aproximação inicial somente para os graus

de liberdade físicos e em seguida, executar os esquemas multi- p com a aproximação inicial obtida.

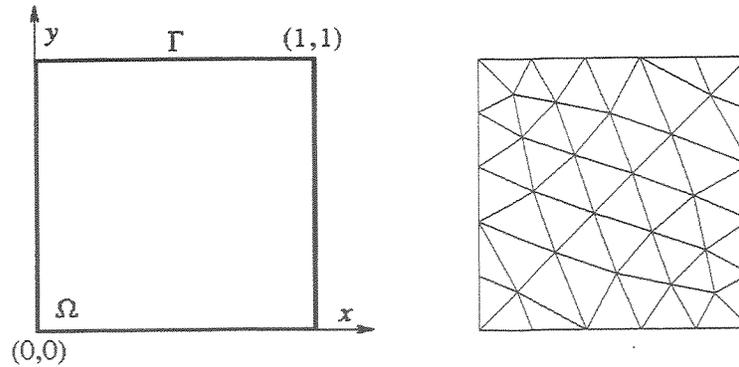
Observa-se que em nenhum desses casos se considera a aplicação de operadores de transferência simultâneos para os graus de liberdade físicos (discretização h) e os hierárquicos (discretização p). Essa forma de se abordar o problema, apesar de simples, viabiliza a construção do método não lhe impondo restrições de caráter teórico.

Neste trabalho, considera-se apenas a primeira das alternativas descritas acima. O design de malhas hp adaptáveis é feito de acordo com a proposta apresentada por (Novotny et al., 1999). Nessa estratégia adaptável, os parâmetros h e p da malha são obtidos simultaneamente em cada elemento através da solução de um problema de minimização com restrição. Nesse caso, minimiza-se o número de graus de liberdade a partir de um dado nível de erro.

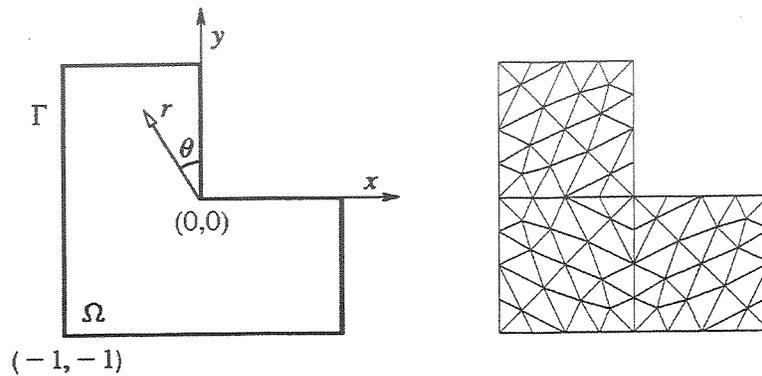
4.4 Resultados Numéricos para o Problema de *Poisson* 2D

Apresentam-se, a seguir, resultados numéricos sobre o desempenho dos métodos multi- p e multi- hp quando aplicados a problemas elípticos escalares bidimensionais. Adota-se como problema modelo a solução da equação de *Poisson* com condições de contorno de *Dirichlet* homogêneas (Equação (4.1)). A partir daí, consideram-se duas configurações para o domínio Ω : uma em formato quadrado e outra em formato L , tal como no Capítulo 2 (Figura 4.2). O termo de carga f é escolhido de modo que a solução u do problema apresente gradientes acentuados no primeiro caso e seja suave no segundo, já que neste último caso Ω apresenta uma singularidade geométrica no ponto $(0, 0)$ (Novotny et al., 1999).

Os esquemas multi- p e multi- hp são analisados em duas situações distintas: com e sem a utilização do procedimento de condensação estática. No caso em que não se emprega a condensação estática, compara-se o desempenho dos métodos multi- p e multi- hp com vários métodos iterativos clássicos como: **SOR**, **GC** e **GCP** com pré-condicionamento diagonal (**GCD**), **SSOR** (**GCSS**) e *Gauss-Seidel* simetrizado (**GCGS**). No caso da utilização da condensação estática, além dos métodos iterativos citados, empregam-se outros três tipos de pré-condicionador para **GCP** na comparação: o pré-condicionador desenvolvido em (Babuška et al., 1991) (**BCMP**), a versão modificada deste último proposta neste trabalho (**BCMPGS**) e, finalmente, o pré-condicionador multi- p de



(a) Problema 1: Ω quadrado; malha 54 elementos.



(b) Problema 2: Ω em L ; malha 108 elementos.

Figura 4.2: Domínios e malhas utilizadas nos dois exemplos analisados (Novotny et al., 1999).

ciclo V descrito anteriormente (**GCMP**).

A análise de desempenho é medida com base no cálculo do número total de operações de ponto flutuante executadas ($Mflops^2$) por cada método iterativo até que seus respectivos algoritmos atinjam convergência. Utiliza-se ainda o critério do *número equivalente de iterações em malha fina*³ para se comparar as diferentes técnicas iterativas consideradas. A metodologia para o cálculo do número de operações executadas é baseada no trabalho de (Bittencourt, 1996). No caso dos métodos

²Cada *flop* é definido como uma única operação computacional de multiplicação ou divisão. Um *Mflop* equivale a 10^6 *flops*.

³O *número equivalente de iterações em malha fina* é definido como o quociente entre o número total de operações de ponto flutuante executadas até que se atinja convergência e o número de operações necessárias para se produzir uma única iteração de um esquema de relaxação (e.g. *Gauss-Seidel*) na malha de discretização mais refinada.

multi- p , o cálculo do número de operações realizadas pelos operadores de transferência é eliminado uma vez que se utilizam operadores injetivos e o cálculo do número de operações realizadas pelas relaxações nos níveis complementares é adicionado pois esse tipo de operação é ausente nos métodos multigrid clássicos (vide Apêndice D).

No que se segue, todas as matrizes globais geradas pelo processo de discretização pelo MEF utilizam uma estrutura esparsa com armazenamento por linhas (Bittencourt, 1996).

4.4.1 Método multi- p sem condensação estática

Na presente análise, fixam-se os valores dos parâmetros $\nu_0 = 1$, $\nu_1 = 2$ e $\nu_2 = 1$ em cada ciclo (vide algoritmos generalizados na seção 4.1). Do mesmo modo, o número de relaxações nos níveis complementares é fixado como $\mu = 1$. O critério de convergência estabelecido é dado por $\|\mathbf{A}\mathbf{u} - \mathbf{f}\|_2 / \|\mathbf{f}\|_2 < \delta$ com precisão $\delta = 10^{-6}$.

As Tabelas 4.1 e 4.2 trazem o número de graus de liberdade (NGL) associados a cada ordem p de aproximação ($2 \leq p \leq 8$) para ambos os problemas considerados.

Tabela 4.1: Multi- p padrão: domínio Ω quadrado.

Ordem p	2	3	4	5	6	7	8
NGL	89	214	393	626	913	1254	1649

Tabela 4.2: Multi- p padrão: domínio Ω em L .

Ordem p	2	3	4	5	6	7	8
NGL	185	439	801	1271	1849	2535	3329

A Tabela 4.3 fornece o número de iterações equivalentes em malha fina necessárias para que cada algoritmo avaliado atinja a precisão δ estabelecida quando se considera o exemplo do domínio Ω quadrado.

A partir da Tabela 4.3, nota-se que, nesse exemplo, as estratégias multi- p são mais custosas que a maior parte dos métodos iterativos clássicos. Nesse contexto, destaca-se o método **GCGS**.

A Figura 4.3 apresenta o número de operações realizadas pelos métodos iterativos e estratégias multi- p considerados em função da ordem p de aproximação (escala semi-log) ou do número de graus de liberdade (escala log-log) associados ao problema com domínio Ω quadrado. No segundo caso, os coeficientes angulares das retas obtidas aparecem entre colchetes ao lado da legenda de cada método

Tabela 4.3: Multi- p padrão: domínio Ω quadrado.
Método - Número de iterações equivalentes

Ordem p	2	3	4	5	6	7	8
V	42	64	63	70	77	83	89
FMV	44	69	70	66	78	91	105
FMVV	49	65	66	65	73	80	88
AcFMV	36	55	67	65	74	83	93
AcFMVV	47	63	63	61	68	74	69
W	42	66	75	85	132	179	211
GC	38	121	184	205	224	235	258
GCD	34	39	39	40	41	42	42
GCSS	36	35	34	34	34	36	37
GCGS	23	27	25	25	23	24	24
SOR	106	108	111	113	116	119	122

iterativo.

Observa-se a partir desses gráficos que os menores custos para a solução do problema 1 usando-se as estratégias multi- p foram alcançados pelo ciclo acelerado **AcFMVV**. Comparativamente aos métodos iterativos clássicos este método ainda é menos econômico. Nesse exemplo, o método **GCGS** demonstrou-se o mais eficiente.

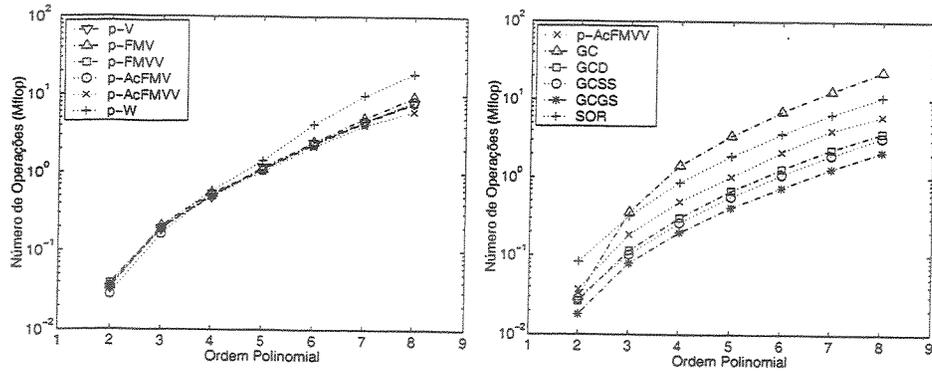
A Tabela 4.4 fornece o número de iterações equivalentes em malha fina necessárias para que cada algoritmo avaliado atinja a precisão δ estabelecida quando se considera o exemplo de Ω com formato L .

Tabela 4.4: Multi- p padrão: domínio Ω em L .
Método - Número de iterações equivalentes

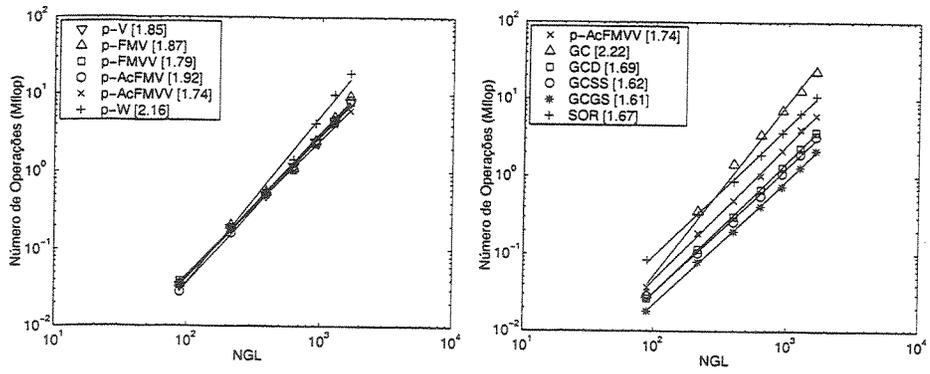
Ordem p	2	3	4	5	6	7	8
V	86	86	96	106	106	115	124
FMV	75	70	71	67	79	92	106
FMVV	74	73	58	66	63	70	77
AcFMV	54	67	68	65	75	66	73
AcFMVV	73	70	63	71	68	64	69
W	86	84	88	115	134	208	251
GC	49	158	222	232	237	259	275
GCD	38	45	48	50	53	56	58
GCSS	48	47	43	46	47	47	49
GCGS	26	30	29	30	31	31	32
SOR	134	141	146	152	155	164	166

No que concerne a quantidade total de iterações equivalentes executadas, os mesmos comentários feitos para o caso do problema 1 se aplicam ao problema 2.

A Figura 4.4 mostra o número de operações executadas pelos métodos iterativos e estratégias



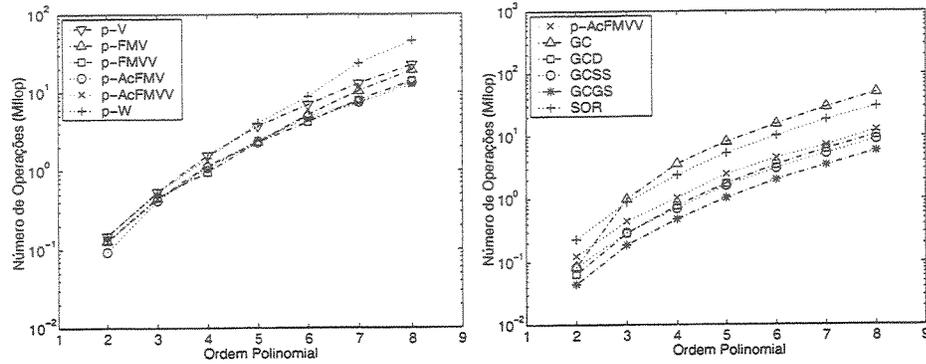
(a) Número de operações (Mflop) vs. Ordem polinomial.



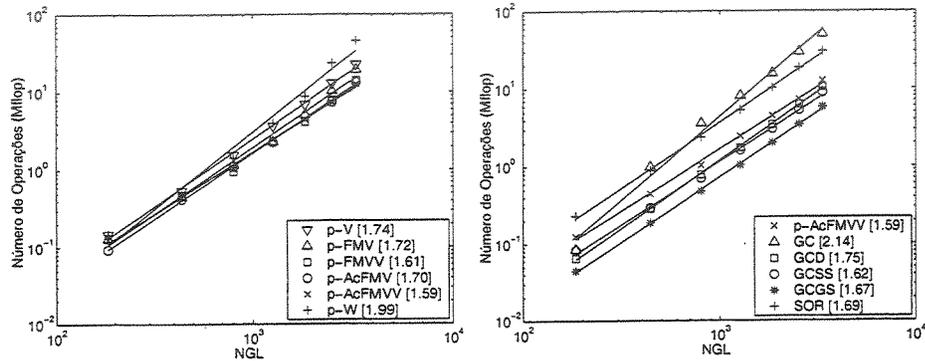
(b) Número de operações (Mflop) vs. Número de graus de liberdade.

Figura 4.3: Problema 1: Ω quadrado. Método multi- p sem condensação.

multi- p considerados em função da ordem p de aproximação ou do número de graus de liberdade associados ao exemplo com domínio Ω em formato L .



(a) Número de operações (Mflop) vs. Ordem polinomial.



(b) Número de operações (Mflop) vs. Número de graus de liberdade.

Figura 4.4: Problema 2: Ω em L . Método multi- p sem condensação.

É possível notar a partir desses gráficos (Figura 4.4) que, para o problema 2, o ciclo **AcFMVV** demonstrou novamente a maior eficiência entre as estratégias multi- p . Relativamente aos métodos iterativos clássicos, apesar do ciclo **AcFMVV** não ter apresentado os melhores resultados em termos absolutos, sua taxa de crescimento com o aumento do número de graus de liberdade hierárquicos foi a menor entre todos os métodos considerados.

4.4.2 Método multi- p com condensação estática

Na presente situação, adotam-se os mesmos valores para os parâmetros ν_0 , ν_1 , ν_2 e μ usados na análise precedente. O critério de convergência utilizado também é o mesmo. Particularmete no

caso do pré-condicionador multi- p , usa-se $\nu_1 = 1$, $\nu_2 = 1$ e $\mu = 1$. Para o pré-condicionador **BCMP** modificado (**BCMPGS**), emprega-se um número fixo de 3 relaxações de *Gauss-Seidel* sobre o bloco associado aos vértices ($p = 1$) a cada iteração do algoritmo **GC**.

As Tabelas 4.5 e 4.6 trazem o número de graus de liberdade associados a cada ordem p de aproximação ($2 \leq p \leq 8$) para ambos os problemas considerados.

Tabela 4.5: Multi- p com condensação: domínio Ω quadrado.

Ordem p	2	3	4	5	6	7	8
NGL	89	160	231	302	373	444	515

Tabela 4.6: Multi- p com condensação: domínio Ω em L .

Ordem p	2	3	4	5	6	7	8
NGL	185	331	477	623	769	915	1061

A Tabela 4.7 fornece o número de iterações equivalentes em malha fina necessárias para que cada método iterativo atinja a precisão δ estabelecida quando se considera o exemplo do domínio Ω com formato quadrado.

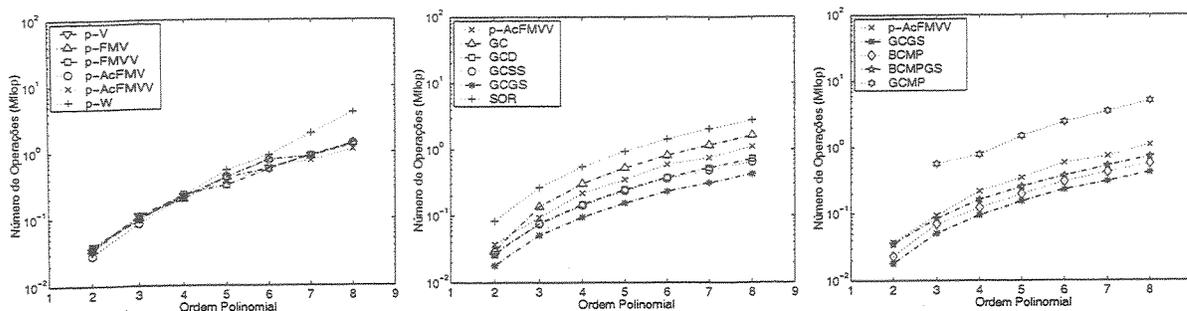
Tabela 4.7: Multi- p com condensação: domínio Ω quadrado.

Método - Número de iterações equivalentes							
Ordem p	2	3	4	5	6	7	8
V	42	53	53	61	58	64	71
FMV	44	49	48	63	79	62	75
FMVV	49	47	57	47	56	65	76
AcFMV	36	40	52	64	76	64	73
AcFMVV	47	44	53	52	60	55	61
W	42	46	54	80	91	141	221
GC	38	65	74	79	82	85	90
GCD	34	36	36	37	38	38	40
GCSS	36	35	35	35	36	35	36
GCGS	23	24	23	23	23	23	23
SOR	106	126	134	140	145	150	153
BCMP	29	33	30	30	31	31	32
BCMPGS	44	40	40	38	38	39	40
GCMP	-	269	193	224	247	259	281

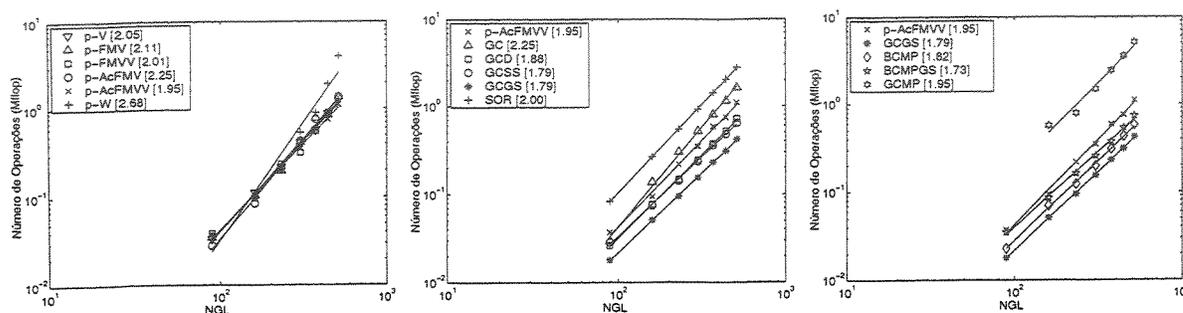
Com base nos resultados fornecidos pela tabela acima, observa-se o mesmo comportamento apresentado na solução do problema 1 sem o uso da condensação estática. Como esperado, o algoritmo **BCMP** teve uma performance notável não superando, no entanto, à do algoritmo **GCGS**. O desempenho do algoritmo **BCMPGS** foi comparável ao do algoritmo **CGD**.

A Figura 4.5 apresenta o número de operações realizadas pelos métodos iterativos e estratégias

multi- p considerados em função da ordem p de aproximação (escala semi-log) ou do número de graus de liberdade (escala log-log) associados ao exemplo com domínio Ω quadrado.



(a) Número de operações (Mflop) vs. Ordem polinomial.



(b) Número de operações (Mflop) vs. Número de graus de liberdade.

Figura 4.5: Problema 1: Ω quadrado. Método multi- p com condensação.

Observa-se, a partir dos resultados apresentados pelos gráficos da Figura 4.5, que o ciclo **AcFMVV** foi o mais efetivo entre as estratégias multi- p . Entretanto, o desempenho desse ciclo ainda esteve abaixo daquele apresentado pela maioria dos métodos iterativos clássicos com destaque para o método **GCGS**. Na categoria dos pré-condicionadores, o algoritmo modificado **BCMPGS** apresentou a menor taxa de crescimento com o aumento do número de graus de liberdade hierárquicos.

A Tabela 4.8 fornece o número de iterações equivalentes em malha fina necessárias para cada método iterativo atingir a precisão δ estabelecida quando se considera o caso do domínio Ω com formato L .

Os mesmos comentários feitos para o problema 1 com elementos condensados também são pertinentes no caso do problema 2 sob as mesmas condições.

Tabela 4.8: Multi- p com condensação: domínio Ω em L .

Método - Número de iterações equivalentes							
Ordem p	2	3	4	5	6	7	8
V	86	76	80	92	93	103	100
FMV	75	64	67	63	79	62	76
FMVV	74	62	66	58	56	53	47
AcFMV	54	63	67	65	77	89	74
AcFMVV	73	68	63	63	72	68	75
W	86	74	82	102	154	191	300
GC	49	77	89	94	98	101	106
GCD	38	42	44	47	46	50	51
GCSS	48	49	47	49	48	49	49
GCGS	26	28	29	28	28	29	28
SOR	134	160	169	181	188	193	195
BCMP	37	38	38	40	41	42	43
BCMPGS	86	62	49	51	51	51	54
GCMP	-	157	168	206	196	225	245

A Figura 4.6 mostra o número de operações realizadas pelos métodos iterativos e estratégias multi- p em função da ordem p de aproximação ou do número de graus de liberdade associados ao exemplo com domínio Ω em formato L .

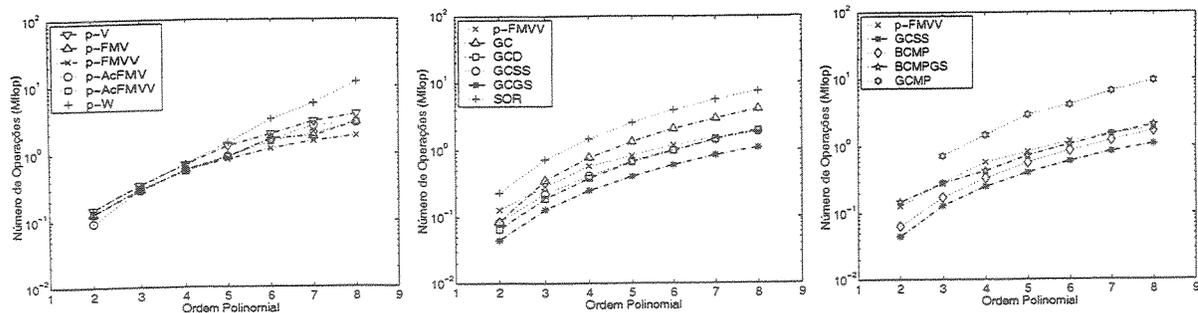
Diferentemente dos casos anteriores, a Figura 4.6 mostra que o ciclo **FMVV** foi o mais efetivo entre as estratégias multi- p . Este ciclo juntamente com o pré-condicionador **BCMPGS** exibiram as menores taxas de crescimento com o aumento do número de graus de liberdade hierárquicos.

4.4.3 Método multi- hp adaptável

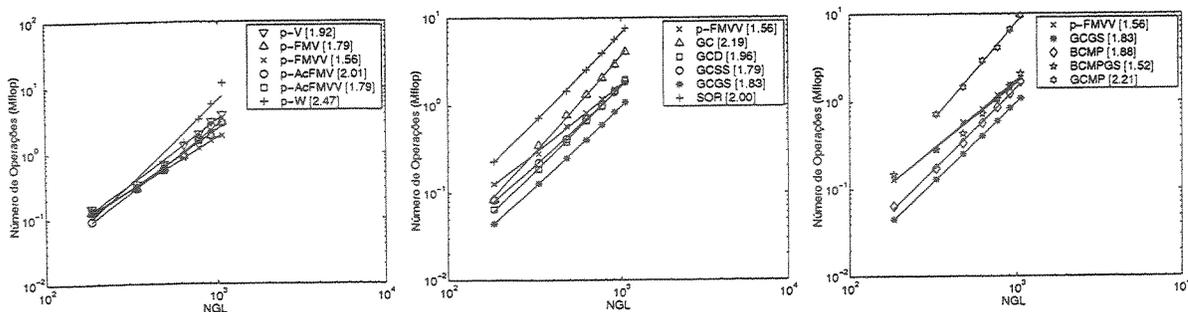
Nesta análise, utiliza-se o primeiro procedimento descrito na seção 4.3. Os valores dos parâmetros ν_0 , ν_1 , ν_2 , μ e δ permanecem os mesmos que os fixados na subseção 4.4.1. Da mesma forma, os parâmetros dos pré-condicionadores multi- p e **BCMPGS** preservam os mesmos valores fixados na subseção precedente.

O número de graus de liberdade (físicos e hierárquicos) de cada malha do processo adaptável, tanto para o exemplo com Ω quadrado quanto em formato L , é fornecido pelas Tabelas 4.9, 4.10, 4.11 e 4.12. As duas primeiras tabelas não consideram o procedimento de condensação estática sobre os elementos da malha diferentemente do que ocorre com as duas últimas tabelas.

A Figura 4.7 mostra o decaimento do erro relativo $\eta = \frac{\|u_h - u\|}{\|u\|} \times 100$ (%) ao longo do processo hp adaptável em função do número de graus de liberdade para ambos os exemplos estudados. Os



(a) Número de operações (Mflop) vs. Ordem polinomial.



(b) Número de operações (Mflop) vs. Número de graus de liberdade.

Figura 4.6: Problema 2: Ω em L . Método multi- p com condensação.

Tabela 4.9: Multi- hp padrão: domínio Ω quadrado.

Malha	1	2	3	4	5
NGL	18	583	1335	1858	2769

Tabela 4.10: Multi- hp padrão: domínio Ω em L .

Malha	1	2	3	4
NGL	39	783	665	1117

Tabela 4.11: Multi- hp com condensação: domínio Ω quadrado.

Malha	1	2	3	4	5
NGL	18	447	759	890	1177

Tabela 4.12: Multi- hp com condensação: domínio Ω em L .

Malha	1	2	3	4
NGL	39	643	549	803

resultados para o erro relativo quando se aplica o procedimento de condensação estática são idênticos aos apresentados na Figura 4.7 visto que o desencadeamento do processo adaptável se dá somente após a recuperação da solução associada aos graus de liberdade condensados.

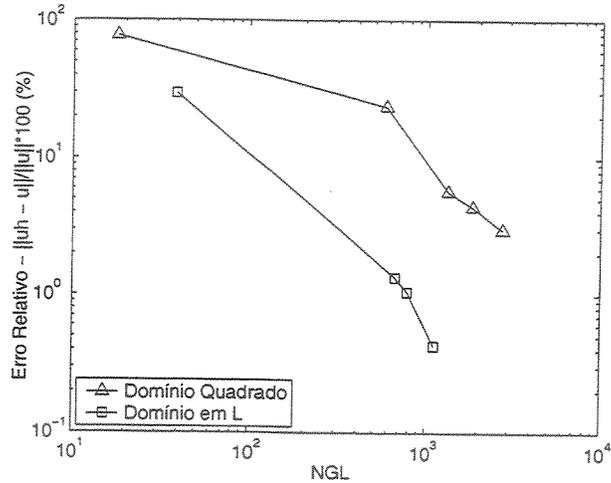


Figura 4.7: Erro relativo η (%) em cada malha do processo adaptável vs. Número de graus de liberdade.

As Tabelas 4.13 e 4.14 fornecem o número de iterações equivalentes em malha fina realizadas para que cada método iterativo alcance a precisão δ estabelecida sendo que a primeira delas considera o exemplo com Ω quadrado e a segunda, com Ω em formato L . Em ambas as tabelas, os valores obtidos foram gerados com elementos não condensados.

Tabela 4.13: Multi- hp padrão: domínio Ω quadrado.

Malha	Método - Núm. iter. equiv.				
	1	2	3	4	5
AcFMVV	-	80	99	78	84
GC	19	197	256	258	266
GCD	21	49	55	55	59
GCSS	20	35	37	37	37
GCGS	20	30	33	32	32
SOR	84	120	129	134	129

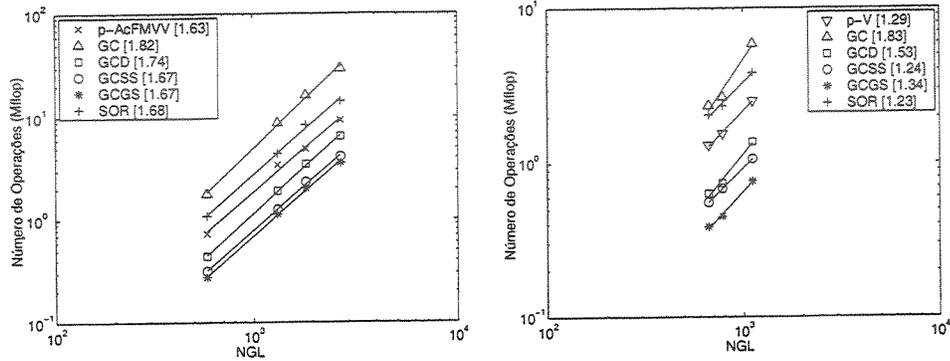
Assim como no caso das estratégias multi- p , os métodos multi- hp são, em geral, mais custosos que os métodos iterativos clássicos.

A Figura 4.8 mostra o número de operações realizadas pelos métodos iterativos e estratégias multi- p considerados em função do número de graus de liberdade (escala log-log) quando se considera os exemplos com domínio Ω quadrado e em formato L . Os coeficientes angulares das retas obtidas

Tabela 4.14: Multi- hp padrão: domínio Ω em L .

Método - Núm. iter. equiv.				
Malha	1	2	3	4
V	-	149	146	138
AcFMVV	-	116	113	119
GC	18	257	263	321
GCD	20	71	71	75
GCSS	24	66	62	58
GCGS	20	43	43	42
SOR	76	226	230	210

aparecem entre colchetes ao lado da legenda de cada método iterativo. Ambos os gráficos gerados consideram apenas elementos não condensados.



(a) Problema 1: Ω quadrado.

(b) Problema 2: Ω em L .

Figura 4.8: Número de operações (Mflop) vs. Número de graus de liberdade. Método multi- hp sem condensação.

A partir da Figura 4.8, observa-se que o método iterativo **GCGS** apresentou, para ambos os exemplos, os melhores resultados entre todos os algoritmos considerados. A menor taxa de crescimento do número de operações com o aumento dos graus de liberdade (físicos e hierárquicos) foi alcançada pela estratégia **AcFMVV** no caso do exemplo 1 (domínio Ω quadrado). Para o exemplo 2 (domínio Ω em formato L), as menores taxas de crescimento foram obtidas para os algoritmos **SOR** e **GCSS**.

As Tabelas 4.15 e 4.16 e a Figura 4.9 fornecem resultados análogos aos obtidos acima para a análise dos problemas 1 e 2. Nas situações a seguir, porém, leva-se em conta o uso do procedimento de condensação estática.

As mesmas observações feitas para o caso das estratégias multi- p com condensação estática

Tabela 4.15: Multi- hp com condensação: domínio Ω quadrado.

Método - Núm. iter. equiv.					
Malha	1	2	3	4	5
AcFMVV	-	75	102	94	87
GC	19	83	104	105	106
GCD	21	48	54	55	57
GCSS	20	37	40	41	39
GCGS	20	30	32	32	32
SOR	84	141	156	162	155
BCMP	-	39	41	42	43
BCMPGS	-	56	55	55	53
GCMP	-	227	330	384	343

Tabela 4.16: Multi- hp com condensação: domínio Ω em L .

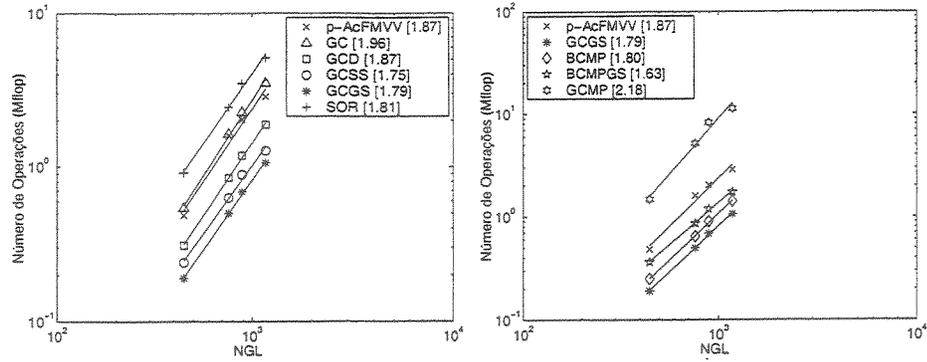
Método - Núm. iter. equiv.				
Malha	1	2	3	4
V	-	160	142	153
AcFMVV	-	108	116	133
GC	18	124	129	134
GCD	20	71	71	71
GCSS	24	68	65	61
GCGS	20	42	42	40
SOR	76	241	238	226
BCMP	-	64	60	62

(veja subseção 4.4.2) se aplicam para os resultados apresentados nas Tabelas 4.15 e 4.16.

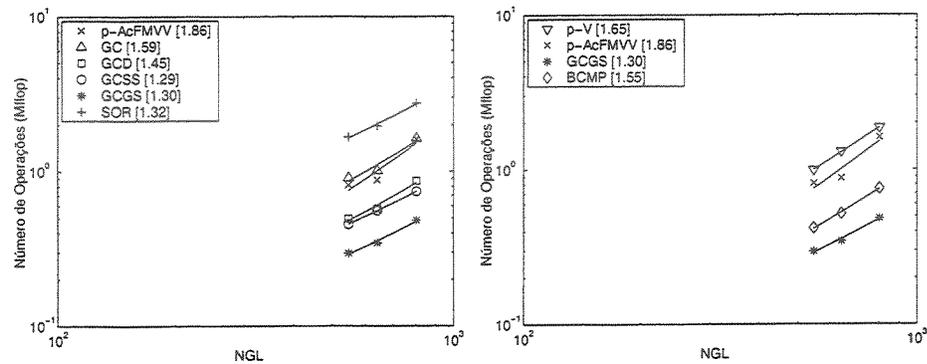
A partir da Figura 4.9, observa-se que o método iterativo **GCGS** demonstrou novamente os melhores resultados para ambos os exemplos analisados. As performances mais efetivas no caso da taxa de crescimento do número de operações com o aumento do número de graus de liberdade foram apresentadas pelo algoritmo **BCMPGS**, no primeiro exemplo, e pelos algoritmos **GCSS** e **GCGS**, no segundo exemplo. Os algoritmos **BCMPGS** e **GCMP** não apresentaram convergência no caso do problema 2 em razão da baixa ordem polinomial ($p = 3$) produzida pelo processo adaptável.

4.5 Resultados Numéricos para Problemas de Elasticidade Linear 3D

Uma vez tendo obtido informações suficientes a respeito do comportamento dos métodos multi- p em problemas 2D, avalia-se, nesta seção, o desempenho computacional desses métodos quando aplicados a problemas de elasticidade linear 3D discretizados com malhas não-estruturadas. As



(a) Problema 1: Ω quadrado.



(b) Problema 2: Ω em L .

Figura 4.9: Número de operações (Mflop) vs. Número de graus de liberdade. Método multi- hp com condensação.

análises efetuadas seguem o mesmo padrão da seção anterior, i.e., compara-se a performance de vários métodos iterativos a partir do cálculo do número total de operações de ponto flutuante executadas pelos seus respectivos algoritmos até que se atinja convergência.

Com base no que foi apresentado na seção anterior, decidiu-se restringir a presente análise aos casos que consideram o procedimento de condensação estática já que o tamanho das matrizes associadas a exemplos **3D** supera largamente o das matrizes **2D**. A aplicação dos pré-condicionadores multi- p em malhas **3D** foi descartada face aos fracos resultados obtidos por essa técnica nos casos **2D**. As técnicas multi- hp adaptáveis foram desconsiderados nesta seção uma vez que não se dispunha de uma estratégia de refino hp para problemas de elasticidade **3D**.

Os exemplos considerados se referem às vigas elásticas lineares do capítulo anterior: viga

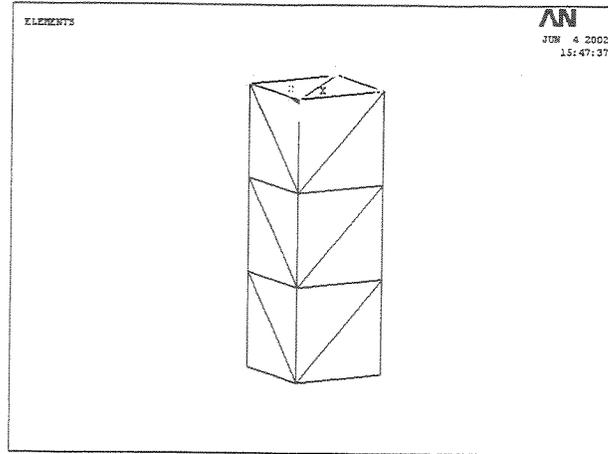


Figura 4.10: Malha de 18 elementos tetraédricos sobre a viga submetida ao seu peso próprio. Exemplo 1 associado à análise dos métodos multi- p 3D lineares.

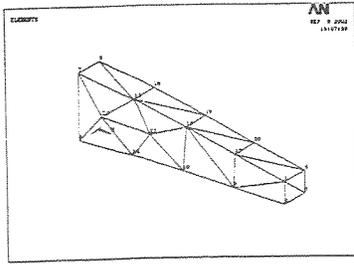
submetida ao peso próprio ($c = 3m$), viga em balanço e viga bi-engastada (Figuras 4.10, 4.11 e 4.12). Nessas três situações, foram adotados os seguintes valores para os parâmetros de controle dos métodos multi- p : $\nu_0 = 1$ (ciclos V internos às estratégias aninhadas), $\nu_2 = 1$ (pós-relaxações na etapa ascendente) e $\mu = 1$ (relaxações nos níveis complementares). No caso do número de pré-relaxações ν_1 executadas na etapa descendente dos ciclos, foi adotado um esquema semi-adaptável no qual a quantidade de relaxações cresce do nível mais fino para o mais grosso com um passo constante. No exemplo da viga submetida exclusivamente ao peso próprio, ν_1 recebeu os seguintes valores, para uma ordem decrescente dos níveis p ($8 \geq p \geq 2$): $\nu_1 = 2, 3, \dots, 8$. No exemplo da viga em balanço, esses valores foram escolhidos como $\nu_1 = 12, 18, \dots, 48$, enquanto que no exemplo da viga bi-engastada adotou-se $\nu_1 = 6, 9, \dots, 24$. O critério de parada δ referente à norma do erro relativo associado resíduo das iterações multi- p foi fixado em 10^{-4} no caso da viga submetida exclusivamente ao peso próprio e da viga bi-engastada e, em 2×10^{-3} no caso viga em balanço.

As Tabelas 4.17, 4.18 e 4.19 trazem, para os três exemplos considerados, o número de graus de liberdade associados a cada ordem p de aproximação ($2 \leq p \leq 8$) após se efetuar o procedimento de condensação estática.

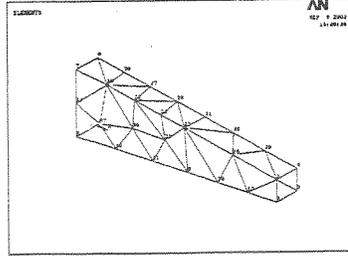
Tabela 4.17: Viga submetida ao peso próprio.

Ordem p	2	3	4	5	6	7	8
NGL	162	432	846	1404	2106	2952	3942

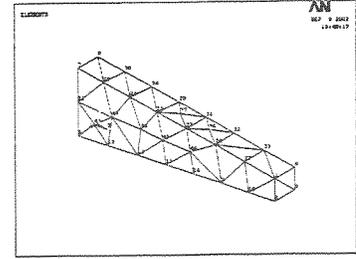
As Tabelas 4.20, 4.21 e 4.22 fornecem o número de iterações equivalentes em malha fina



(a) 34 elementos.

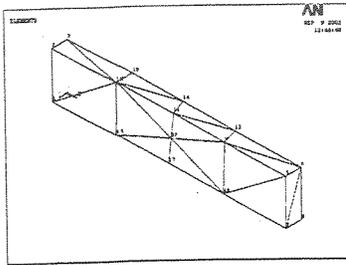


(b) 65 elementos.

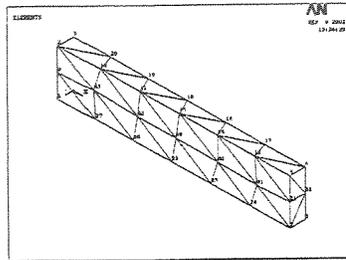


(c) 102 elementos.

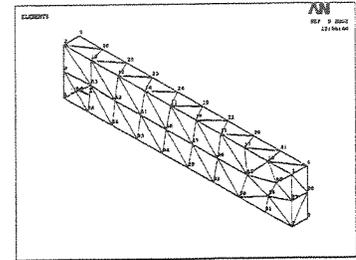
Figura 4.11: Malhas empregadas na análise da viga em balanço submetida ao peso próprio e a uma carga de superfície de valor constante. Exemplo 2 associado à análise dos métodos multi- p 3D lineares.



(a) 35 elementos.



(b) 84 elementos.



(c) 145 elementos.

Figura 4.12: Malhas empregadas na análise da viga bi-engastada submetida ao peso próprio e a uma carga de superfície de valor constante. Exemplo 3 associado à análise dos métodos multi- p 3D lineares.

Tabela 4.18: Viga em balanço.

Ordem p		2	3	4	5	6	7	8
NGL	34 elem.	273	750	1488	2487	3747	5268	7050
	65 elem.	474	1338	2685	4515	6828	9624	12903
	102 elem.	696	1998	4038	6816	10332	14586	19578

Tabela 4.19: Viga bi-engastada.

Ordem p		2	3	4	5	6	7	8
NGL	35 elem.	246	705	1422	2397	3630	5121	6870
	84 elem.	546	1599	3255	5514	8376	11841	15909
	145 elem.	930	2727	5556	9417	14310	20235	27192

necessárias para que cada método iterativo analisado atinja a precisão δ estabelecida quando se consideram os três exemplos de elasticidade linear **3D**. Em cada caso, apresentam-se apenas os resultados relativos às malhas com o menor número de elementos.

Tabela 4.20: Multi- p **3D** com condensação: Viga submetida ao peso próprio.

Método - Número de iterações equivalentes							
Ordem p	2	3	4	5	6	7	8
V	1141	682	521	448	417	402	411
FMVV	577	312	220	160	134	107	91
AcFMV	503	226	174	134	135	108	100
GCD	51	85	121	157	184	207	244
GCGS	43	45	58	60	65	74	75

Tabela 4.21: Multi- p **3D** com condensação: Viga em balanço.

Método - Número de iterações equivalentes							
Ordem p	2	3	4	5	6	7	8
V	8029	5904	5140	4519	3953	3398	2891
FMV	8076	5211	4036	3078	2486	2072	1727
AcFMV	8038	5221	4475	3846	3427	3004	2582
GCD	187	234	431	466	500	543	606
GCGS	142	144	159	167	171	184	196

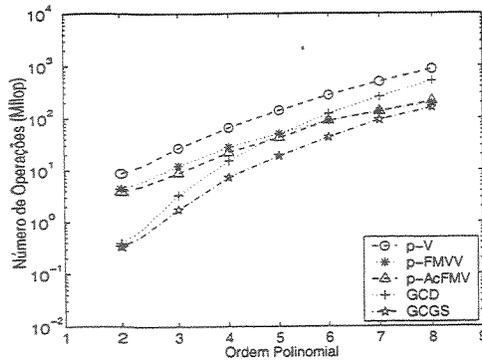
Tabela 4.22: Multi- p **3D** com condensação: Viga bi-engastada.

Método - Número de iterações equivalentes							
Ordem p	2	3	4	5	6	7	8
V	8655	6177	4457	3537	2958	2493	2107
FMV	8283	5094	3261	2362	1802	1452	1198
AcFMV	8265	4958	3503	2847	2383	2025	1719
GCD	259	409	416	455	499	545	544
GCGS	127	143	149	156	173	183	176

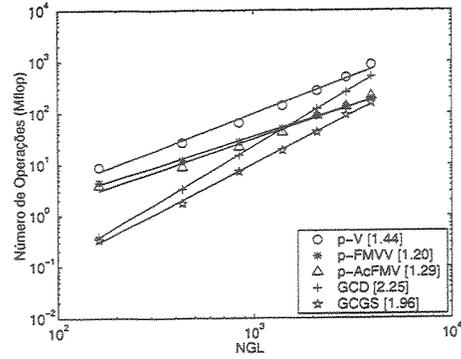
Com base nos resultados fornecidos por essas tabelas, observa-se que, a menos do primeiro exemplo (viga submetida exclusivamente ao seu peso próprio), os métodos iterativos clássicos foram incontestavelmente superiores aos métodos multi- p relativamente ao custo computacional para se obter soluções aproximadas dos problemas considerados. Em todas as análises realizadas, destaca-se a excelente performance alcançada pelo método **GCGS**. É possível notar ainda que as estratégias de ciclos multi- p são tanto melhores quanto maior é o número de níveis p de aproximação.

As Figuras 4.13, 4.14 e 4.15 apresentam o número de operações realizadas pelos métodos iterativos e estratégias multi- p considerados em função da ordem p de aproximação (escala semi-log) ou do número de graus de liberdade (escala log-log) associados aos três exemplos analisados.

Em todos os gráficos apresentados, pode-se verificar novamente o bom desempenho do método

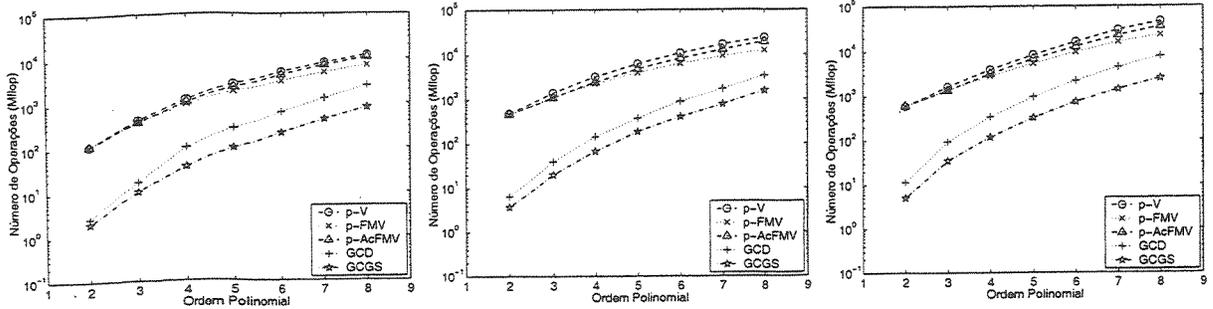


(a) Número de operações (Mflop) vs. Ordem polinomial.

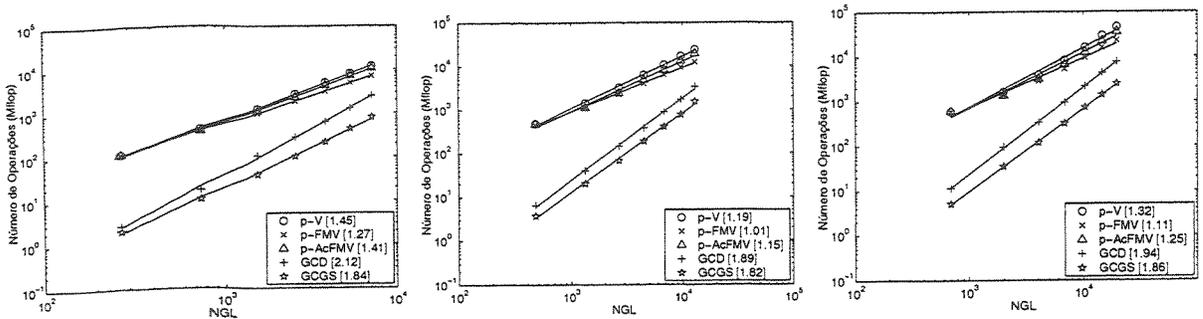


(b) Número de operações (Mflop) vs. Número de graus de liberdade.

Figura 4.13: Viga submetida ao peso próprio. Malha de 18 elementos.

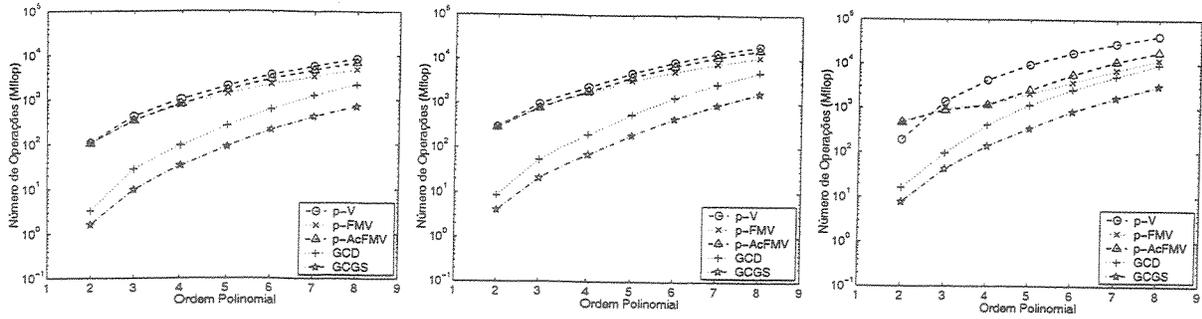


(a) Número de operações (Mflop) vs. Ordem polinomial. Malhas de 34, 65 e 102 elementos.

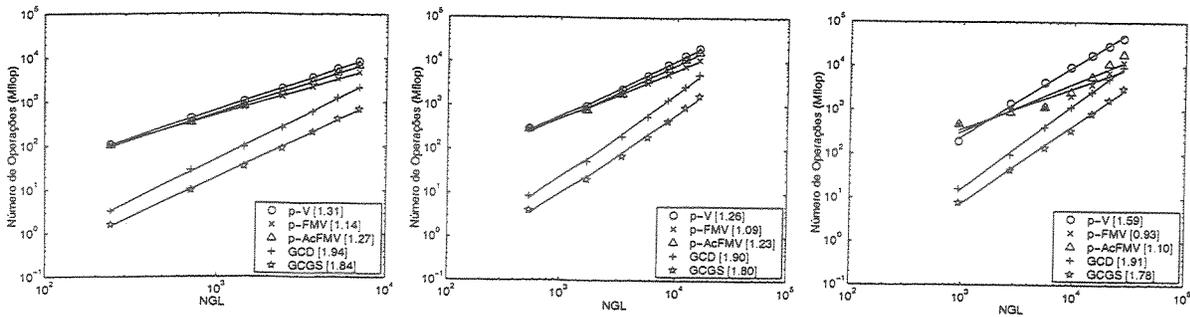


(b) Número de operações (Mflop) vs. Número de graus de liberdade. Malhas de 34, 65 e 102 elementos.

Figura 4.14: Viga em balanço.



(a) Número de operações (Mflop) vs. Ordem polinomial. Malhas de 35, 84 e 145 elementos.



(b) Número de operações (Mflop) vs. Número de graus de liberdade. Malhas de 35, 84 e 145 elementos.

Figura 4.15: Viga bi-engastada.

GCGS. Da mesma forma, observa-se que, entre as estratégias multi- p , os ciclos aninhados do tipo FMV demonstraram os melhores resultados. É relevante mencionar o fato de que as taxas de crescimento do custo computacional dos métodos multi- p são sempre inferiores àquelas associadas aos métodos de gradiente conjugado pré-condicionado. Isso confirma as observações feitas anteriormente em relação a performance dos ciclos multi- p na situação em que se usa um número elevado de níveis de aproximação p (Tabelas 4.20, 4.21, 4.22).

4.6 Conclusões

Com base no que foi apresentado neste Capítulo, é possível constatar a aplicabilidade dos métodos iterativos multi- p 2D e 3D além do pré-condicionador multi- p 2D na solução de sistemas de equações lineares oriundos da versão- p do MEF. Embora a superioridade das técnicas de gradiente conjugado pré-condicionado, em termos do custo computacional, ter sido patente nos exemplos

estudados, ainda resta a possibilidade de se avaliar os métodos multi- p em arquiteturas de código paralelas além de se considerar a aplicação de técnicas de relaxação mais efetivas que o clássico *Gauss-Seidel* sobre os níveis do processo (Guo e Katz, 1998). Nota-se que, pela redução expressiva do número de graus de liberdade na solução do sistema algébrico, o procedimento de condensação estática torna-se fortemente recomendado. Observa-se ainda que a variante do pré-condicionador **BCMP** para problemas **2D** apresentou um comportamento bastante satisfatório podendo se tornar uma alternativa razoável para a solução de sistemas lineares com elevado número de graus de liberdade físicos (e.g., discretizações hp).

Capítulo 5

Implementação de um Código Orientado por Objetos para a Versão- p

Este capítulo pretende situar os aspectos mais relevantes envolvidos na implementação de um código de elementos finitos baseado na formulação p 3D usando-se a filosofia de orientação por objetos (particularmente C++) e o processo unificado de modelagem de *software* (UML) (Booch et al., 1999). Atenção especial é dada às estruturas de dados e aos procedimentos de cálculo que diferenciam a estrutura de programação do tipo p daquela do tipo h .

Partindo-se de uma arquitetura projetada para a versão- h do MEF (Bittencourt, 2000; Silva, 2002), salientam-se os pontos chaves que possibilitam a extensão dessa formulação para a do tipo p . Este capítulo divide-se em cinco seções: a primeira descreve basicamente a arquitetura do modulo de solução original que é comum às versões h e p do MEF; a segunda identifica as classes do conjunto original que sofreram modificações para a inclusão da versão p ; a terceira trata dos aspectos associadas às operações locais que particularizam a estrutura do tipo p em relação à do tipo h ; a quarta se ocupa dos procedimentos globais que devem ser incluídos na estrutura do código h para que esta suporte aproximações p ; e a quinta descreve sucintamente as pequenas modificações que devem ser feitas sobre um algoritmo multigrid clássico para que se obtenha um algoritmo multigrid algébrico baseado na versão- p do MEF.

5.1 Arquitetura do Módulo de Solução

Na programação orientada a objetos, a organização do código se dá basicamente pela definição de classes que contêm as características e os métodos necessários à concretização das idéias do programador. A definição dessas classes se refere particularmente à modelagem dos conceitos de interesse. Isoladamente, isso não reflete o alcance desse tipo de programação que envolve aspectos bem mais elaborados como herança, polimorfismo, sobrecarga de operadores, entre outros. Nesta seção, pretende-se apenas descrever as principais características das classes que constituem a base de programas sobre a qual o código da versão-*p* foi implementado.

O conjunto de classes que compõe a versão atual do código foi elaborado de modo a incluir diferentes conjuntos de funções de interpolação além da solução de problemas não-lineares (Silva, 2002). As funcionalidades da versão atual foram centradas nas classes **DiscreteModel** e **Solver**. A primeira agrega todas as características e atributos do modelo discreto e fornece as interfaces para que outros objetos possam acessar os componentes desse modelo. A segunda contém os algoritmos para a solução dos modelos estruturais lineares e não-lineares. Essa última classe recebe informações de objetos do tipo **DiscreteModel** para poder decidir qual algoritmo deve ser aplicado de acordo com as características do modelo analisado.

A classe auxiliar **SolutionManager** conecta a interface do usuário com as classes associadas a solução do modelo discreto. Em outras palavras, essa classe converte as informações do usuário em informações úteis para as classes que executam os cálculos. Essa classe ainda é responsável pelo gerenciamento dos arquivos ASCII de entrada e saída e pela instanciação dos objetos **DiscreteModel** e **Solver**.

As agregações que formam a classe **DiscreteModel** são mostradas na Figura 5.1. **Coords**, **Groups**, **DOFs**, **DOFBCs**, **LSets**, **Materials**, **NoNo** e **NoEl** são, respectivamente, instâncias de classes de nós (**Nodes**), grupos de elementos finitos (**FEGroups**), equações (**Equations**), condições de contorno de *Dirichlet* (**DOFBoundaryConditions**), carregamentos (**LoadSets**), materiais (**MaterialArray**) e topologias de malha (**NodeNodeNeighborhood** e **NodeElementNeighborhood**). Tem-se ainda a classe **ProblemDescription** que descreve de forma sucinta os parâmetros essenciais do modelo discreto. Essa classe tem por objetivo centralizar as informações a cerca do modelo discreto facilitando a tomada de decisões com relação a quais algoritmos deverão

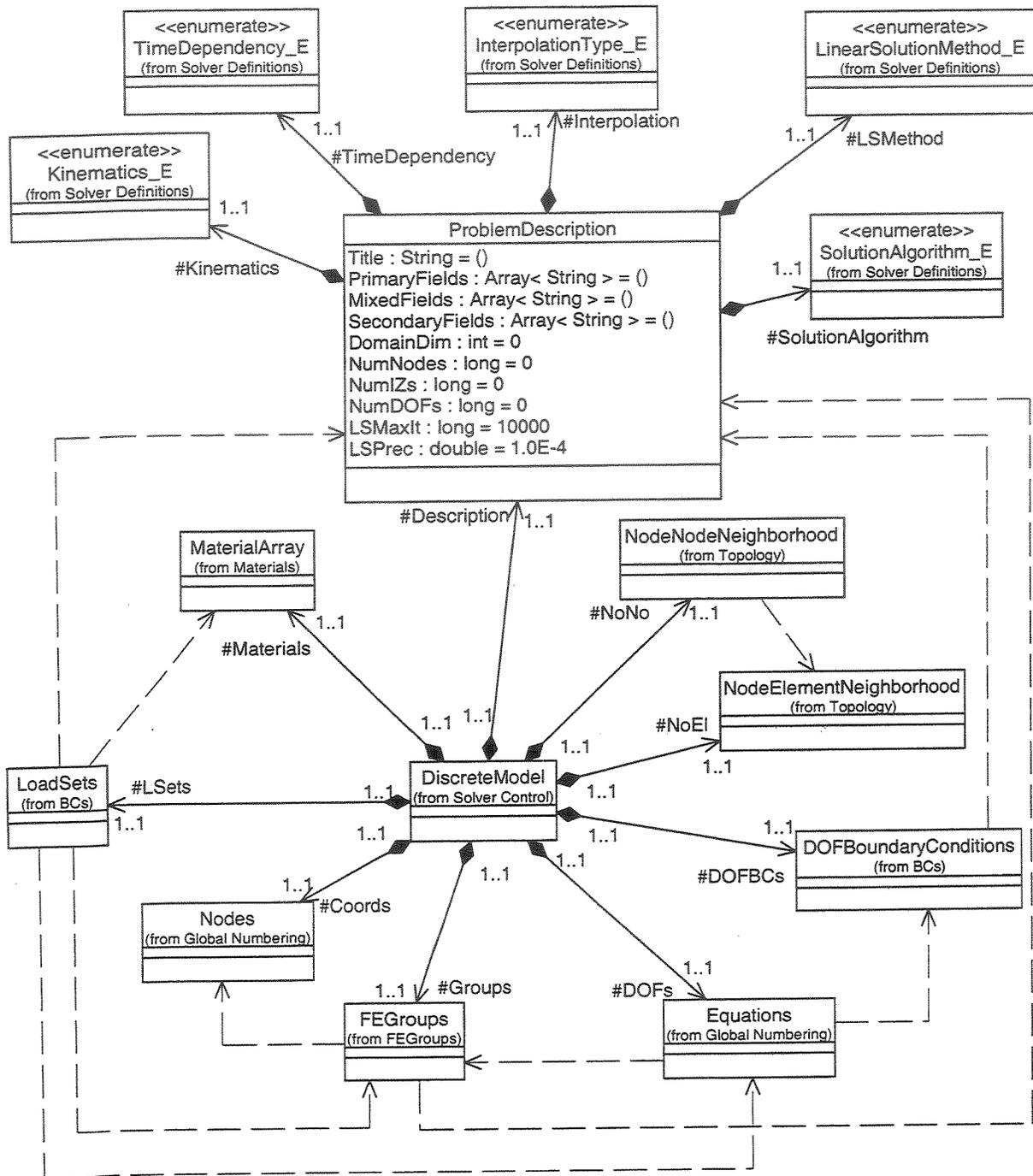


Figura 5.1: Composição da classe DiscreteModel (Silva, 2002).

ser executados bem como auxiliar outros objetos a executar tais algoritmos. A maior parte das entidades nessas classes são construídas com base nas classes templates desenvolvidas em (Bittencourt, 2000).

A classe abstrata **FiniteElement** implementa o conceito de elemento finito. Essa classe representa a unidade de discretização do domínio geométrico do problema sobre a qual se efetuam as operações de mapeamento entre o elemento de referência e os elementos da malha durante a execução dos procedimentos de integração e diferenciação numéricas. Essa classe é responsável ainda pela conexão entre todas as entidades requeridas pelos procedimentos de integração e diferenciação numéricas bem como os de avaliação de quantidades físicas. Os operadores que dependem do modelo cinemático e do tipo de meio considerados tais como matriz de rigidez tangente, vetor de carga local, estimadores de erro e termos de análise de sensibilidade também são manipulados por essa classe. Como exemplos de especialização da classe **FiniteElement**, tem-se as derivações **SolidHexahedric** e **SolidTetrahedric** que tratam de problemas estruturais sólidos (baseados no operador de elasticidade infinitesimal) com elementos de formato hexaédrico e tetraédrico, respectivamente.

O esquema de interpolação da classe **FiniteElement** é encapsulado pela classe **Interpolation**, a qual mantém duas associações com **ShapeFunctions**: uma para interpolação geométrica (**GeoInterpolation**) e outra para interpolação dos campos de resposta (**FieldInterpolation**). Fundamentalmente, a classe **Interpolation** gerencia a interpolação do modelo discreto sendo responsável pela interpolação das variáveis associadas à geometria do domínio bem como ao campo de resposta do problema analisado. Essa classe efetua ainda o cálculo de derivadas globais e de Jacobianos de integração.

ShapeFunctions é uma classe base (abstrata) que define a interface pública dos vários tipos de funções de forma implementados (funções do tipo h e/ou do tipo p). A implementação das expressões das funções de forma bem como de suas derivadas locais é deixada para as especializações dessa classe. Os atributos dessa classe ainda podem armazenar os valores das funções de forma e de suas respectivas derivadas locais nos pontos de integração com o objetivo de se evitar avaliações repetitivas. **SerendipityHexahedric** é um exemplo de especialização da classe **ShapeFunctions** que contém as funções de forma do tipo *Serendipity* (e suas respectivas derivadas locais) associadas ao elemento hexaédrico (versão- h).

NumericalIntegration é uma classe abstrata a partir da qual são derivadas as classes que

contém as regras de integração numérica específicas para regiões uni, bi e tridimensionais. As heranças derivadas dessa classe de integração numérica cobrem ainda os casos de baixa e alta ordem polinomial. Um exemplo de especialização da classe **NumericalIntegration** é a classe **HexahedricGaussLegendre** usada para a integração numérica em elementos hexaédricos (versão-*h*). Essa classe contém os pontos e pesos tensorizados da regra de quadratura de *Gauss-Legendre*;

FiniteDeformation é uma especialização abstrata da classe **FiniteElement** para o tratamento de problemas de elasticidade finita com o uso de elementos de formato genérico (hexaedros, tetraedros, etc.). **FiniteStructuralHexahedra** e **FiniteStructuralTetrahedra** são exemplos de especialização da classe **FiniteDeformation** para o tratamento de problemas de elasticidade finita com elementos de formatos hexaédrico e tetraédrico, respectivamente.

A classe **FiniteElementGroup** agrupa conjuntos de elementos finitos com características similares como propriedades geométricas, formato dos elementos, tipo de material, esquemas de interpolação e integração, etc. Seus principais atributos são a incidência do grupo e um apontador **WorkEl** para **FiniteElement** o qual pode ser inicializado como qualquer uma das especializações dessa classe. Quando é necessário executar algum cálculo sobre a entidade elemento finito como, por exemplo, a construção da matriz de rigidez local de determinado elemento, **WorkEl** recebe todos os dados desse elemento – incidência, coordenadas nodais, numeração das equações de seus graus de liberdade, número de identificação de material e propriedades geométricas – e assume sua identidade específica.

A classe **FEGroups** representa o conjunto de todos os elementos do modelo discreto sendo seu principal atributo um vetor de objetos **FiniteElementGroup**. A classe **GeometricProperties** armazena o conjunto de propriedades geométricas associadas ao modelo cinemático do domínio elementar de integração.

Material é a classe base para a representação de modelos de material. **IsotropicElasticMaterial** é um exemplo de especialização dessa classe abstrata para o tratamento de materiais elásticos isotrópicos lineares. Os modelos de materiais são armazenados na classe **MaterialArray**.

As condições de contorno e os carregamentos são descritos pelas classes **DOFBoundaryConditions** e **LoadSets** (Figura 5.1). **DOFBoundaryConditions** é a classe usada para a manipulação de condições de contorno de *Dirichlet* homogêneas e não-homogêneas as quais são

armazenadas, respectivamente, nas classes **EliminatedDOF** e **PrescribedBC**. **ApplyPrescribedDirichlet** é uma classe responsável pela imposição das condições de contorno de *Dirichlet* não-homogêneas no sistema de equações algébricas final. **LoadSets** é a classe usada para o gerenciamento de todos os conjuntos de carregamentos presentes no modelo discreto (forças de corpo, de superfície e nodais). No caso de problemas lineares essa classe representa o conjunto dos vários tipos de carregamentos aplicados ao modelo e no caso de problemas não-lineares, o conjunto de todos os passos de carregamento. **LoadSet** é uma classe usada na manipulação dos diferentes tipos de carregamento: forças de corpo, de superfície e nodais

Equations é a classe usada para a numeração global dos graus de liberdade do modelo discreto (exclusivamente para a versão-*h*).

No que se refere aos procedimentos de solução especificamente, tem-se as classes **Topology** e **SetSparse**. **Topology** armazena e gerencia as informações topológicas referentes à malha de elementos finitos. As informações obtidas nessa classe são de fundamental importância para a montagem das estruturas esparsas das matrizes globais. **SetSparse** contém funções para alocação e construção de matrizes globais esparsas.

As estruturas de dados associadas a vetores e matrizes são definidas em um conjunto separado de classes denominado *DataStructure* (Bittencourt, 2000). Os métodos de solução de sistemas lineares (e.g., método direto de eliminação de *Gauss*, métodos iterativos como Gradientes Conjugados pré-condicionado, etc.) são implementados nessas classes para cada tipo de armazenamento de dados (e.g., matrizes cheias, simétricas, skyline e esparsas).

5.2 Modificações Efetuadas nas Classes Básicas para Acomodação da Versão-*p*

Nessa seção, descrevem-se basicamente as modificações e inserções realizadas em algumas das classes da seção anterior para se implementar o código baseado na versão-*p* 3D do MEF.

Em relação aos procedimentos de integração numérica, foram realizadas as seguintes modificações:

- **NumericalIntegration** : 1. Inclusão de atributos para o armazenamento das coordenadas e dos pesos das regras de integração multi-dimensionais de alta ordem geradas a partir da tensorização de regras unidimensionais; 2. Definição de uma variável para armazenamento da ordem polinomial dos integrandos associados a construção das matrizes locais; 3. Inclusão de regras para a determinação da quantidade de pontos necessária para a integração exata de funções polinomiais uma vez fornecida a ordem dos integrandos mencionados acima.
- **TetrahedricGaussLRRJacobi** : Especialização da classe **NumericalIntegration** criada para se efetuar integrações numéricas de alta ordem sobre elementos tetraédricos do tipo p . Essa classe calcula os pontos e pesos tensorizados da regra de quadratura de *Gauss-Jacobi* com a distribuição de pontos de *Lobatto* na primeira coordenada local e de *Radau* nas outras duas coordenadas;
- **MultiDimJacobiOperations** : Classe estática criada para manipular as operações fundamentais associadas aos procedimentos de integração numérica do tipo *Gauss-Jacobi* (vide Apêndice B) e de *diferenciação por colocação* (Karniadakis e Sherwin, 1999).

Em relação às funções de base e procedimentos de interpolação, consideram-se as alterações que se seguem.

- **ShapeFunctions** : Adequação da classe original aos procedimentos de integração e diferenciação numéricas de alta ordem.
- **HierarchicalSherwinTetrahedric** : Especialização da classe **ShapeFunctions** criada para se alocar as funções de forma tensorizadas associadas ao elemento tetraédrico de (Sherwin e Karniadakis, 1995). Nesse caso, as derivadas locais das funções de forma não são explicitamente implementadas dentro da classe uma vez que se utiliza o procedimento de derivação numérica;

No caso da manipulação de elementos finitos, as seguintes classes da arquitetura descrita na seção anterior foram alteradas:

- **FiniteElement** : 1. Inclusão de um atributo para o armazenamento da ordem polinomial do elemento de trabalho (*WorkEl*); 2. Adequação do procedimento de cálculo dos vetores

de carga locais para se levar em conta o termo associado ao carregamento de superfície; 3. Inclusão do procedimento de condensação estática local; 4. Alteração dos métodos usados para o cálculo do vetor de carga e das matrizes locais para se garantir a continuidade C^0 da função de aproximação global.

- **SolidTetrahedric** : Alteração do cálculo de estruturas locais (matrizes contendo derivadas das funções de base) no sentido de se garantir a continuidade C^0 da função de aproximação global.
- **FiniteDeformation** : o mesmo que para a classe **SolidTetrahedric**.
- **FiniteStructuralTetrahedra** : o mesmo que para a classe **SolidTetrahedric**.

Para os casos de materiais, grupo de elementos, propriedades geométricas, coordenadas nodais, numeração de equações e condições de contorno, tem-se as seguintes modificações:

- **FiniteElementGroup** : 1. Inclusão de um atributo para o armazenamento de informações associadas a manutenção da continuidade da função de aproximação global; 2. Adequação de procedimentos para a manipulação de informações relacionadas a ordem polinomial de cada elemento pertencente ao grupo de elementos;
- **Equations** : Substituição dessa classe pela classe **P_Equations** projetada especificamente para se gerenciar a numeração de graus de liberdade hierárquicos e as informações associadas à manutenção da continuidade da função de aproximação global.
- **LoadSet** : Adequação dos procedimentos de montagem do vetor de carga global às informações recebidas da classe **P_Equations**.

Finalmente, no que se refere a procedimentos de discretização, controle do processo de solução e métodos numéricos para sistemas de equações, as seguintes classes foram revisadas:

- **Topology** : Adequação de procedimentos de manipulação de informações associadas à topologia da malha de elementos finitos relativamente às informações recebidas da classe **P_Equations**.
- **SetSparse** : Adequação dos métodos para montagem das matrizes globais com relação à manutenção da continuidade da função de aproximação global, à manipulação da ordem polinomial dos elementos da malha e ao procedimento de condensação estática.

- **Solver** : o mesmo que para a classe **SetSparse**.

5.3 Estruturas Locais que Distinguem as Versões h e p

Do ponto de vista local foi possível isolar as operações relacionadas exclusivamente a versão- p através da criação de uma classe estática. Essa classe, denominada **MultiDimJacobiOperations**, não possui atributos, apenas procedimentos que recebem parâmetros vindos das classes base **NumericalIntegration** e **ShapeFunctions** e retornam para essas classes os resultados dos cálculos solicitados. Basicamente, essa classe estática implementa os procedimentos necessários a obtenção de pontos e pesos de integração multi-dimensional de alta ordem associados às regras de quadratura de *Gauss-Jacobi*. Além disso, essa classe também é responsável pelas operações de derivação numérica multi-dimensional de alta ordem baseadas no procedimento de *diferenciação por colocação* apresentado em (Karniadakis e Sherwin, 1999).

A obtenção efetiva dos pontos e pesos de integração, bem como das derivadas locais das funções de forma, necessita essencialmente dos seguintes métodos:

- **JacobiPolynomials** : método para a avaliação dos polinômios de *Jacobi* de peso misto em pontos específicos (geralmente os próprios pontos de integração);
- **DerivJacobiPolynomials** : método para a avaliação das derivadas primeiras dos polinômios de *Jacobi* em pontos específicos;
- **JacobiRoots** : método que implementa o algoritmo de *Newton-Raphson* com deflação polinomial para o cálculo das raízes dos polinômios de *Jacobi* de ordem n (vide seção B.3 do Apêndice B). Essas raízes são os próprios pontos da regra de quadratura unidimensional de *Gauss-Jacobi*;
- **JacobiWeights** : procedimento que contém as expressões para o cálculo dos pesos das regras de quadratura unidimensionais segundo as distribuições de pontos de *Gauss-Jacobi*, *Gauss-Jacobi-Radau* e *Gauss-Jacobi-Lobatto*;
- **GetNum1DIntPoints** : método para a avaliação do número de pontos de integração dada a ordem da função que deve ser integrada;

- `MultiDimIntegPoints` : operação que retorna os pontos e pesos de integração para cada coordenada local tensorizada que compõe a regra de quadratura multi-dimensional para regiões não-estruturadas;
- `CollocationDiffMatrix1D` : procedimento para a construção da matriz de *diferenciação por colocação* unidimensional. Essa matriz é utilizada tanto no cálculo de diferenciações ordinárias quanto parciais. O método empregado no procedimento de *diferenciação por colocação* depende da definição de outros dois procedimentos (`DerivPQ` e `DerivDerivPQ`) relacionados ao cálculo dos polinômios de Lagrange que interpolam a função que deve ser derivada;
- `TensorPartialDerivSF` : operação que calcula as derivadas parciais, em relação às coordenadas tensorizadas, das funções de forma associadas a domínios não-estruturados;
- `TensorCollapseChainDeriv` : operação que efetua a regra de derivação em cadeia das funções de forma associadas a domínios não-estruturados;
- `MultiDimDerivativeSF` : operação que retorna as derivadas parciais, em relação às coordenadas colapsadas dos elementos não-estruturados, das funções de forma hierárquicas. Essas derivadas são avaliadas nos pontos de integração.

As expressões para os pesos das regras de quadratura bem como um algoritmo para o cálculo das raízes dos polinômios de *Jacobi* podem ser obtidos nas seções B.2 e B.3 do Apêndice B. A descrição geral do procedimento de *diferenciação por colocação* pode ser encontrada em (Karniadakis e Sherwin, 1999).

O procedimento de diferenciação parcial das funções de forma envolve o cálculo de matrizes análogas a matriz de diferenciação por colocação unidimensional. Nesses casos, porém, as matrizes não são construídas explicitamente em virtude de suas estruturas acentuadamente esparsas. Isso é claramente verificado quando se observa a forma como o procedimento de diferenciação por colocação é construído. Por exemplo, a derivada parcial em relação a coordenada colapsada ξ de uma função genérica u sobre o elemento de referência tetraédrico T^3 é dada por

$$\frac{\partial u}{\partial \xi}(\xi_i, \eta_j, \zeta_k) = \sum_{r=0}^{Q_1} \sum_{s=0}^{Q_2} \sum_{t=0}^{Q_3} \frac{dh_r(\xi)}{d\xi} \Big|_{\xi_i} h_s(\eta_j) h_t(\zeta_k) u_{rst}, \quad (5.1)$$

sendo $h_r(\xi)$ o polinômio de *Lagrange* obtido nos Q_1 pontos de colocação (que são os próprios pontos de integração) e u_{rst} , os coeficientes da expansão polinomial da função u . A partir daí, pode-se

definir a matriz de diferenciação parcial \mathbf{D}_ξ que, aplicada sobre o vetor \mathbf{u} , representa a derivada da função u em relação a ξ aplicada nos pontos de integração. No caso do domínio tetraédrico colapsado, $\mathbf{D}_\xi \mathbf{u}$ é escrita em termos das coordenadas tensorizadas (a, b, c) tal que

$$\frac{\partial \mathbf{u}}{\partial \xi} = \mathbf{D}_\xi \mathbf{u} = \Lambda \left(\frac{4}{(1-b)(1-c)} \right) \mathbf{D}_a \mathbf{u}, \quad (5.2)$$

sendo

$$\mathbf{D}_a [m(ijk)][n(rst)] = \left. \frac{dh_r(a)}{da} \right|_{a_i} h_s(b_j) h_t(c_k), \quad (5.3)$$

$$\Lambda(f(a, b, c)) [m(ijk)][n(rst)] = f(a_i, b_j, c_k) \delta_{mn} \quad (5.4)$$

e

$$m(ijk) = n(rst) = i + j \cdot Q_1 + k \cdot Q_1 \cdot Q_2. \quad (5.5)$$

O fato de que \mathbf{D}_a é uma matriz muito esparsa pode ser apreciado se for considerada a propriedade de colocação dos polinômios de *Lagrange*, i.e., $h_s(b_j) = \delta_{sj}$ e $h_t(c_k) = \delta_{tk}$. O alto nível de esparsidade se torna particularmente evidente no caso de \mathbf{D}_a em razão da ordenação dos índices m e n dados em (5.5). Nesse caso, a coordenada a é acessada mais rapidamente que b e c de modo que a matriz de derivação parcial apresenta o seguinte formato em bloco:

$$\mathbf{D}_\xi = \begin{bmatrix} \mathbf{D}^{1d} & 0 & 0 & 0 \\ 0 & \mathbf{D}^{1d} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \mathbf{D}^{1d} \end{bmatrix}, \quad (5.6)$$

sendo

$$\mathbf{D}^{1d}[i][r] = \left. \frac{dh_r(a)}{da} \right|_{a_i}, \quad (0 \leq i, r \leq Q_1). \quad (5.7)$$

No caso das derivadas parciais em relação às coordenadas colapsadas b e c , as matrizes \mathbf{D}_b e \mathbf{D}_c não apresentam a estrutura em blocos de \mathbf{D}_a mas continuam tendo o mesmo nível de esparsidade em razão da propriedade de colocação dos polinômios de *Lagrange*. No interior dessas matrizes, as linhas e colunas da submatriz \mathbf{D}^{1d} ficarão organizadas em outros tipos de estrutura esparsa diferentes do formato bloco-diagonal mostrado em (5.6).

5.4 Estruturas Globais Associadas à Versão- p 3D

Da mesma forma que na seção precedente, observa-se que existem procedimentos globais associados à versão- p que não necessariamente dizem respeito à versão- h . Particularmente no caso da numeração dos graus de liberdade globais essa diferença é decisiva em termos da concepção de um código para o MEF. Além da forma peculiar de se numerar os graus de liberdade, a versão- p exige ainda que sejam criados procedimentos para assegurar a colagem das funções locais sobre a malha de elementos de modo a garantir a continuidade C^0 da função de aproximação global.

Diferentemente do caso das operações executadas em nível local, a inclusão das operações globais para a versão- p em uma arquitetura baseada na versão- h é uma tarefa bem menos trivial. Isso se deve basicamente ao fato dos métodos de integração e diferenciação numéricas ocuparem níveis de comunicação bem mais baixos que os de numeração de graus de liberdade e conectividade entre as funções de forma. Em outras palavras, os procedimentos locais estão bem mais isolados que os globais na arquitetura do programa utilizado.

Apesar desse tipo de dificuldade, foi feita a tentativa de se isolar ao máximo as funcionalidades globais que eram exclusivamente relacionadas à versão- p . Nesse sentido, foi criada a classe **P_Equations** contendo os procedimentos de numeração dos graus de liberdade e de conectividade entre as funções de forma associados ao método p . Inicialmente essa classe foi imaginada como uma classe que tivesse o mesmo papel da classe **Equations** mas que funcionasse paralelamente a ela. Isso garantiria a dupla funcionalidade do código. No entanto, percebeu-se a certa altura do desenvolvimento que a arquitetura do tipo h , em certos pontos, não comportava a duplicação de funcionalidades a menos que se abrisse mão totalmente de aspectos relacionados à eficiência do código. Por fim, decidiu-se por particularizar o programa em desenvolvimento para a finalidade proposta.

A seguir, são listados os principais métodos contidos na classe **P_Equations**:

- **Set3DPNumberingDOF** : procedimento que encapsula as diversas rotinas associadas à conectividade das funções de forma e à numeração dos graus de liberdade;
- **ReorderVertex** : rotina relacionada à conectividade das funções de forma e que reordena as incidências dos nós físicos da malha segundo o algoritmo para conectividade entre elementos

tetraédricos apresentado na segunda seção do Capítulo 3. Essa rotina inicialmente reordena as incidências de forma decrescente e em seguida utiliza o procedimento `AnticlockwiseBaseVertex` para posicionar de forma coerente o sistema local de coordenadas associado aos elementos;

- `EdgeNodesGeneration`: procedimento responsável pela geração de nós hierárquicos sobre cada aresta global da malha. Essa tarefa é cumprida com base na própria incidência reordenada pela rotina `ReorderVertex`. Um nó associado a uma aresta global é identificado pela combinação de dois nós físicos da malha (vértices globais);
- `FaceNodesGeneration`: procedimento responsável pela geração de nós hierárquicos sobre cada face global da malha. Novamente, os nós das faces são obtidos a partir da incidência reordenada dos nós físicos. Um nó associado a uma face global é identificado pela combinação de três nós físicos da malha;
- `InteriorNodesGeneration`: procedimento responsável pela geração de nós hierárquicos associados ao interior de cada elemento da malha. Esse procedimento é obtido de forma automática sabendo-se apenas a numeração dos elementos fornecida pelo gerador de malhas;
- `SharedEdgeIdentif`: rotina que retorna para cada aresta global a quantidade e a identificação dos elementos que compartilham esta aresta;
- `SharedFaceIdentif`: rotina que retorna para cada face global a identificação dos dois elementos que compartilham esta face;
- `EdgeFaceOrderDefinition`: rotina que calcula a ordem polinomial de cada aresta e face globais de acordo com a regra do mínimo. Essa regra estabelece que arestas e faces compartilhadas devem conservar a menor ordem polinomial entre os elementos que dividem esta aresta ou face. Observa-se que a ordem polinomial de cada elemento deve ser fornecida pelo arquivo de entrada de dados que descreve o problema a ser analisado;
- `DirichletIdentification`: procedimento que identifica, a partir dos nós físicos eliminados (classe `EliminatedDOF`), as arestas e faces com condições de contorno de *Dirichlet* homogêneas. Os graus de liberdade associados a essas entidades topológicas também serão eliminados antes de se resolver o sistema de equações lineares final;
- `NeumannIdentification`: método que identifica as faces globais da malha do problema que estão sob condições de contorno de *Neumann* não-homogêneas. Observa-se que o arquivo de

entrada que descreve o problema analisado deve fornecer a incidência das faces (triangulares) que possuem esse tipo de condição de contorno para que se possa reconhecer quais elementos deverão receber um tratamento especial ao se calcular seus respectivos vetores de carga;

- `SetNumberPDof` : procedimento que determina a numeração dos graus de liberdade associados a todas as entidades topológicas do modelo discreto (vértices, arestas, faces e interiores). Essa numeração segue exatamente o critério de ordenação dessas entidades, i.e., os graus de liberdade associados aos nós físicos (vértices) são numerados primeiramente seguidos por aqueles associados às arestas, faces e, finalmente, interiores;
- `LocaltoGlobalMapping` : procedimento que mapeia cada grau de liberdade local no correspondente grau de liberdade global associado ao modelo discreto do problema. Assim, para cada grau de liberdade local, encontra-se a sua numeração no sistema global de equações;
- `GlobalContinuity` : rotina responsável pela construção do vetor que indica quais funções hierárquicas deverão ter o seu sinal invertido para que se garanta a continuidade C^0 da função de aproximação global. As posições desse vetor identificam nas matrizes locais os coeficientes que devem ter o seu sinal trocado e o conteúdo das componentes desse vetor representam exatamente o fator multiplicativo que deve ser aplicado a tais funções (i.e., -1);
- `POrderRenumberingDof` : rotina que reordena os graus de liberdade numerados no procedimento `SetNumberPDof` segundo uma ordem polinomial crescente dos coeficientes associados as funções de forma. Essa rotina é essencial ao tratamento de procedimentos iterativos de solução baseados em métodos multi- p .

5.5 Estruturas Básicas em Métodos Multigrid Algébricos

A estrutura de código empregada como base para a construção dos métodos multigrid algébricos associados à versão- p do MEF foi inteiramente definida a partir de (Bittencourt, 1996). Nesse caso, as estratégias multi- p foram implementadas numa única classe denominada `LinearMultigrid`. Essa classe recebe, através de um banco de dados, as matrizes e vetores globais gerados independentemente pelo programa de elementos finitos. Além disso, o banco de dados fornece ainda um vetor contendo o número de graus de liberdade associados a cada nível de aproximação p . A informação contida nesse vetor é necessária para que se possa definir os níveis de iteração do método multi- p .

Os parâmetros escolhidos pelo usuário são o nível de erro especificado para o resíduo das iterações, o número de pré e pós-relaxações e o número máximo de iterações permitidas.

A idéia central na elaboração da classe **LinearMultigrid** é a definição de dois métodos principais: **FinetoCoarsePart** e **CoarsetoFinePart**. O primeiro deles perfaz a parte descendente dos ciclos multi- p na qual se executam ν_1 relaxações, calcula-se e transfere-se o resíduo sucessivamente do nível mais fino (i.e., maior p) para o mais grosso (i.e., menor p). Esse método resolve ainda, de forma direta, o sistema linear associado ao nível mais grosso. O segundo método é responsável pela parte ascendente dos ciclos na qual se prolonga o resíduo e se executam ν_2 relaxações até que se atinja o nível mais fino.

Observa-se que a diferença fundamental entre os métodos multi- p e os métodos multigrid clássicos é o fato dos diferentes níveis do processo serem percorridos em submatrizes da matriz de rigidez global de ordem polinomial mais elevada e não em diferentes matrizes de rigidez geradas a partir de uma série de malhas físicas associadas ao problema. Outro aspecto essencial que reflete a diferença entre esses métodos é o tipo de operadores de transferência empregados. No caso das estratégias multi- p , são utilizados operadores injetivos o que simplifica sobremaneira os procedimentos de restrição e prolongamento do resíduo.

Capítulo 6

Conclusões Gerais e Perspectivas Futuras

A partir do que foi exposto ao longo deste trabalho, é possível identificar os seguintes pontos fundamentais:

- Avaliação e obtenção de conjuntos de funções de forma hierárquicas definidos sobre elementos triangulares e tetraédricos para a aplicação em problemas lineares (equação de *Poisson* e elasticidade infinitesimal) e não-lineares (elasticidade finita) com malhas não-estruturadas. Este estudo resultou no desenvolvimento de novos conjuntos de funções hierárquicas com propriedades de ortogonalidade total entre seus modos internos;
- Seleção de conjuntos de funções de forma hierárquicas, com base nas propriedades de condicionamento e esparsidade locais, para aplicação em métodos multigrid algébricos associados à versão- p do MEF. Segundo o tipo de análise adotado, as funções sugeridas por (Webb e Abouchakra, 1995), no caso **2D**, e as de (Sherwin e Karniadakis, 1995), no caso **3D**, demonstraram-se as mais apropriadas;
- Aplicação da formulação p **3D** desenvolvida originalmente por (Karniadakis e Sherwin, 1999) para a aplicação em problemas de mecânica dos fluidos computacional com malhas não-estruturadas, na solução de problemas de elasticidade linear e não-linear. A formulação adotada demonstrou-se bastante flexível do ponto de vista das operações locais e eficiente em termos das soluções obtidas, sobretudo no caso não-linear. Observou-se que a precisão das

soluções obtidas ainda poderia ser melhorada se o elemento linear de partida utilizado fosse substituído por um elemento quadrático;

- Desenvolvimento e avaliação da performance computacional de métodos multigrid algébricos associados à versão- p do MEF para aplicação em problemas elípticos lineares (equação de *Poisson* 2D e elasticidade infinitesimal 3D) com malhas não-estruturadas. As análises conduzidas nos casos 2D e 3D revelaram que o método iterativo de gradientes conjugados pré-condicionado é claramente superior (em valores absolutos) às estratégias multi- p . Nesses casos, os resultados apontam na direção do uso de arquiteturas computacionais paralelas e na substituição das técnicas de relaxação de *Gauss-Seidel* por algoritmos iterativos mais eficientes. Especificamente no caso 2D, as investigações realizadas resultaram na construção de uma variante para o pré-condicionador modelo de (Babuška et al., 1991) que poderá se demonstrar viável em aplicações do tipo *hp*. Finalmente, observou-se que a utilização do procedimento de condensação estática dos modos internos é um aspecto fundamental na solução de sistemas algébricos lineares resultantes da aplicação da versão- p do MEF;
- Implementação, em linguagem C++, de um código de elementos finitos para a solução de problemas de elasticidade linear e não-linear 3D através da formulação p . Embora a arquitetura do código base disponível (desenvolvida para a versão- h do MEF) não fosse totalmente adequada a inclusão da formulação p , foi possível implementar o conjunto de classes necessárias ao cumprimento dos objetivos iniciais de forma minimamente integrada, eficiente e num prazo relativamente curto. Obviamente, é possível listar uma série de detalhes que poderiam enriquecer o desempenho da implementação realizada.

Tendo em vista os itens listados acima e as metas estabelecidas no Capítulo introdutório, pode-se dizer que o desenvolvimento deste trabalho alcançou pleno sucesso.

Como sugestões para a continuidade deste trabalho de tese, pode-se propor:

- Avaliar o desempenho da formulação p 3D implementada em problemas de elasticidade linear e não-linear ao se substituir o elemento tetraédrico linear do refinamento de partida por um elemento tetraédrico quadrático (e.g., elemento de *Lagrange* tipo-2);
- Utilizar as técnicas de soma fatorada (Karniadakis e Sherwin, 1999) para se manipular de forma mais eficiente as operações locais, sobretudo as de integração numérica;

- Implementar os procedimentos necessários ao tratamento de geometrias curvas bem como à aplicação de condições de contorno de *Dirichlet* não-homogêneas (i.e., imposição de deslocamentos);
- Estender as aplicações da formulação p 3D implementada para problemas com não-linearidades de material (e.g., hiperelasticidade finita);
- Desenvolver algoritmos para o pré-condicionamento dos sistemas algébricos resultantes da aplicação da formulação p 3D implementada com base nas técnicas de desacoplamento dos modos associados ao contorno dos elementos, como feito no caso dos algoritmos 2D;
- Testar outras técnicas de relaxação para os métodos multi- p que sejam mais eficazes que o método de *Gauss-Seidel* (Guo e Katz, 1998);
- Aplicar as estratégias multi- p em arquiteturas computacionais paralelas;
- Adequar a arquitetura do código atual para incorporar simultaneamente as abordagens h e p do MEF.

Referências Bibliográficas

- Abouchakra, R. Hierarchical tetrahedral elements using orthogonal polynomials. In: *1996 Canadian Conference on Electrical and Computer Engineering (CCECE'96)*, v. 18, p. 525–528, Calgary, part 2 (of 2), 1996.
- Adjerid, S., Aiffa, M., Flaherty, J. Hierarchical finite element bases for triangular and tetrahedral elements. *Comput. Methods Appl. Mech. Engrg.*, v. 190p.2925–2941, 2001.
- Apostol, T. *Calculus*, v. 2. John Wiley and Sons, New York, 1969.
- Axelsson, O., Barker, V. *Finite element solution of boundary value problems – theory and computation*. Academic Press, Orlando, 1984.
- Babuška, I. The p and h - p versions of the finite element method: the state of the art. In: Dwoyer, D. L., Hussaini, M. Y., Voigt, R. G., (Eds.), *Finite Elements: Theory and Applications*. Springer-Verlag, New York, 1988.
- Babuška, I., Craig, A., Mandel, J., Pitkaranta, J. Efficient preconditioning for the p -version finite element method in two dimensions. *SIAM J. Numer. Anal.*, v. 28, n.3, p.624–661, 1991.
- Babuška, I., Griebel, M., Pitkaranta, J. The problem of selecting the shape functions for p -type finite elements. *Int. J. Numer. Methods Engrg.*, v. 28p.1891–1908, 1989.
- Babuška, I., Guo, B. Q. The h - p version of the finite element method for domains with curved boundaries. *SIAM J. Numer. Anal.*, v. 25, n.4, p.837–861, 1988.
- Babuška, I., Guo, B. Q. The h - p version of the finite element method for problems with nonhomogeneous essential boundary condition. *Computer Methods in Applied Mechanics and Engineering*, v. 74p.1–28, 1989.

- Babuška, I., Helman, H. C. Some aspects of parallel implementation of the finite element method on message passing architectures. *J. Comput. Appl. Math.*, v. 27p.157–187, 1989.
- Babuška, I., Suri, M. The optimal convergence rate of the p -version of the finite element method. *SIAM J. Numer. Anal.*, v. 24p.750–776, 1987.
- Babuška, I., Suri, M. The p and h - p versions of the finite element method, an overview. *Compt. Methods Appl. Mech. Engrg.*, v. 80p.5–26, 1990.
- Babuška, I., Szabó, B. A. , Katz, I. N. The p -version of the finite element method. *SIAM J. Numer. Anal.*, v. 18, n.3, p.515–545, 1981.
- Banerjee, U., Suri, M. The effect of numerical quadrature in the p -version of the finite element method. *Mathematics of Computation*, v. 59, n.199, p.1–20, 1992.
- Bathe, K. *Finite Element Procedures*. Prentice Hall, Englewood Cliffs, 1996.
- Bernardi, C. Optimal finite-element interpolation on curved domains. *SIAM Journal in Numerical Analysis*, v. 26, n.5, p.1212–1240, 1989.
- Bittencourt, M. *Métodos multigrid e iterativos adaptáveis aplicados em malhas não-estruturadas*. Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, 1996., Tese (Doutorado).
- Bittencourt, M. Using C++ templates to develop finite element classes. *Engineering Computations*, v. 17, n.7, p.775–788, 2000.
- Bittencourt, M., Douglas, C. , Feijóo, R. Non-nested and non-structured multigrid methods applied to elastic problems. Part I: the two-dimensional case. Relatório de Pesquisa e Desenvolvimento 28/98, LNCC, Rio de Janeiro, 1998a.
- Bittencourt, M., Douglas, C. , Feijóo, R. Non-nested and non-structured multigrid methods applied to elastic problems. Part II: the three-dimensional case. Relatório de Pesquisa e Desenvolvimento 29/98, LNCC, Rio de Janeiro, 1998b.
- Bittencourt, M., Feijóo, R. Non-nested multigrid methods in finite element linear structural analysis. In: *Virtual Proceedings of the 8th Copper Mountain Conference on Multigrid Methods*, <http://www.mgnet.org>. The University of Colorado, 1997.

- Bittencourt, M., Feijóo, R. Object-oriented non-nested multigrid methods. In: I. Idelsohn, E. O., Dvorkin, E., (Eds.), *Computational Mechanics - New Trends and Applications*, p. 1–15, Barcelona. CIMNE, 1998.
- Booch, G., Rumbaugh, J. , Jacobson, I. *The Unified Modeling Language: User Guide*. Addison Wesley, Reading (MA), 1999.
- Bramble, J. H., Pasciak, J. E. , Schatz, A. E. The construction of preconditioners for elliptic problems by substructuring, I. *Math. Comput.*, v. 47p.103–134, 1986.
- Bramble, J. H., Pasciak, J. E. , Schatz, A. E. The construction of preconditioners for elliptic problems by substructuring, IV. *Math. Comput.*, v. 53p.1–24, 1989.
- Brandt, A. Multi-level adaptive solutions to boundary-value problems. *Mathematics of Computation*, v. 31, n.138, p.333–390, 1977.
- Briggs, W. *A multigrid tutorial*. SIAM Books, Philadelphia, 1987.
- Brussino, G., Herbin, R., Christidis, Z. , Sonnad, V. Parallel multilevel finite element method with hierarchical basis functions. Technical report, IBM Kingston, New York, 1989.
- Cao, W., Guo, B. A preconditioner with inexact element face solver for three dimensional p -version finite element methods. *Journal of Computational and Applied Mathematics*, v. 144p.131–144, 2002.
- Carnevali, P., Morris, R. B., Tsuji, Y. , Taylor, G. New basis functions and computational procedures for p -version finite element analysis. *Int. J. Numer. Methods Engrg.*, v. 36p.3759–3779, 1993.
- Casarin, M. A. Diagonal edge preconditioners in p -version and spectral element methods. *SIAM J. Sci. Comput.*, v. 18, n.2, p.610–620, 1997.
- Chen, Q., Babuška, I. Approximate optimal points for polynomial interpolation of real functions in an interval and in a triangle. *Comput. Methods Appl. Mech. Engrg.*, v. 128p.405–417, 1995.
- Chilton, L., Suri, M. Locking free mixed hp finite element methods for curvilinear domains. *Comput. Methods Appl. Mech. Engrg.*, v. 190p.3427–3442, 2001.

- Ciarlet, P., Raviart, P. Interpolation theory over curved elements with applications to finite element methods. *Computer Methods in Applied Mechanics and Engineering*, v. 1p.217–249, 1972.
- Coons, S. Surfaces for computer-aided design of space forms. Project MAC, Design Division, Department of Mechanical Engineering, MIT, Massachusetts - USA, 1964.
- Craig, A. W., Zienkiewicz, O. C. A multigrid algorithm using a hierarchical finite element basis. In: Paddon, D. J., Holstein, (Eds.), *Multigrid Methods for Integral and Differential Equations*, p. 310–312. Clarendon Press, Oxford, 1985.
- Devloo, P., Oden, J. , Pattani, P. Shape optimization of structures: A literature survey. *Comput. Meth. Appl. Mech. Engrg.*, v. 70p.203–235, 1988.
- Dey, S., Shephard, M. , Flaherty, J. Geometry representation issues associated with p -version finite element computations. *Comput. Meth. Appl. Mech. Engrg.*, v. 150p.39–55, 1997.
- Dubiner, M. Spectral methods on triangles and other domains. *SIAM J. Sci. Comput.*, v. 6p.345, 1991.
- Dunavant, D. A. High degree efficient symmetrical gaussian quadrature rules for the triangle. *Int. J. Numer. Methods Engrg.*, v. 21p.1129–1148, 1985.
- Duster, A., Rank, E. The p -version of the finite element method compared to an adaptive h -version for the deformation theory of plasticity. *Comput. Methods Appl. Mech. Engrg.*, v. 190p.1925–1935, 2001.
- Edgar, N. B., Surana, K. S. On the conditioning number and the selection criteria for p -version approximation functions. *Computers and Structures*, v. 60, n.4, p.521–530, 1996.
- Foresti, S., Brussino, G., Hassanzadeh, S. , Sonnad, V. Multilevel solution method for the p -version of finite elements. *Comput. Phys. Comm.*, v. 53p.349–355, 1989.
- Foresti, S., Hassanzadeh, S., Murakami, H. , Sonnad, V. A comparison of preconditioned iterative techniques using rapid operator application against direct solution methods. *Int. J. Numer. Methods Engrg.*, v. 32p.1137–1144, 1991.
- Golub, G., Loan, C. V. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, 1989.

- Golub, G., O'Leary, D. Some history of the conjugate gradient and lanczos methods. *SIAM Rev.*, v. 31p.50–102, 1989.
- Gordon, W. Blending-function methods of bivariate and multivariate interpolation and approximation. *SIAM J. Numer. Anal.*, v. 8, n.1, p.158–177, 1971.
- Gordon, W., Hall, C. Construction of curvilinear co-ordinate systems and applications to mesh generation. *Int. J. Numer. Meth. Engrg.*, v. 7p.461–477, 1973a.
- Gordon, W., Hall, C. Transfinite element methods: Blending-function interpolation over arbitrary curved element domains. *Numer. Math.*, v. 21p.109–129, 1973b.
- Guo, X.-Z., Katz, I. A parallel multi- p method. *Computers and Mathematics with Applications*, v. 39p.115–123, 2000.
- Guo, X.-Z., Katz, I. N. Performance enhancement of the multi- p preconditioner. *Computers and Mathematics with Applications*, v. 36p.1–8, 1998.
- Hackbush, W. *Multigrid Methods*. Springer-Verlag, Berlin, 1985.
- Hestenes, M., Stiefel, E. Methods of conjugate gradients for solving linear systems. *J. Res. Nat. Bur. Stand.*, v. 49p.409–436, 1952.
- Hesthaven, J. From electrostatics to almost optimal nodal sets for polynomial interpolation in a simplex. *SIAM Journal in Numerical Analysis*, v. 35, n.2, p.655–676, 1998.
- Holzer, S., Yosibash, Z. The p -version of the finite element method in incremental elasto-plastic analysis. *Computer Methods in Applied Mechanics and Engineering*, v. 39p.1859–1878, 1996.
- Hu, N., Guo, X.-Z., Katz, I. N. Multi- p preconditioners. *SIAM J. Sci. Comput.*, v. 18, n.6, p.1676–1697, 1997.
- Hu, N., Katz, I. N. Multi- p methods: iterative algorithms for the p -version of the finite element analysis. *SIAM J. Sci. Comput.*, v. 16, n.6, p.1308–1332, 1995.
- Hughes, T. J. R. *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Prentice-Hall, New York, 1987.
- Jinyun, Y. Symmetric gaussian quadrature formulae for tetrahedronal regions. *Comp. Methods Appl. Mech. Engrg.*, v. 43p.349–353, 1984.

- Johnson, C. *Numerical Solution of Partial Differential Equations by the Finite Element Method*. Cambridge University Press, Cambridge, 1990.
- Kapotin, I. E., Kolotilina, L. Y. , Yeregin, A. Y. Block SSOR preconditionings for high order 3D FE systems, II, incomplete BSSOR preconditionings. *Linear Algebra Appl.*, v. 154-156p.647-674, 1991.
- Karniadakis, G. E., Sherwin, S. J. *Spectral/hp Element Methods for CFD*. Oxford University Press, Oxford, 1999.
- Katz, I. Nodal variables for complete conforming finite elements of arbitrary polynomial order. *Computers and Mathematics with Applications*, v. 4p.85-112, 1978.
- Kim, C., Suri, M. On the p -version of the finite element method in the presence of numerical integration. *Numerical Methods for Partial Differential Equations*, v. 9p.593-629, 1993.
- Korneev, V. G., Jensen, S. Preconditioning of the p -version of the finite element method. *Comp. Methods Appl. Mech. Engrg.*, v. 150p.215-238, 1997.
- Korneev, V. G., Jensen, S. Domain decomposition preconditioning in the hierarchical p -version of the finite element method. *Applied Numerical Mathematics*, v. 29p.479-518, 1999.
- Krause, R., Mucke, R. , Rank, E. hp -version finite elements for geometrically non-linear problems. *Communications in Numerical Methods and Engineering*, v. 11p.887-897, 1995.
- Lenoir, M. Optimal isoparametric finite elements and error estimates for domains involving curved boundaries. *SIAM Journal on Numerical Analysis*, v. 3p.562-580, 1986.
- Maitre, J., Pourquier, O. About the conditioning of matrices in the p -version of the finite element method for second order elliptic problems. *Journal of Computational and Applied Mathematics*, v. 63p.341-348, 1995.
- Mandel, J. Hierarchical preconditioning and partial orthogonalization for the p -version finite element method. In: Chan, T., Glowinski, R. , Windlund, O., (Eds.), *Proceedings of the 3th International Symposium on Domain Decomposition Methods for Partial Differential Equations*, p. 141-156, Huston, Texas. SIAM, 1990a.
- Mandel, J. Iterative solvers by substructuring for the p -version finite element method. *Comp. Methods Appl. Mech. Engrg.*, v. 80p.117-128, 1990b.

- Mandel, J. On block diagonal and schur complement preconditioning. *Numer. Math.*, v. 58p.79–93, 1990c.
- Mandel, J. Two-level domain decomposition preconditioning for the p -version finite element method in three dimensions. *Int. J. Numer. Methods Engrg.*, v. 29p.1095–1108, 1990d.
- Mandel, J. An iterative solver for the p -version finite elements in three dimensions. *Comp. Methods Appl. Mech. Engrg.*, v. 116p.175–183, 1994.
- McCormick, S. *Multigrid methods*, v. 3 de *Frontiers Series*. SIAM Books, Philadelphia, 1987.
- Nie, Q., Niu, Q. p -Version large strain finite element formulation and application in elastic-plastic deformation. *Computers and Structures*, v. 65, n.5, p.761–765, 1997.
- Nogueira Jr., A. C. *Desenvolvimento de um elemento finito do tipo hierárquico para análise estática de placas e cascas a partir do elemento superparamétrico quadrilateral linear*. Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, 1998., Dissertao (Mestrado).
- Nogueira Jr., A. C., Bittencourt, M. Estudo comparativo entre funções de base hierárquicas para a versão- p do método de elementos finitos. In: *Proceedings of the 15th Brazilian Congress of Mechanical Engineering (COBEM 1999)*, p. 1–10, Águas de Lindóia - SP - Brazil. State University of Campinas, 1999.
- Nogueira Jr., A. C., Bittencourt, M. Algebraic and adaptive multigrid schemes for the p and hp versions of the finite element method. In: *Proceedings of the 22th Iberian Latin-American Congress on Computational Methods in Engineering (CILAMCE 2001)*, Campinas - SP - Brazil. State University of Campinas, 2001a.
- Nogueira Jr., A. C., Bittencourt, M. Hierarchical basis functions for the p -version of the finite element method (in portuguese). *Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería*, v. 17, n.1, p.37–59, 2001b.
- Nogueira Jr., A. C., Bittencourt, M. Algebraic multigrid schemes for the p -version of the finite element method applied to 3D elliptical problems. In: *Proceedings of the 5th World Congress on Computational Mechanics (WCCM V)*, Vienna - Austria. Technical University of Vienna and IACM, 2002.

- Novotny, A. A., Pereira, J. T. , Fancello, E. A. A fast hp adaptive finite element mesh design for 2D elliptical boundary value problems. Submitted for publication, 1999.
- Oden, J., Carey, G. , Becker, E. *Finite Elements An Introduction*, v. 1 de *Texas Finite Element Series*. Englewood Cliffs, New Jersey, Prentice-Hall, Inc., USA, 1981.
- Paczelt, I., Szabó, B. , Szabó, T. Solution of contact problem using the hp -version of the finite element method. *Computers and Mathematics with Applications*, v. 38p.49–69, 1999.
- Patera, A. A spectral method for fluid dynamics: Laminar flow in a channel expansion. *Journal of Computational Physics*, v. 54p.468, 1984.
- Pavarino, L. Additive schwarz methods for the p -version finite element method. *Numer. Math.*, v. 66p.493–515, 1994a.
- Pavarino, L. Schwarz methods with local refinement for the p -version finite element method. *Numer. Math.*, v. 69p.185–211, 1994b.
- Pavarino, L., Widlund, O. A polylogarithmic bound for an iterative substructuring method for spectral elements in three dimensions. *SIAM Journal in Numerical Analysis*, v. 33p.1303–1355, 1996.
- Pavarino, L., Widlund, O. Iterative substructuring methods for spectral elements: problems in three dimensions based on numerical quadrature. *Comput. Math. Appl.*, v. 33p.193–209, 1997.
- Peano, A. *Hierarchies of conforming finite elements*. Washington University - St. Louis, 1975., Dissertao (Mestrado).
- Peano, A. Hierarchies of conforming finite elements for plane elasticity and plate bending. *Computers and Mathematics with Applications*, v. 2p.211–224, 1976.
- Rossow, M., Katz, I. Hierarchical finite elements and precomputed arrays. *Int. J. Numer. Methods Eng.*, v. 12p.977–999, 1993.
- Sherwin, S., Casarin, M. Low energy basis preconditioning for elliptic substructured solvers based on unstructured spectral/ hp element discretizations. *Journal of Computational Physics*, v. 171p.1–24, 2001.

- Sherwin, S., Karniadakis, G. A new triangular and tetrahedral basis for high-order (*hp*) finite element methods. *Int. J. Numer. Methods Engrg.*, v. 38p.3775–3802, 1995.
- Sherwin, S., Karniadakis, G. Tetrahedral *hp* finite elements: Algorithms and flow simulations. *Journal of Computational Physics*, v. 124p.14–45, 1996.
- Silva, C. *Análise de Sensibilidade, Otimização e Orientação por Objetos em Hiperelasticidade Não-Linear*. Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, 2002., Tese (Doutorado).
- Szabó, B. A., Actis, R. , Holzer, S. Solution of elastic-plastic stress analysis problems by the *p*-version of the finite element method. In: Babuška, I., et al., J. F., (Eds.), *IMA Volumes in Mathematics and its Applications*, v. 75, p. 395–416. Springer Verlag, New York, 1995.
- Szabó, B. A., Babuška, I. *Finite Element Analysis*. Wiley Interscience, New York, 1991.
- Szabó, B. A., Peano, A. G. Hierarchic finite elements. In: Kardestuncer, H., Norrie, D., (Eds.), *Finite Element Handbook*, p. 2227–2233. McGraw-Hill, New York, 1987.
- Timoshenko, S., Goodier, J. *Theory of Elasticity*. McGraw-Hill, 3rd ed., 1970.
- Vijayakar, S., Busby, H. , Houser, D. Finite element analysis of quasi-prismatic bodies using Chebyshev polynomials. *Int. J. Num. Meth. Engrg.*, v. 24p.1461–1477, 1987.
- Volpert, Y., Szabó, T., Paczelt, I. , Szabó, B. Application of the space enrichment method to problems of mechanical contact. *Finite Elements in Analysis and Design*, v. 24p.157–170, 1997.
- Warburton, T. *Spectral/*hp* Methods on Polymorphic Multi-Domains: Algorithms and Applications*. Brown University, 1998., Tese (Doutorado).
- Webb, J. P., Abouchakra, R. Hierarchal triangular elements using orthogonal polynomials. *Int. J. Numer. Methods Engrg.*, v. 38p.245–257, 1995.
- Zienkiewicz, O., Craig, A. Adaptive refinement, error estimates, multigrid solution and hierarchic finite element method concepts. In: Babuška, I., Zienkiewicz, O., Gago, J. , Oliveira, E., (Eds.), *Accuracy Estimates and Adaptive Refinements in Finite Element Computations*, Cap. 2, p. 25–59. John Wiley & Sons, 1986.

- Zienkiewicz, O., de S.R. Gago, J. , Kelly, D. The hierarchical concept in the finite element analysis. *Computers and Structures*, v. 16p.53–65, 1983.
- Zienkiewicz, O., Irons, B., Scott, F. , Campbell, J. Tree-dimensional stress analysis. In: *Symposium on High Speed Computing of Elastic Structures (IUTAM)*, p. 413–431, Liege - France, 1970.
- Zienkiewicz, O., Kelly, D., de S.R. Gago, J. , Babuška, I. Hierarchical finite element approaches, error estimates and adaptive refinement. In: *MAFELAP*, p. 313–346, Brunel University, 1981.
- Zienkiewicz, O., Taylor, R. *The Finite Element Method*. McGraw Hill, New York, 1989.
- Zienkiewicz, O., Zhu, J. , Gong, N. Effective and practical h - p version adaptive analysis procedures for the finite element method. *International Journal for Numerical Methods in Engineering*, v. 28p.879–891, 1989.
- Zumbusch, G. *Simultaneous h - p adaption in multilevel finite elements*. Freien Universitat Berlin, 1995., Tese (Doutorado), Germany.

Apêndice A

Polinômios Ortogonais

A.1 Polinômios de Legendre

Os polinômios de *Legendre* são caracterizados por satisfazerem a equação diferencial linear de *Legendre*, para $n = 1, 2, 3, \dots$

$$(1 - x^2)y'' - 2xy' + n(n + 1)y = 0, \quad -1 \leq x \leq 1. \quad (\text{A.1})$$

Esses polinômios são usualmente denotados por $P_n(x)$, sendo n o seu grau. A *fórmula recursiva de Bonnet* fornece um meio simples e barato de se obter esse tipo de polinômio

$$(n + 1)P_{n+1}(x) = (2n + 1)xP_n(x) - nP_{n-1}(x). \quad (\text{A.2})$$

As derivadas dos polinômios de *Legendre* também satisfazem uma relação recursiva, para $n = 1, 2, 3, \dots$, dada por

$$(2n + 1)P_n(x) = P'_{n+1}(x) - P'_{n-1}(x). \quad (\text{A.3})$$

Uma propriedade notável desses polinômios é a sua relação de ortogonalidade em termos do produto interno dado por

$$\int_{-1}^{+1} P_i(x)P_j(x)dx = \begin{cases} \frac{2}{2i+1} & \text{para } i = j \\ 0 & \text{para } i \neq j \end{cases}. \quad (\text{A.4})$$

A normalização dos polinômios $P_i(x)$ e $P_j(x)$ na relação de ortogonalidade acima resulta em

$$\int_{-1}^1 P_i(x)P_j(x)dx = \delta_{ij}, \quad (\text{A.5})$$

sendo δ_{ij} o delta de *Kronecker*.

Outra característica relevante desse tipo polinômio é fato de todas as suas raízes pertencerem ao intervalo $-1 < x < 1$.

A.2 Polinômios de Jacobi

Os polinômios de *Jacobi* de peso misto $P_n^{\alpha,\beta}(x)$, sendo n o seu grau, representam uma família de soluções polinomiais do problema de *Sturm-Liouville* singular descrito pela seguinte equação diferencial linear

$$(1-x^2)y'' + [\beta - \alpha - (\alpha + \beta + 2)x]y' + n(n + \alpha + \beta + 1)y = 0, \quad -1 \leq x \leq 1, \quad (\text{A.6})$$

sendo $\alpha > -1, \beta > -1$ e $n = 1, 2, 3, \dots$

Esses polinômios podem ser gerados a partir da seguinte relação de recorrência

$$P_n^{\alpha,\beta}(x) = \frac{cx + e}{b}P_{n-1}^{\alpha,\beta}(x) - \frac{f}{b}P_{n-2}^{\alpha,\beta}(x), \quad (\text{A.7})$$

com $P_{-1}^{\alpha,\beta}(x) = 0, P_0^{\alpha,\beta}(x) = 1$ e os coeficientes b, c, e, f , calculados da seguinte maneira

$$\begin{aligned} u &= n - 1; g = \alpha + \beta + n; s = g + u; h = s + 1; t = s - 1; \\ b &= 2ngt; c = hst; e = s(\alpha^2 - \beta^2); f = 2(\alpha + u)(\beta + u)h. \end{aligned} \quad (\text{A.8})$$

Para n, m, α, β inteiros, os polinômios de *Jacobi* possuem à seguinte propriedade de ortogonalidade com respeito à função peso $(1-x)^\alpha(1+x)^\beta$

$$\int_{-1}^1 P_l^{\alpha,\beta}(x)P_m^{\alpha,\beta}(x)(1-x)^\alpha(1+x)^\beta dx = \begin{cases} \frac{2^{\alpha+\beta+m}}{2m+\alpha+\beta+1} \frac{(m+\alpha)!(m+\beta)!}{m!(m+\alpha+\beta)!} & \text{para } l = m \\ 0 & \text{para } l \neq m \end{cases} \quad (\text{A.9})$$

A normalização de $P_l^{\alpha,\beta}(x)$ e $P_m^{\alpha,\beta}(x)$ produz a relação de ortogonalidade dada por

$$\int_{-1}^1 P_l^{\alpha,\beta}(x)P_m^{\alpha,\beta}(x)(1-x)^\alpha(1+x)^\beta dx = \delta_{lm}, \quad (\text{A.10})$$

sendo δ_{lm} o delta de *Kronecker*.

Observa-se que os polinômios de *Legendre* são um caso particular dos polinômios de *Jacobi* quando $\alpha = \beta = 0$, i.e., $P_n^{0,0}(x) = P_n(x)$.

Apêndice B

Integração Numérica 1D do Tipo *Gauss-Jacobi*

B.1 Elementos Básicos das Regras de Integração Numéricas¹

A integração de uma função $u(x)$ num intervalo $[-1, 1]$ com respeito à função peso $(1-x)^\alpha(1+x)^\beta$ (com $\alpha > -1$ e $\beta > -1$) pode ser numericamente avaliada usando-se o método de *Gauss* o qual é descrito por uma soma discreta da seguinte forma:

$$\int_{-1}^1 (1-x)^\alpha(1+x)^\beta u(x) dx = \sum_{i=0}^{Q-1} w_i^{\alpha,\beta} u(x_i) + R(u), \quad (\text{B.1})$$

sendo Q o número de pontos de integração, x_i e $w_i^{\alpha,\beta}$, as coordenadas dos pontos e os pesos da regra de integração e $R(u)$, o resíduo da aproximação numérica de integração. Esse tipo de regra é elaborado de modo que $R(u) = 0$ se $u(x) \in \mathcal{P}_{2Q-k}([-1, 1])$, com \mathcal{P}_{2Q-k} sendo o espaço de polinômios unidimensionais de ordem $2Q - k$. Nesse caso, k é determinado pelo tipo de distribuição de pontos de quadratura escolhido podendo assumir os valores $k = 1$ (a distribuição clássica de *Gauss*), $k = 2$ (a distribuição de *Gauss-Radau*) e $k = 3$ (a distribuição de *Gauss-Lobatto*).

¹(Karniadakis e Sherwin, 1999)

A derivação da expressão (B.1) considera o caso em que $u(x)$ é um polinômio de ordem $2Q - k$ decomposto no seguinte formato:

$$u(x) = \sum_{i=0}^{Q-1} u(x_i)h_i(x) + s(x)r(x), \quad (\text{B.2})$$

sendo

$$\begin{aligned} h_i(x_j) &= \delta_{ij}, \quad h(x) \in \mathcal{P}_{Q-1}, \\ s(x_i) &= 0, \quad s(x) \in \mathcal{P}_Q, \\ r(x) &\in \mathcal{P}_{Q-k} \end{aligned} \quad (\text{B.3})$$

e δ_{ij} , o delta de *Kronecker*. Observa-se que a função $u(x)$ foi decomposta numa soma ponderada de polinômios $h_i(x)$ adicionada ao produto dos polinômios $s(x)$ e $r(x)$. Nesse caso, $h_i(x)$ são exatamente os polinômios de *Lagrange* definidos nas Q coordenadas x_i , $s(x)$ é um polinômio de ordem Q com raízes nas coordenadas x_i e $r(x)$ é um polinômio de ordem $Q - k$ geralmente denominado resto.

Integrando-se a expressão (B.2) com respeito à função peso $(1 - x)^\alpha(1 + x)^\beta$ no intervalo $[-1, 1]$ e comparando o resultado com (B.1), chega-se a

$$w_i^{\alpha,\beta} = \int_{-1}^1 (1 - x)^\alpha(1 + x)^\beta h_i(x) dx, \quad (\text{B.4})$$

$$R(u) = \int_{-1}^1 (1 - x)^\alpha(1 + x)^\beta s(x)r(x) dx. \quad (\text{B.5})$$

Se $s(x)$ for especificado como um polinômio conhecido então as coordenadas dos pontos de integração x_i ficarão determinadas pelas raízes de $s(x)$ uma vez que $s(x_i) = 0$. A escolha adequada de $s(x)$ pode produzir um resíduo nulo, i.e., $R(u) = 0$. Por exemplo, se $s(x)$ for escolhido como sendo o polinômio de *Jacobi* de pesos α, β e ordem Q (vide seção A.2 do Apêndice A), ou seja, $s(x) = P_Q^{\alpha,\beta}(x)$, então

$$R(u) = \int_{-1}^1 (1 - x)^\alpha(1 + x)^\beta P_Q^{\alpha,\beta}(x)r(x) dx. \quad (\text{B.6})$$

Observa-se, a partir das relações de ortogonalidade apresentadas na seção A.2, que, se $k = 1$, $r(x)$ é um polinômio de ordem $Q - 1$ e portanto $R(u) = 0$. Essa escolha determina a distribuição clássica de *Gauss-Jacobi* para os pontos de quadratura. Desse modo, as coordenadas dos pontos de integração serão dadas por $x_i = x_{i,Q}^{\alpha,\beta}$, sendo $x_{i,Q}^{\alpha,\beta}$ as raízes do polinômio de *Jacobi* $P_Q^{\alpha,\beta}(x)$ (com

$i = 0, \dots, Q - 1$). Nota-se ainda que

$$x_{0,Q}^{\alpha,\beta} < x_{1,Q}^{\alpha,\beta} < \dots < x_{Q-1,Q}^{\alpha,\beta}. \quad (\text{B.7})$$

As distribuições de pontos de integração de *Radau* e *Lobatto* demandam a inclusão de um ou ambos os extremos do intervalo $[-1, 1]$, respectivamente. Essa exigência evidentemente modifica a forma de $s(x)$ o que por sua vez afeta os valores de α e β na função peso $(1-x)^\alpha(1+x)^\beta$. Assim, as coordenadas dos pontos de integração para esses tipos de distribuição de pontos são determinadas por polinômios de *Jacobi* com valores de α e β diferentes daqueles obtidos para a regra clássica de *Gauss-Jacobi*.

B.2 Regras do Tipo *Gauss-Jacobi*

Esta seção apresenta as expressões que fornecem os valores das coordenadas dos pontos e dos pesos de integração segundo as distribuições de pontos de *Gauss-Jacobi*, *Gauss-Jacobi-Radau* e *Gauss-Jacobi-Lobatto*. O caso particular da distribuição clássica de *Gauss-Legendre* também pode ser obtido levando-se em conta apenas que, nessa situação, $\alpha = \beta = 0$. Em qualquer um dos casos mencionados acima, as coordenadas dos pontos de integração são determinadas em termos das raízes $x_{i,m}^{\alpha,\beta}$ dos polinômios de *Jacobi* de ordem m . As expressões para os pesos das regras de integração foram obtidas a partir da referência (Karniadakis e Sherwin, 1999).

B.2.1 Regra de *Gauss-Jacobi*

Para esse tipo de quadratura não há restrições impostas às coordenadas dos pontos de integração de modo que esses pontos são obtidos a partir das raízes do polinômio $P_Q^{\alpha,\beta}(x)$. As expressões para os pontos e pesos da regra de *Gauss-Jacobi*, considerando-se apenas valores inteiros de α e β (com $\alpha, \beta > -1$), são dadas por

$$x_i = x_{i,Q}^{\alpha,\beta}, \quad i = 0, \dots, Q - 1, \quad (\text{B.8})$$

$$w_i^{\alpha,\beta} = H_{i,Q}^{\alpha,\beta}, \quad i = 0, \dots, Q - 1, \quad (\text{B.9})$$

$$R(u) = 0 \quad \text{se} \quad u(x) \in \mathcal{P}_{2Q-1}([-1, 1]), \quad (\text{B.10})$$

com

$$H_{i,Q}^{\alpha,\beta} = \frac{2^{\alpha+\beta+1}(\alpha+Q)!(\beta+Q)!}{Q!(\alpha+\beta+Q)! [1-(x_i)^2]} \left[\frac{d}{dx} (P_Q^{\alpha,\beta}(x)) \Big|_{x=x_i} \right]^{-2}. \quad (\text{B.11})$$

B.2.2 Regra de Gauss-Jacobi-Radau

A regra de *Gauss-Jacobi-Radau* requer que se inclua um dos extremos do intervalo $[-1, 1]$ como ponto de integração. Se for escolhida a coordenada $x = -1$, então $s(x)$ em (B.2) terá a seguinte forma:

$$s(x) = (1+x)s_1(x),$$

com

$$s_1(x) \in \mathcal{P}_{Q-1}([-1, 1]).$$

Assim, a expressão para o resíduo $R(u)$ fornecida em (B.4) será dada por

$$R(u) = \int_{-1}^1 (1-x)^\alpha (1+x)^{\beta+1} s_1(x) r(x) dx.$$

Se $s_1(x) = P_{Q-1}^{\alpha,\beta+1}(x)$, então $R(u) = 0$ quando $k = 2$ pois $r(x) \in \mathcal{P}_{Q-2}([-1, 1])$.

As expressões para os pontos e pesos da regra de *Gauss-Jacobi-Radau*, considerando-se apenas valores inteiros de α e β (com $\alpha, \beta > -1$), são dadas por

$$x_i = \begin{cases} -1, & i = 0 \\ x_{i-1, Q-1}^{\alpha, \beta+1}, & i = 1, \dots, Q-1 \end{cases}, \quad (\text{B.12})$$

$$w_i^{\alpha, \beta} = \begin{cases} (\beta+1)B_{0, Q-1}^{\alpha, \beta}, & i = 0 \\ B_{i, Q-1}^{\alpha, \beta}, & i = 1, \dots, Q-1 \end{cases}, \quad (\text{B.13})$$

$$R(u) = 0 \quad \text{se } u(x) \in \mathcal{P}_{2Q-2}([-1, 1]), \quad (\text{B.14})$$

com

$$B_{i, Q-1}^{\alpha, \beta} = \frac{2^{\alpha+\beta}(\alpha+Q-1)!(\beta+Q-1)!(1-x_i)}{(Q-1)!(\beta+Q)(\alpha+\beta+Q)! [P_{Q-1}^{\alpha, \beta}(x_i)]^2}. \quad (\text{B.15})$$

B.2.3 Regra de *Gauss-Jacobi-Lobatto*

A regra de *Gauss-Jacobi-Lobatto* requer que se incluam ambos os extremos do intervalo $[-1, 1]$ como pontos de integração. Nesse caso, $s(x)$ em (B.2) terá a seguinte forma:

$$s(x) = (1 - x)(1 + x)s_2(x),$$

com

$$s_2(x) \in \mathcal{P}_{Q-2}([-1, 1]).$$

Desse modo, a expressão para o resíduo $R(u)$ fornecida em (B.4) será dada por

$$R(u) = \int_{-1}^1 (1 - x)^{\alpha+1}(1 + x)^{\beta+1}s_2(x)r(x)dx.$$

Se $s_2(x) = P_{Q-2}^{\alpha+1, \beta+1}(x)$, então $R(u) = 0$ quando $k = 3$ pois $r(x) \in \mathcal{P}_{Q-3}([-1, 1])$.

As expressões para os pontos e pesos da regra de *Gauss-Jacobi-Lobatto*, considerando-se apenas valores inteiros de α e β (com $\alpha, \beta > -1$), são dadas por

$$x_i = \begin{cases} -1, & i = 0 \\ x_{i-1, Q-2}^{\alpha+1, \beta+1}, & i = 1, \dots, Q-2 \\ 1, & i = Q-1 \end{cases}, \quad (\text{B.16})$$

$$w_i^{\alpha, \beta} = \begin{cases} (\beta + 1)C_{0, Q-2}^{\alpha, \beta}, & i = 0 \\ C_{i, Q-2}^{\alpha, \beta}, & i = 1, \dots, Q-2 \\ (\alpha + 1)C_{Q-1, Q-2}^{\alpha, \beta}, & i = Q-1 \end{cases}, \quad (\text{B.17})$$

$$R(u) = 0 \text{ se } u(x) \in \mathcal{P}_{2Q-3}([-1, 1]), \quad (\text{B.18})$$

com

$$C_{i, Q-2}^{\alpha, \beta} = \frac{2^{\alpha+\beta+1}(\alpha + Q - 1)!(\beta + Q - 1)!}{(Q - 1)(Q - 1)!(\alpha + \beta + Q)! [P_{Q-1}^{\alpha, \beta}(x_i)]^2}. \quad (\text{B.19})$$

B.3 Avaliação das Raízes dos Polinômios de *Jacobi*

Na seção precedente, observou-se que os pesos das diferentes regras de quadratura do tipo *Gauss-Jacobi* possuem expressões matemáticas fechadas em termos das coordenadas x_i . Entretanto, o mesmo não ocorre no caso das próprias coordenadas dos pontos de integração. Essas coordenadas

são definidas a partir das raízes dos polinômios de *Jacobi* de tal modo que

$$\begin{aligned} x_i &= x_{i,m}^{\alpha,\beta}, \\ P_m^{\alpha,\beta}(x_{i,m}^{\alpha,\beta}) &= 0, \quad i = 0, 1, \dots, m-1. \end{aligned}$$

Nesse caso, as raízes $x_{i,m}^{\alpha,\beta}$ podem ser avaliadas numericamente usando-se um algoritmo iterativo padrão como o de *Newton-Raphson*. Esse algoritmo é basicamente descrito pela relação recursiva

$$x_i^{k+1} = x_i^k - \frac{f(x_i^k)}{f'(x_i^k)}, \quad k = 0, 1, \dots \quad \text{até que} \quad |x_i^{k+1} - x_i^k| < \varepsilon \quad \text{com} \quad \varepsilon > 0, \quad (\text{B.20})$$

sendo x_i^{k+1} e x_i^k as aproximações para a raiz de $f(x)$ nos passos $n+1$ e n , respectivamente, e $f(x) = P_m^{\alpha,\beta}(x)$.

Como, em geral, $f(x)$ possui várias raízes ($P_m^{\alpha,\beta}$ tem m raízes reais), é necessário aplicar o procedimento de *deflação polinomial* sobre o algoritmo iterativo padrão. Esse procedimento consiste em fatorar as raízes do polinômio original que já foram determinadas pelo algoritmo. Isso significa que a função $f(x)$ em (B.20) deve ser escrita como

$$f_{m-n}(x) = \frac{P_m^{\alpha,\beta}(x)}{\prod_{i=0}^{n-1} (x - x_i)} \quad (\text{B.21})$$

sendo x_i ($i = 0, \dots, n-1$) as raízes conhecidas de $P_m^{\alpha,\beta}(x)$.

Uma manipulação algébrica sobre $f_{m-n}(x)$ e sua derivada primeira fornece a seguinte relação para o algoritmo iterativo:

$$\frac{f_{m-n}(x)}{f'_{m-n}(x)} = \frac{P_m^{\alpha,\beta}(x)}{[P_m^{\alpha,\beta}(x)]' - P_m^{\alpha,\beta}(x) \sum_{i=0}^{n-1} [1/(x - x_i)]}. \quad (\text{B.22})$$

A aproximação inicial x_i^0 utilizada pelo algoritmo de *Newton-Raphson* com *deflação polinomial* pode ser obtida a partir das raízes do polinômio de *Chebyshev* as quais são escritas de forma explícita como

$$x_{i,m}^{-\frac{1}{2},-\frac{1}{2}} = -\cos\left(\frac{2i+1}{2m}\pi\right), \quad i = 0, 1, \dots, m-1. \quad (\text{B.23})$$

Do ponto de vista numérico, é possível verificar (Karniadakis e Sherwin, 1999) que uma aproximação inicial dada pela média entre a i -ésima raiz do polinômio de *Chebyshev* $x_{i,m}^{-\frac{1}{2},-\frac{1}{2}}$ e a raiz conhecida de $P_m^{\alpha,\beta}(x)$ obtida no passo anterior x_{k-1} incrementa a eficiência do algoritmo

iterativo com deflação.

A partir das relações (B.22) e (B.23) torna-se simples elaborar um programa computacional para se determinar as raízes do polinômio $P_m^{\alpha,\beta}(x)$. A seguir, apresenta-se um algoritmo de *Newton-Raphson* com *deflação polinomial* escrito a partir do programa de uso comercial MATLAB 5.3.

```

%-----%
% Algoritmo de Newton-Raphson com processo de deflacao para localizar
% raizes de polinomios de Jacobi de pesos alfa,beta e ordem m

function [x] = jacobi_root(alfa,beta,m)

x = zeros(m,1);
epsilon = 1e-16;

for k = 0:(m-1)

    r = -cos(pi * (2*k + 1)/(2*m));

    if k > 0
        r = (r + x(k))/2;
    end

    delta = 1;
    while abs(delta) > epsilon,

        s = 0;
        for i = 0:(k-1)
            s = s + 1/(r - x(i+1));
        end

        Pm = jacobi_polynomials(r,m,alfa,beta);
        DPm = D_jacobi_polynomials(r,m,alfa,beta);
        delta = -Pm/(DPm - Pm * s);
        r = r + delta;

    end

    x(k+1) = r;

end

%-----%
% Subrotina para calculo dos polinomios de Jacobi

function [Pm] = jacobi_polynomials(r,m,alfa,beta)

% Polinomios de Jacobi calculados no ponto r

Pn = 1;
Pn1 = 0.5*(alfa - beta + (alfa + beta + 2) * r);

```

```

for n = 1:(m-1)

    a1n = 2*(n+1)*(n + alfa + beta +1)*(2*n + alfa + beta);
    a2n = (2*n + alfa + beta + 1)*(alfa^2 - beta^2);
    a3n = (2*n + alfa + beta)*(2*n + alfa + beta + 1)*(2*n + alfa + beta + 2);
    a4n = 2*(n + alfa)*(n + beta)*(2*n + alfa + beta + 2);

    Pn2 = (1/a1n)*((a2n + a3n * r)*Pn1 - a4n*Pn);

    Pn = Pn1;
    Pn1 = Pn2;

end

if m > 1
    Pm = Pn2;
elseif m == 1
    Pm = Pn1;
else
    Pm = Pn;
end

%-----%
% Subrotina para calculo das derivadas de primeira ordem dos polinomios
% de Jacobi

function [DPm] = D_jacobi_polynomials(r,m,alfa,beta)

% Derivadas dos polinomios de Jacobi calculados no ponto r

Pn = 1;
Pn1 = 0.5*(alfa - beta + (alfa + beta + 2) * r);

for n = 1:m

    a1n = 2*(n+1)*(n + alfa + beta +1)*(2*n + alfa + beta);
    a2n = (2*n + alfa + beta + 1)*(alfa^2 - beta^2);
    a3n = (2*n + alfa + beta)*(2*n + alfa + beta + 1)*(2*n + alfa + beta + 2);
    a4n = 2*(n + alfa)*(n + beta)*(2*n + alfa + beta + 2);

    Pn2 = (1/a1n)*((a2n + a3n * r)*Pn1 - a4n*Pn);

    b1n = (2*n + alfa + beta)*(1 - r^2);
    b2n = (n)*(alfa - beta - (2*n + alfa + beta) * r);
    b3n = 2*(n + alfa)*(n + beta);

    DPn1 = (1/b1n)*(b2n*Pn1 + b3n*Pn);

    Pn = Pn1;
    Pn1 = Pn2;

end
DPm = DPn1;

```

Apêndice C

Discretização da Formulação Variacional para Elasticidade Finita

C.1 Discretização para Não-Linearidades Geométricas

Utilizando a notação sugerida por (Silva, 2002), é possível escrever, a partir da formulação variacional de problemas de elasticidade finita

$$a(\mathbf{x}, \delta \mathbf{x}) = \int_{\Omega} \mathbf{S} \cdot \delta \mathbf{E} \, dV = \int_{\Omega} \mathbf{S}(\mathbf{x}) \cdot D\mathbf{E}[\delta \mathbf{x}] \, dV. \quad (\text{C.1})$$

A derivação dessa expressão, no sentido de *Gateaux*, fornece

$$\begin{aligned} Da(\mathbf{x}, \delta \mathbf{x})[\Delta \mathbf{u}] &= \delta a(\mathbf{x}; \Delta \mathbf{u}, \delta \mathbf{x}) = \int_{\Omega} D(\mathbf{S} \cdot \delta \mathbf{E})[\Delta \mathbf{u}] \, dV = \\ &= \int_{\Omega} \mathbf{C}(\mathbf{x}) : D\mathbf{E}[\Delta \mathbf{u}] \cdot \delta \mathbf{E} \, dV + \int_{\Omega} \mathbf{S} \cdot D\delta \mathbf{E}[\Delta \mathbf{u}] \, dV, \end{aligned} \quad (\text{C.2})$$

sendo

$$\mathbf{E} = \frac{1}{2}(\nabla \mathbf{u} + \nabla \mathbf{u}^T + \nabla \mathbf{u}^T \nabla \mathbf{u}) = \frac{1}{2}(\mathbf{F}^T \mathbf{F} - \mathbf{I}),$$

$$\mathbf{F} = \mathbf{I} + \nabla \mathbf{u},$$

$$D\mathbf{E}[\delta \mathbf{x}] = \delta \mathbf{E} = \frac{1}{2}(\mathbf{F}^T D\mathbf{F}[\delta \mathbf{x}] + D\mathbf{F}^T[\delta \mathbf{x}] \mathbf{F}) = \frac{1}{2}(\mathbf{F}^T \nabla \delta \mathbf{x} + \nabla \delta \mathbf{x}^T \mathbf{F}) = \frac{1}{2}(\mathbf{F}^T \delta \mathbf{F} + \delta \mathbf{F}^T \mathbf{F}),$$

$$D\delta \mathbf{E}[\Delta \mathbf{u}] = \frac{1}{2}(D\mathbf{F}^T[\Delta \mathbf{u}] \delta \mathbf{F} + \delta \mathbf{F}^T D\mathbf{F}[\Delta \mathbf{u}]) = \frac{1}{2}(\nabla \Delta \mathbf{u}^T \delta \mathbf{F} + \delta \mathbf{F}^T \nabla \Delta \mathbf{u}).$$

Considerando um único elemento finito e , escreve-se a variação do tensor de deformação finita como

$$[\Delta \mathbf{E}] = \mathbf{B}_N \Delta \hat{\mathbf{u}}_e, \quad \mathbf{B}_N \equiv \mathbf{B}_N(\mathbf{X}, \mathbf{u}).$$

Com isso, discretiza-se o integrando do primeiro termo de (C.2) da seguinte forma:

$$\mathbf{C}(\mathbf{x}) : \mathbf{DE}[\Delta \mathbf{u}] \cdot \mathbf{DE}[\delta \mathbf{x}] = [\mathbf{C}(\mathbf{x})] \mathbf{B}_N \Delta \hat{\mathbf{u}}_e \cdot \mathbf{B}_N \delta \hat{\mathbf{x}}_e = \mathbf{B}_N^T [\mathbf{C}(\mathbf{x})] \mathbf{B}_N \Delta \hat{\mathbf{u}}_e \cdot \delta \hat{\mathbf{x}}_e,$$

sendo $[\mathbf{C}(\mathbf{x})]$ uma matriz 6×6 que corresponde à forma algorítmica de $\mathbf{C}(\mathbf{x})$ (i.e., o tensor de elasticidade linear).

Usando a simetria do segundo tensor de *Piola-Kirchhoff*, tem-se¹

$$\begin{aligned} \mathbf{S}(\mathbf{x}) \cdot \left[\frac{1}{2} \left(\nabla \Delta \mathbf{u}^T \nabla \delta \mathbf{x} + \nabla \delta \mathbf{x}^T \nabla \Delta \mathbf{u} \right) \right] &= \mathbf{S}(\mathbf{x}) \cdot \left(\nabla \Delta \mathbf{u}^T \nabla \delta \mathbf{x} \right)^S = \mathbf{S}(\mathbf{x}) \cdot \left(\nabla \Delta \mathbf{u}^T \nabla \delta \mathbf{x} \right) = \\ &= \left(\nabla \Delta \mathbf{u} \right) \mathbf{S}(\mathbf{x}) \cdot \nabla \delta \mathbf{x}, \end{aligned}$$

de onde se obtém o formato para implementação

$$\nabla \Delta \mathbf{u} \mathbf{S}(\mathbf{x}) \cdot \nabla \delta \mathbf{x} = \Xi(\mathbf{x}) \mathbf{B}_G \Delta \hat{\mathbf{u}}_e \cdot \mathbf{B}_G \delta \hat{\mathbf{x}}_e = \mathbf{B}_G^T \Xi(\mathbf{x}) \mathbf{B}_G \Delta \hat{\mathbf{u}}_e \cdot \delta \hat{\mathbf{x}}_e, \quad (\text{C.3})$$

sendo $\Xi(\mathbf{x})$ uma matriz simétrica formada a partir das componentes de $\mathbf{S}(\mathbf{x})$.

Por fim, resta discretizar (C.1) e assim obter a expressão das forças internas. Seguindo o mesmo procedimento anterior, tem-se

$$\mathbf{S} \cdot \delta \mathbf{E} = \{\mathbf{S}\} \cdot \mathbf{B}_N \delta \hat{\mathbf{x}}_e = \mathbf{B}_N^T \{\mathbf{S}\} \cdot \delta \hat{\mathbf{x}}_e$$

sendo $\{\mathbf{S}\}$ o vetor com a parte simétrica do tensor \mathbf{S} . As matrizes \mathbf{B}_N e \mathbf{B}_G são apresentadas explicitamente na próxima seção.

O sistema global de uma iteração de *Newton-Raphson*

$${}^{(k)} [\mathbf{K}]_{n+1} {}^{(k+1)} [\Delta \hat{\mathbf{u}}] = \left[\hat{\mathbf{f}}_{ext} \right]_{n+1} - {}^{(k)} \left[\hat{\mathbf{f}}_{int} \right]_{n+1}, \quad (\text{C.4})$$

é obtido usualmente através do processo de superposição das matrizes e vetores associados aos elementos da malha, sendo

$${}^{(k)} [\mathbf{K}]_{n+1} {}^{(k+1)} [\Delta \hat{\mathbf{u}}] = {}^{(k)} \left[\mathbf{A}_{e=1}^{Ne} \mathbf{K}_e \right]_{n+1} {}^{(k+1)} [\Delta \hat{\mathbf{u}}] =$$

¹O sobrescrito S no segundo membro da primeira igualdade indica a parte simétrica do tensor.

$$= \mathbf{A}_{e=1}^{Ne} \left\{ \int_{\mathcal{B}_e} {}^{(k)} [\mathbf{B}_N^T [\mathbf{C}(\mathbf{x})] \mathbf{B}_N]_{n+1} dV + \int_{\mathcal{B}_e} {}^{(k)} [\mathbf{B}_G^T \boldsymbol{\Xi}(\mathbf{x}) \mathbf{B}_G]_{n+1} dV \right\} {}^{(k+1)} [\Delta \hat{\mathbf{u}}] \quad (\text{C.5})$$

a matriz de rigidez tangente e

$${}^{(k)} [\hat{\mathbf{f}}_{int}]_{n+1} = \mathbf{A}_{e=1}^{Ne} \left\{ \int_{\mathcal{B}_e} {}^{(k)} [\mathbf{B}_N^T \{\mathbf{S}\}]_{n+1} dV \right\}, \quad (\text{C.6})$$

o vetor de forças internas. $[\hat{\mathbf{f}}_{ext}]_{n+1}$ é o vetor obtido dos carregamentos externos no subpasso $n+1$:

$$[\hat{\mathbf{f}}_{ext}]_{n+1} = [\mathbf{A}_{e=1}^{Ne} \hat{\mathbf{f}}_{ext_e}]_{n+1} = \mathbf{A}_{e=1}^{Ne} \left\{ \int_{\mathcal{B}_e} \mathbf{N}^T [\mathbf{b}_{n+1}]_e dV + \int_{\Gamma_{2e}} \mathbf{N}^T [\bar{\mathbf{t}}_{n+1}]_e dA \right\}. \quad (\text{C.7})$$

C.2 Formas Matriciais de Discretização

Considere uma discretização pelo **MEF**. Para um elemento tetraédrico e de 4 nós (descritos pelas coordenadas \mathbf{X}_e^i) e n_p graus de liberdade locais associados às funções de base N^i ($i = 1, \dots, n_p$) pertencentes ao elemento, são válidas as seguintes expressões de interpolação através das próprias funções de base:

$$\mathbf{X}_e(\zeta, \eta, \zeta) = \sum_{i=1}^4 \mathbf{X}_e^i N^i(\zeta, \eta, \zeta), \quad (\text{C.8})$$

$$\mathbf{u}_e = \sum_{i=1}^{n_p} \mathbf{u}_e^i N^i, \quad (\text{C.9})$$

$$\mathbf{x}_e = \mathbf{x}(\mathbf{X}_e) = \mathbf{X}_e + \mathbf{u}_e = \sum_{i=1}^{n_p} (\mathbf{X}_e^i + \mathbf{u}_e^i) N^i = \sum_{i=1}^{n_p} \mathbf{x}_e^i N^i, \quad (\text{C.10})$$

sendo ζ, η, ζ as coordenadas do ponto material em termos das coordenadas locais do elemento de referência Ω_{ref} . Uma malha \mathcal{T}_h fornece um mapeamento $\Omega^e \rightarrow \Omega_{ref}$ inversível, sendo possível relacionar univocamente as coordenadas materiais \mathbf{X} e as coordenadas locais (e suas derivadas) dentro de cada elemento. Em outras palavras, não é necessário fazer referência às coordenadas locais no desenvolvimento que se segue.

Do ponto de vista da implementação, é interessante expressar as interpolações de grandezas através de operações matriciais. No caso do deslocamento, tem-se

$$\mathbf{u}_e = \mathbf{N} \hat{\mathbf{u}}_e, \quad (\text{C.11})$$

$$\mathbf{N} = \begin{bmatrix} N^1 & 0 & 0 & N^2 & 0 & 0 & \dots & N^{Nn} & 0 & 0 \\ 0 & N^1 & 0 & 0 & N^2 & 0 & \dots & 0 & N^{Nn} & 0 \\ 0 & 0 & N^1 & 0 & 0 & N^2 & \dots & 0 & 0 & N^{Nn} \end{bmatrix}, \quad (\text{C.12})$$

$$\hat{\mathbf{u}}_e^T = \begin{bmatrix} u_1^1 & u_2^1 & u_3^1 & u_1^2 & u_2^2 & u_3^2 & \dots & u_1^{Nn} & u_2^{Nn} & u_3^{Nn} \end{bmatrix}, \quad (\text{C.13})$$

sendo \mathbf{u}_e o deslocamento de um ponto material do elemento e e $\hat{\mathbf{u}}_e$, o vetor de deslocamentos associado aos graus de liberdade do elemento.

Faz-se o mesmo para o incremento de deslocamento, ou seja,

$$\Delta \mathbf{u}_e = \mathbf{N} \Delta \hat{\mathbf{u}}_e, \quad (\text{C.14})$$

$$\Delta \hat{\mathbf{u}}_e^T = \begin{bmatrix} \Delta u_1^1 & \Delta u_2^1 & \Delta u_3^1 & \Delta u_1^2 & \Delta u_2^2 & \Delta u_3^2 & \dots & \Delta u_1^{Nn} & \Delta u_2^{Nn} & \Delta u_3^{Nn} \end{bmatrix}. \quad (\text{C.15})$$

Dado o gradiente material do campo de deslocamentos

$$\nabla \mathbf{u} = \frac{\partial u_i}{\partial X_j} = u_{i,j},$$

define-se sua forma discreta através do produto matricial

$$[\nabla \mathbf{u}]_e = \mathbf{B}_G \hat{\mathbf{u}}_e, \quad \mathbf{B}_G \equiv \mathbf{B}_G(\mathbf{X}), \quad (\text{C.16})$$

sendo

$$\mathbf{B}_G = \begin{bmatrix} N_{,1}^1 & 0 & 0 & N_{,1}^2 & 0 & 0 & \dots & N_{,1}^{Nn} & 0 & 0 \\ N_{,2}^1 & 0 & 0 & N_{,2}^2 & 0 & 0 & \dots & N_{,2}^{Nn} & 0 & 0 \\ N_{,3}^1 & 0 & 0 & N_{,3}^2 & 0 & 0 & \dots & N_{,3}^{Nn} & 0 & 0 \\ 0 & N_{,1}^1 & 0 & 0 & N_{,1}^2 & 0 & \dots & 0 & N_{,1}^{Nn} & 0 \\ 0 & N_{,2}^1 & 0 & 0 & N_{,2}^2 & 0 & \dots & 0 & N_{,2}^{Nn} & 0 \\ 0 & N_{,3}^1 & 0 & 0 & N_{,3}^2 & 0 & \dots & 0 & N_{,3}^{Nn} & 0 \\ 0 & 0 & N_{,1}^1 & 0 & 0 & N_{,1}^2 & \dots & 0 & 0 & N_{,1}^{Nn} \\ 0 & 0 & N_{,2}^1 & 0 & 0 & N_{,2}^2 & \dots & 0 & 0 & N_{,2}^{Nn} \\ 0 & 0 & N_{,3}^1 & 0 & 0 & N_{,3}^2 & \dots & 0 & 0 & N_{,3}^{Nn} \end{bmatrix} \quad (\text{C.17})$$

e

$$[\nabla \mathbf{u}]_e^T = \begin{bmatrix} u_{1,1} & u_{1,2} & u_{1,3} & u_{2,1} & u_{2,2} & u_{2,3} & u_{3,1} & u_{3,2} & u_{3,3} \end{bmatrix}. \quad (\text{C.18})$$

$[\nabla \mathbf{u}]_e^T$ é a forma matricial para implementação de $\nabla \mathbf{u}$ em um ponto material do elemento e .

O tensor de deformação infinitesimal

$$\boldsymbol{\epsilon} = \nabla \mathbf{u} = \frac{1}{2} \left(\frac{\partial u_i}{\partial X_j} + \frac{\partial u_j}{\partial X_i} \right) = \frac{1}{2} (u_{i,j} + u_{j,i})$$

é usualmente expresso em implementações como²

$$[\boldsymbol{\epsilon}]_e = \mathbf{B}_L \hat{\mathbf{u}}_e, \quad \mathbf{B}_L \equiv \mathbf{B}_L(\mathbf{X}), \quad (\text{C.19})$$

com

$$\mathbf{B}_L = \begin{bmatrix} N_{,1}^1 & 0 & 0 & N_{,1}^2 & 0 & 0 & \dots & N_{,1}^{Nn} & 0 & 0 \\ 0 & N_{,2}^1 & 0 & 0 & N_{,2}^2 & 0 & \dots & 0 & N_{,2}^{Nn} & 0 \\ 0 & 0 & N_{,3}^1 & 0 & 0 & N_{,3}^2 & \dots & 0 & 0 & N_{,3}^{Nn} \\ N_{,2}^1 & N_{,1}^1 & 0 & N_{,2}^2 & N_{,1}^2 & 0 & \dots & N_{,2}^{Nn} & N_{,1}^{Nn} & 0 \\ N_{,3}^1 & 0 & N_{,1}^1 & N_{,3}^2 & 0 & N_{,1}^2 & \dots & N_{,3}^{Nn} & 0 & N_{,1}^{Nn} \\ 0 & N_{,3}^1 & N_{,2}^1 & 0 & N_{,3}^2 & N_{,2}^2 & \dots & 0 & N_{,3}^{Nn} & N_{,2}^{Nn} \end{bmatrix} \quad (\text{C.20})$$

e

$$\begin{aligned} [\boldsymbol{\epsilon}]_e^T &= \begin{bmatrix} \epsilon_{11} & \epsilon_{22} & \epsilon_{33} & 2\epsilon_{12} & 2\epsilon_{13} & 2\epsilon_{23} \end{bmatrix} \\ &= \begin{bmatrix} u_{1,1} & u_{2,2} & u_{3,3} & (u_{1,2} + u_{2,1}) & (u_{1,3} + u_{3,1}) & (u_{2,3} + u_{3,2}) \end{bmatrix}. \end{aligned} \quad (\text{C.21})$$

Seguindo a mesma idéia para a variação do tensor de deformação finita tem-se

$$D\mathbf{E}[\Delta \mathbf{u}] = \Delta \mathbf{E} = \frac{1}{2} \left(\mathbf{F}^T \nabla \Delta \mathbf{u} + \nabla \Delta \mathbf{u}^T \mathbf{F} \right),$$

$$\mathbf{F}^T \nabla \Delta \mathbf{u} = (\mathbf{I} + \nabla \mathbf{u})^T \nabla \Delta \mathbf{u} = (\delta_{ni} + u_{n,i}) (\Delta u_{n,j}),$$

$$[\Delta \mathbf{E}]^T = \begin{bmatrix} \Delta E_{11} & \Delta E_{22} & \Delta E_{33} & 2\Delta E_{12} & 2\Delta E_{13} & 2\Delta E_{23} \end{bmatrix}, \quad (\text{C.22})$$

com

$$\Delta E_{11} = \Delta u_{1,1} + u_{n,1} \Delta u_{n,1},$$

$$\Delta E_{22} = \Delta u_{2,2} + u_{n,2} \Delta u_{n,2},$$

$$\Delta E_{33} = \Delta u_{3,3} + u_{n,3} \Delta u_{n,3},$$

$$2\Delta E_{12} = \Delta u_{1,2} + \Delta u_{2,1} + u_{n,1} \Delta u_{n,2} + u_{n,2} \Delta u_{n,1},$$

²O subscrito L em \mathbf{B}_L indica matriz de deformação *linear*

$$2\Delta E_{13} = \Delta u_{1,3} + \Delta u_{3,1} + u_{n,1}\Delta u_{n,3} + u_{n,3}\Delta u_{n,1},$$

$$2\Delta E_{23} = \Delta u_{2,3} + \Delta u_{3,2} + u_{n,2}\Delta u_{n,3} + u_{n,3}\Delta u_{n,2},$$

$$[\Delta \mathbf{E}]_e = \mathbf{B}_F [\nabla \Delta \mathbf{u}]_e, \quad \mathbf{B}_F \equiv \mathbf{B}_F(\mathbf{X}, \mathbf{u}), \quad (\text{C.23})$$

$$\mathbf{B}_F = \begin{bmatrix} 1 + u_{1,1} & 0 & 0 & u_{2,1} & 0 & 0 & u_{3,1} & 0 & 0 \\ 0 & u_{1,2} & 0 & 0 & 1 + u_{2,2} & 0 & 0 & u_{3,2} & 0 \\ 0 & 0 & u_{1,3} & 0 & 0 & u_{2,3} & 0 & 0 & 1 + u_{3,3} \\ u_{1,2} & 1 + u_{1,1} & 0 & 1 + u_{2,2} & u_{2,1} & 0 & u_{3,2} & u_{3,1} & 0 \\ u_{1,3} & 0 & 1 + u_{1,1} & u_{2,3} & 0 & u_{2,1} & 1 + u_{3,3} & 0 & u_{3,1} \\ 0 & u_{1,3} & u_{1,2} & 0 & u_{2,3} & 1 + u_{2,2} & 0 & 1 + u_{3,3} & u_{3,2} \end{bmatrix},$$

$$[\nabla \Delta \mathbf{u}]_e^T = \begin{bmatrix} \Delta u_{1,1} & \Delta u_{1,2} & \Delta u_{1,3} & \Delta u_{2,1} & \Delta u_{2,2} & \Delta u_{2,3} & \Delta u_{3,1} & \Delta u_{3,2} & \Delta u_{3,3} \end{bmatrix},$$

$$[\nabla \Delta \mathbf{u}]_e = \mathbf{B}_G \Delta \hat{\mathbf{u}}_e,$$

$$[\Delta \mathbf{E}] = \mathbf{B}_F [\nabla \Delta \mathbf{u}]_e = \mathbf{B}_F \mathbf{B}_G \Delta \hat{\mathbf{u}}_e = \mathbf{B}_N \Delta \hat{\mathbf{u}}_e, \quad \mathbf{B}_N \equiv \mathbf{B}_N(\mathbf{X}, \mathbf{u}). \quad (\text{C.24})$$

Na prática, entretanto, é mais conveniente calcular \mathbf{B}_N através a soma

$$\mathbf{B}_N = \mathbf{B}_L + \mathbf{B}_U$$

sendo (Bathe, 1996)

$$\mathbf{B}_U = \begin{bmatrix} u_{1,1}N_{,1}^1 & u_{2,1}N_{,1}^1 & u_{3,1}N_{,1}^1 & u_{1,1}N_{,1}^2 & \dots & u_{3,1}N_{,1}^{Nn} \\ u_{1,2}N_{,2}^1 & u_{2,2}N_{,2}^1 & u_{3,2}N_{,2}^1 & u_{1,2}N_{,2}^2 & \dots & u_{3,2}N_{,2}^{Nn} \\ u_{1,3}N_{,3}^1 & u_{2,3}N_{,3}^1 & u_{3,3}N_{,3}^1 & u_{1,3}N_{,3}^2 & \dots & u_{3,3}N_{,3}^{Nn} \\ u_{1,1}N_{,2}^1 + u_{1,2}N_{,1}^1 & u_{2,1}N_{,2}^1 + u_{2,2}N_{,1}^1 & u_{3,1}N_{,2}^1 + u_{3,2}N_{,1}^1 & u_{1,1}N_{,2}^2 + u_{1,2}N_{,1}^2 & \dots & u_{3,1}N_{,2}^{Nn} + u_{3,2}N_{,1}^{Nn} \\ u_{1,1}N_{,3}^1 + u_{1,3}N_{,1}^1 & u_{2,1}N_{,3}^1 + u_{2,3}N_{,1}^1 & u_{3,1}N_{,3}^1 + u_{3,3}N_{,1}^1 & u_{1,1}N_{,3}^2 + u_{1,3}N_{,1}^2 & \dots & u_{3,1}N_{,3}^{Nn} + u_{3,3}N_{,1}^{Nn} \\ u_{1,2}N_{,3}^1 + u_{1,3}N_{,2}^1 & u_{2,2}N_{,3}^1 + u_{2,3}N_{,2}^1 & u_{3,2}N_{,3}^1 + u_{3,3}N_{,2}^1 & u_{1,2}N_{,3}^2 + u_{1,3}N_{,2}^2 & \dots & u_{3,2}N_{,3}^{Nn} + u_{3,3}N_{,2}^{Nn} \end{bmatrix}$$

Diferentemente da matriz \mathbf{B}_L , que depende somente das derivadas das funções de forma em relação à geometria não-deformada, a matriz \mathbf{B}_N depende também do estado corrente de deformação através da matriz \mathbf{B}_F (ou \mathbf{B}_U). Além disso, observa-se que quando $\hat{\mathbf{u}}_e = \mathbf{0}$, $\mathbf{B}_N = \mathbf{B}_L$.

Tem-se ainda os seguintes termos para as expressões (C.5) e (C.6):

$$\{\mathbf{S}\}^T = \begin{bmatrix} S_{11} & S_{22} & S_{33} & S_{12} & S_{13} & S_{23} \end{bmatrix} \quad (\text{C.25})$$

e

$$\Xi(\mathbf{x}) = \begin{bmatrix} \mathbf{S}(\mathbf{x}) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}(\mathbf{x}) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{S}(\mathbf{x}) \end{bmatrix}.$$

As seguintes expressões também são usadas na implementação:

$$\begin{aligned} \det \mathbf{F} &= \det(\mathbf{I} + \nabla \mathbf{u}) = \\ &= \begin{vmatrix} (1 + u_{1,1}) & u_{1,2} & u_{1,3} \\ u_{2,1} & (1 + u_{2,2}) & u_{2,3} \\ u_{3,1} & u_{3,2} & (1 + u_{3,3}) \end{vmatrix} = \begin{aligned} &(1 + u_{1,1})(1 + u_{2,2})(1 + u_{3,3}) + u_{1,2}u_{2,3}u_{3,1} \\ &+ u_{1,3}u_{2,1}u_{3,2} - u_{1,3}(1 + u_{2,2})u_{3,1} \\ &- u_{1,2}u_{2,1}(1 + u_{3,3}) - (1 + u_{1,1})u_{2,3}u_{3,2} \end{aligned}, \end{aligned}$$

$$\begin{aligned} E_{ij} &= \frac{1}{2}(u_{i,j} + u_{j,i}) + \frac{1}{2}(u_{m,i}u_{m,j}), \\ E_{11} &= u_{1,1} + \frac{1}{2}(u_{1,1}u_{1,1} + u_{2,1}u_{2,1} + u_{3,1}u_{3,1}), \\ E_{22} &= u_{2,2} + \frac{1}{2}(u_{1,2}u_{1,2} + u_{2,2}u_{2,2} + u_{3,2}u_{3,2}), \\ E_{33} &= u_{3,3} + \frac{1}{2}(u_{1,3}u_{1,3} + u_{2,3}u_{2,3} + u_{3,3}u_{3,3}), \\ E_{12} &= \frac{1}{2}(u_{1,2} + u_{2,1}) + \frac{1}{2}(u_{1,1}u_{1,2} + u_{2,1}u_{2,2} + u_{3,1}u_{3,2}), \\ E_{13} &= \frac{1}{2}(u_{1,3} + u_{3,1}) + \frac{1}{2}(u_{1,1}u_{1,3} + u_{2,1}u_{2,3} + u_{3,1}u_{3,3}), \\ E_{23} &= \frac{1}{2}(u_{2,3} + u_{3,2}) + \frac{1}{2}(u_{1,2}u_{1,3} + u_{2,2}u_{2,3} + u_{3,2}u_{3,3}). \end{aligned}$$

Apêndice D

Custo Computacional dos Métodos Iterativos e Multigrid Algébricos

O procedimento de cálculo do custo computacional dos métodos iterativos e multigrid algébricos empregados neste trabalho são inteiramente baseados nas estimativas apresentadas em (Bittencourt, 1996). O cálculo do custo associado a esses métodos é feito em termos do número de operações de ponto flutuante executadas ao longo do processo iterativo. A seguir, apresentam-se as expressões utilizadas para a avaliação dos custos envolvidos nos vários métodos considerados.

D.1 Métodos Iterativos Clássicos

A Tabela D.1 traz o custo computacional dos principais métodos iterativos utilizados neste trabalho. As variáveis N_{it} , N e m representam, respectivamente, o número de iterações executadas, o número de graus de liberdade envolvidos (i.e., a ordem das matrizes globais) e o número médio de coeficientes presentes nas linhas das matrizes globais.

D.2 Método Multigrid Algébrico para a Versão- p

A estimativa do custo computacional do método multigrid algébrico considera o cálculo do custo dos diferentes procedimentos envolvidos no processo global de iterações em ciclo. Esses pro-

Método	Número de Operações
Jacobi	$N_{it}N(2m - 1)$
GS	$N_{it}N(2m - 1)$
SOR	$2N_{it}Nm$
GC	$N_{it}N(2m + 4)$
GCD	$N_{it}N(2m + 5)$
GCSS	$N_{it}N(2m + 7)$
GCGS	$N_{it}N(2m + 7)$

Tabela D.1: Expressões para o cálculo do custo computacional dos métodos iterativos empregados neste trabalho.

cedimentos e suas respectivas expressões de custo são listados a seguir:

iterações : com exceção do nível mais grosso (resolvido pelo método de eliminação de *Gauss*), a demanda computacional a cada nível i do processo pode ser calculada através da multiplicação do número total de iterações realizadas em todos os ciclos (NIT_i) pelo custo de cada iteração do método de relaxação empregado (CIT_i), ou seja,

$$\sum_{i=2}^{M_p} (NIT_i)(CIT_i) + \sum_{i=1}^{M_p-1} (NIT_i)_{comp}(CIT_i)_{comp}, \quad (D.1)$$

sendo M_p o número de níveis p utilizados em cada ciclo e *comp*, o rótulo que identifica as iterações executadas nos níveis (ou submatrizes) complementares. No caso de se usar o esquema de relaxação de *Gauss-Seidel* (GS), o custo (CIT_i) pode ser obtido a partir da Tabela D.1.

solução direta no nível mais grosso : o custo desse procedimento está associado a uma única fatoração da matriz do nível mais grosso bem como às operações de substituição executadas sempre que esse nível é visitado (Bittencourt, 1996). Assim, esse custo é calculado por

$$\left[\sum_{i=1}^{N_1} m_i^U (m_i^U + 3) \right] + 2N_1(m_1 - 1)n_1, \quad (D.2)$$

sendo N_1 a ordem da matriz correspondente ao nível mais grosso, m_i^U , o número de elementos em cada linha dessa matriz descontado o elemento da diagonal, m_1 , o número médio de elementos por linha dessa matriz e n_1 , o número total de vezes que o nível mais grosso é visitado durante o processo.

cálculo do resíduo : a cada vez que a operação de restrição é executada entre os níveis i e $i - 1$, é necessário computar o resíduo das relaxações. Esse procedimento demanda $N_i(2m_i - 1)$ operações de ponto flutuante a cada nível do processo de modo que o custo total do cálculo do resíduo pode ser escrito como

$$\sum_{i=2}^{M_p} N_i(2m_i - 1)n_i^r + \sum_{i=1}^{M_p-1} N_i^{comp}(2m_i^{comp} - 1)n_i^r, \quad (D.3)$$

sendo n_i^r o número total de operações de restrição executadas ao longo dos ciclos, N_i^{comp} , a ordem da submatriz associada ao nível complementar i e m_i^{comp} , o número médio de elementos nas linhas dessa submatriz.

Observa-se que, diferentemente dos métodos multigrid clássicos, não se computa os custos relativos às operações de restrição e prolongamento entre níveis. Isso ocorre pelo fato dos ciclos multi- p utilizarem operadores injetivos os quais não contribuem com nenhum custo adicional ao processo iterativo.

O custo computacional total associado ao método multigrid algébrico para a versão- p é dado pela soma das expressões (D.1), (D.2) e (D.3).