

ESTE EXEMPLAR CORRESPONDE A REDAÇÃO FINAL DA
TESE DEFENDIDA POR FABIANO FERNANDES
BARGOS E APROVADA
PELA COMISSÃO JULGADORA EM 10/02/09

Marco Lúcio Bittencourt
ORIENTADOR

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

Implementação de Elementos Finitos de Alta Ordem baseado em Produto Tensorial

Autor: Fabiano Fernandes Bargas

Orientador: Prof. Dr. Marco Lúcio Bittencourt

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

Implementação de Elementos Finitos de Alta Ordem baseado em Produto Tensorial

Autor: Fabiano Fernandes Bargas
Orientador: Prof. Dr. Marco Lúcio Bittencourt

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

Dissertação de mestrado apresentada à comissão de Pós Graduação da Faculdade de Engenharia Mecânica como requisito para a obtenção do título de Mestre em Engenharia Mecânica.

Campinas, 2009
S.P. - Brasil

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

B238i Bargas, Fabiano Fernandes
 Implementação de elementos finitos de alta ordem
 baseado em produto tensorial / Fabiano Fernandes
 Bargas. --Campinas, SP: [s.n.], 2009.

 Orientador: Marco Lúcio Bittencourt.
 Dissertação de Mestrado - Universidade Estadual de
 Campinas, Faculdade de Engenharia Mecânica.

 1. Simulação computacional. 2. Método dos
 Elementos Finitos. 3. MATLAB (Programa de
 computador). 4. Produto tensorial. I. Bittencourt, Marco
 Lúcio. II. Universidade Estadual de Campinas.
 Faculdade de Engenharia Mecânica. III. Título.

Título em Inglês: Implementation of high order finite element based on tensorial
product

Palavras-chave em Inglês: Finite Elements Method, High-Order Methods, Global
Continuity, Matlab, Poisson Problem

Área de concentração: Mecânica dos Sólidos e Projeto Mecânico

Titulação: Mestre em Engenharia Mecânica

Banca examinadora: Alberto Luiz Serpa, Marcílio Alves

Data da defesa: 18/02/2009

Programa de Pós Graduação: Engenharia Mecânica

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

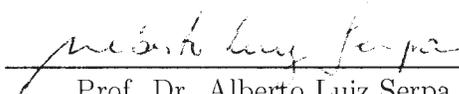
DISSERTAÇÃO DE MESTRADO

Implementação de Elementos Finitos de Alta Ordem baseado em Produto Tensorial

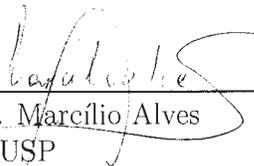
Autor: Fabiano Fernandes Bargas
Orientador: Prof. Dr. Marco Lúcio Bittencourt



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



Prof. Dr. Alberto Luiz Serpa
DMC/FEM/UNICAMP



Prof. Dr. Marcílio Alves
EPUSP/USP

Campinas, 18 de Fevereiro de 2009.

*Às duas mulheres da minha vida,
minha mãe e minha futura esposa.*

Agradecimentos

À Deus, razão da minha existência.

Ao meu orientador, Prof. Dr. Marco Lúcio Bittencourt, pela oportunidade oferecida, pela paciência e pelo tempo dedicado à minha orientação.

À minha família, por entender minha ausência, por me deixar totalmente à vontade em relação as minhas decisões profissionais e pelo apoio sempre recebido.

À pessoa com quem espero passar o restante da minha vida, Danúbia Caporusso, por sempre estar ao meu lado, principalmente nos momentos de desânimo, e pelo interesse e preocupação com o meu trabalho ao ponto de cometer a insanidade de ler todas essas páginas sem entender uma vírgula.

Ao grande amigo e irmão, Eng. Pedro Bergonzi, através de quem o sonho de trabalhar com engenharia começou a se tornar realidade, pelo grande parceiro que foi durante todo esse tempo.

Aos grandes amigos de luta, trabalho, estudo e outros desafios mais, Eng. Jaime Izuka e Eng. Guilherme Alegre, pelo fundamental companheirismo em todos os momentos.

Aos demais amigos de trabalho, Eng. Rodrigo Augusto, Eng. Sérgio Gradella, Felipe Furlan, Guilherme Neves e Bruno Salles pelo espírito colaborador no processo de aprendizagem e pelo excelente ambiente de trabalho.

À Capes pelo fundamental apoio financeiro.

”A distinção entre passado, presente e futuro
é apenas uma ilusão teimosamente persistente.”

Albert Einstein

Resumo

BARGOS, Fabiano Fernandes, *Implementação de Elementos Finitos de Alta Ordem baseado em Produto Tensorial*, Campinas: Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas (UNICAMP), 2009. 71p. Dissertação de Mestrado.

Esse trabalho apresenta uma implementação, em ambiente MatLab, de códigos para o Método dos Elementos Finitos de Alta Ordem em malhas estruturadas e não estruturadas para aplicação em problemas 2D e 3D. Apresenta-se um resumo dos procedimentos para construção das bases de funções para quadrados, triângulos, hexaedros e tetraedros através do produto tensorial. Faz-se um estudo detalhado da continuidade C^0 da aproximação para expansões modais em quadrados e mostra-se que com uma numeração adequada das funções de aresta a continuidade é automaticamente obtida. Por fim, através da imposição de uma solução analítica, analisam-se os problemas de projeção e Poisson, 2D e 3D, em malhas de quadrados, triângulos e hexaedros, para refinamentos h e p .

Palavras Chave

Método de Elementos Finitos, Métodos de Alta Ordem, Continuidade Global, MatLab, Problema de Projeção, Problema de Poisson.

Abstract

BARGOS, Fabiano Fernandes, *Implementation of High Order Finite Elements based on Tensorial Product*, Campinas : Faculty of Mechanical Engineering, State University of Campinas (UNICAMP), Universidade Estadual de Campinas, 2009. 71p. (Master's Thesis).

An implementation in MatLab environment of a code for the High Order Finite Element Method on structured and non-structured mesh for 2D and 3D application problems is showed. The construction of basis functions for squares, triangles, hexahedral and tetrahedral, based on tensorial product, is briefly presented. It is showed that the approximation continuity in modal expansions for squares can be reached with a suitable functions numbering. Finally, through a analytical solution, the 2D and 3D projection and Poisson problems are investigates in squares, triangles and hexahedrons meshes with h and p refinements.

Keywords

Finite Elements Method, High-Order Methods, Global Continuity, MatLab, Projection Problem, Poisson Problem.

Sumário

1	Introdução	1
1.1	Revisão Bibliográfica	4
1.1.1	Espaço de Funções	5
1.1.2	Códigos Computacionais	7
1.1.3	Aplicações	9
1.2	Objetivos	10
1.3	Organização do texto	11
2	Construção das Funções de Forma para o MEF de Alta Ordem usando Produto Tensorial	12
2.1	Construção das Funções de Base Unidimensionais	13
2.1.1	Base Nodal	13
2.1.2	Base Modal	15
2.2	Construção Tensorial de Bases para Quadrados e Hexaedros	18
2.2.1	Funções de Forma para Quadrados	19
2.2.2	Funções de Forma para Hexaedros	20
2.3	Construção Tensorial das Bases para Triângulos e Tetraedros	24
2.3.1	Funções de Forma para Triângulos	25
2.3.2	Funções de Forma para Tetraedros	26
2.4	Continuidade Global C^0 para Quadrados	28
2.4.1	Distribuição das Funções no Elemento	29
2.4.2	Orientação dos sistemas de coordenadas	32
3	Estrutura do Código Matlab para o MEF de Alta Ordem	37
3.1	Programa hp^2 FEM	37
3.1.1	Organização e Funcionamento	39
3.1.2	Principais Estruturas	40

3.2	Construção das Malhas de Alta Ordem	43
3.3	Detalhamento dos Elementos	48
3.3.1	Elementos Bidimensionais - Quadrados e Triângulos	48
3.3.2	Elementos Tridimensionais - Hexaedros e Tetraedros	48
4	Aplicações	51
4.1	Caracterização dos Problemas de Aplicação	52
4.1.1	Problema de Projeção	52
4.1.2	Problema de Poisson	53
4.1.3	Erro na Norma de Energia	54
4.2	Resultados	55
4.2.1	Malhas Consideradas	55
4.2.2	Problema de Projeção $2D$	59
4.2.3	Problema de Projeção $3D$	61
4.2.4	Problema de Poisson $2D$	61
5	Cosiderações Finais e Perspectivas Futuras	64
	Referências Bibliográficas	67

Lista de Figuras

1.1	Representação da aproximação de uma função.	2
2.1	Sistema local de coordenadas ξ_1	13
2.2	Funções de uma expansão <i>nodal</i> de ordem $P = 5$	14
2.3	Sistema local de coordenadas naturais.	15
2.4	Funções de uma expansão <i>modal</i> para ordem polinomial $P = 5$	17
2.5	Definição dos índices p , q e r	18
2.6	Construção tensorial das funções de base para quadrados (Bittencourt, 1991; Vazquez, 2004; Vazquez, 2008).	19
2.7	Funções para uma expansão <i>nodal</i>	21
2.8	Funções para uma expansão <i>modal</i> (os modos de aresta e face foram escalados por um fator de 4 e 16, respectivamente).	22
2.9	Construção tensorial das funções de base para hexaedros (Bittencourt, 1991; Vazquez, 2004; Vazquez, 2008).	23
2.10	Coordenadas baricêntricas para triângulos (Bittencourt, 1991; Vazquez, 2004; Vazquez, 2008).	26
2.11	Coordenadas baricêntricas para tetraedros (Bittencourt, 1991; Vazquez, 2004; Vazquez, 2008).	27
2.12	Referencial local do elemento e respectiva numeração dos nós (modos).	29
2.13	Esquema 1 para distribuição das funções de aresta para uma expansão de grau $P = 4$	30
2.14	Esquema 2 para distribuição das funções de aresta para uma expansão de grau $P = 4$	30
2.15	Continuidade global C^0 para uma malha com dois quadrados em uma expansão <i>modal</i> . Nota-se que a existência ou não da continuidade entre os elementos depende da ordem de distribuição das funções nas arestas.	31
2.16	Expansão <i>modal</i> de grau $P = 4$. Elementos 1 e 2 com referenciais equivalentes.	33

2.17	Expansão <i>modal</i> de grau $P = 4$. Elemento 2 com giro de 90° em relação ao elemento 1.	33
2.18	Expansão <i>modal</i> de grau $P = 4$. Elemento 2 com giro de 180° em relação ao elemento 1.	34
2.19	Expansão <i>modal</i> de grau $P = 4$. Elemento 2 com giro de 270° em relação ao elemento 1.	34
2.20	Expansão <i>nodal</i> de grau $P = 4$. Elementos 1 e 2 com referenciais equivalentes.	35
2.21	Expansão <i>nodal</i> de grau $P = 4$. Elemento 2 com giro de 90° em relação ao elemento 1.	35
2.22	Expansão <i>nodal</i> de grau $P = 4$. Elemento 2 com giro de 180° em relação ao elemento 1.	36
2.23	Expansão <i>nodal</i> de grau $P = 4$. Elemento 2 com giro de 270° em relação ao elemento 1.	36
3.1	Comparação entre o código convencional e vetorizado.	39
3.2	Funcionamento geral do hp^2 FEM.	40
3.3	Pontos de colocação para o quadrado de acordo com a regra de quadratura para uma expansão de grau $P = 8$	45
3.4	Pontos de colocação para o triângulo de acordo com a regra de quadratura para uma expansão de grau $P = 8$	46
3.5	Funcionamento do módulo de alta ordem.	47
3.6	Numeração das arestas dos elementos bidimensionais.	48
3.7	Hexaedro: numeração local das arestas.	49
3.8	Hexaedro: numeração local das faces.	49
3.9	Tetraedro: numeração local das arestas.	50
3.10	Tetraedro: numeração local das faces.	50
4.1	Problema de projecção.	52
4.2	Malhas de quadrados uniformes.	55
4.3	Malhas de quadrados distorcidas.	56
4.4	Malhas de quadrados com refinamento localizado.	56
4.5	Malhas de triângulos aninhadas.	57
4.6	Malhas de triângulos não-aninhadas.	57
4.7	Malhas de hexaedros.	58
4.8	Solução imposta para o problema de projecção $2D$	59
4.9	Convergência da solução para o problema de projecção $2D$	60

4.10	Convergência h e p para o problema de projeção $3D$ com malhas de hexaedros uniformes.	61
4.11	Convergência da solução para o problema de Poisson $2D$	62

Lista de Tabelas

3.1	Número de funções para cada entidade topológica.	44
-----	--	----

Símbolos

Matrizes e Vetores

$[H]$	-	Matriz quadrada qualquer
$[K]$	-	Matriz de rigidez global
$[M]$	-	Matriz de massa global
f	-	Vetor coluna dos termos independentes
\mathbf{n}	-	Vetor normal ao contorno de Ω
\hat{u}	-	Vetor coluna solução qualquer

Outras Notações

b_j	-	Elementos do vetor de carga
c_j	-	Coefficientes que não estão associados as nós dos elementos
cte	-	Valor constante
E	-	Arestas do elemento
F	-	Faces do elemento
i, j, k, l	-	Graus de liberdade
$h_p(\cdot)$	-	Polinômios de Lagrange
k_{ij}	-	Coefficientes da matriz de rigidez
m_{ij}	-	Coefficientes da matriz de massa
L_i	-	Coordenada natural em $[0, 1]$
L_{ip}	-	Pontos na direção L_i
\mathcal{L}_p	-	Polinômios obtidos através dos polinômios de Lagrange

m_{pq}	-	Coefficientes da matriz de massa unidimensional
$\{N_i\}$	-	Espaço Funcional
N_i	-	i -ésima Função do espaço funcional $\{N_i\}$
$N_p(\cdot)$	-	Função de interpolação unidimensional
$N_{pq}(\xi_i)$	-	Função de interpolação bidimensional (quadrado)
$N_{pqr}(L_i)$	-	Função de interpolação bidimensional (triângulo)
$N_{pqr}(\xi_i)$	-	Função de interpolação tridimensional (hexaedro)
$N_{pqrs}(L_i)$	-	Função de interpolação tridimensional (tetraedro)
p, q, r, s	-	Índices da tensorização
P	-	Ordem da expansão
P_i	-	Ordem da expansão na direção ξ_i ou L_i
$\mathcal{P}_n^{\alpha, \beta}(\cdot)$	-	Polinômios de Jacobi de grau n e ponderação α, β
$Re(x)$	-	Parte real de x
u_i	-	Coefficientes associados aos nós dos elementos
$u(\cdot)$	-	Função polinomial qualquer
$u_{ap}(\cdot)$	-	Solução aproximada
v	-	Função teste
V	-	Vértices do elemento
α, β	-	Coefficientes de ponderação dos polinômios de Jacobi
α_i, β_i	-	Coefficientes de ponderação dos polinômios de Jacobi na direção ξ_i ou L_i
Γ	-	Função Gamma
δ_{pq}	-	Delta de Kronecker
e	-	Erro da aproximação polinomial
ξ_i	-	Coordenada local em $[-1, 1]$
$\phi_p(\cdot)$	-	Função de interpolação nodal unidimensional
$\psi_p(\cdot)$	-	Função de interpolação modal unidimensional
ϕ_i	-	Funções de interpolação nodal
ψ_i	-	Funções de interpolação modal
Ω	-	Domínio dos problemas $2D$ e $3D$

Capítulo 1

Introdução

Em um mercado altamente competitivo e com ciclos de produção cada vez mais curtos, a engenharia assistida por computador torna-se um recurso importante. A busca por soluções rápidas, seguras e confiáveis intensifica o uso de ferramentas de simulação numérica, as quais possibilitam a análise dos produtos por meio de protótipos virtuais em todas as fases do projeto. As informações obtidas com as simulações podem auxiliar no processo de tomada de decisão de engenharia. Nesse contexto, tem se destacado o uso do Método dos Elementos Finitos (MEF).

O MEF pode ser entendido como um método matemático para solução de Problemas de Valor de Contorno (PVC), baseado em discretização, que utiliza funções polinomiais na aproximação da solução. A idéia é transformar um problema complexo na soma de diversos problemas simples. Um domínio (muitas vezes complexo) é dividido em elementos simples. Os elementos possuem pontos nodais, ou nós, e sobre cada elemento a distribuição da variável a ser determinada é considerada conhecida. Esses elementos em geral estão distorcidos em um referencial global e são mapeados em um elemento local, não distorcido, sobre o qual são definidas as funções de interpolação. Deve-se satisfazer a condição de continuidade das funções de interpolação sobre as regiões de fronteira entre os elementos e o PVC deve ser satisfeito em cada elemento. Uma vez determinadas as funções que satisfazem ambas as condições, tem-se a solução do problema.

A idéia básica do MEF é que uma função desconhecida $u(x)$ pode ser representada por uma combinação linear de funções conhecidas $\phi_n(x)$, ponderadas por coeficientes indetermi-

nados u_n , tal que,

$$u(x) \approx u_N(x) = \sum_{n=0}^N u_n \phi_n(x),$$

onde os coeficientes u_n são determinados aplicando-se por exemplo o Método de Galerkin. A Figura 1.1 ilustra uma função complexa aproximada por um conjunto de polinômios simples (Reddy, 2006). O conjunto de funções $\phi_n(x)$ também é chamado de funções de base, por constituir uma base no espaço de funções. Essas funções podem ser polinômios de Lagrange, Legendre, Chebyshev ou algum outro membro da família dos polinômios de Jacobi (Karniadakis e Sherwin, 2005; Shen e Tang, 2006).

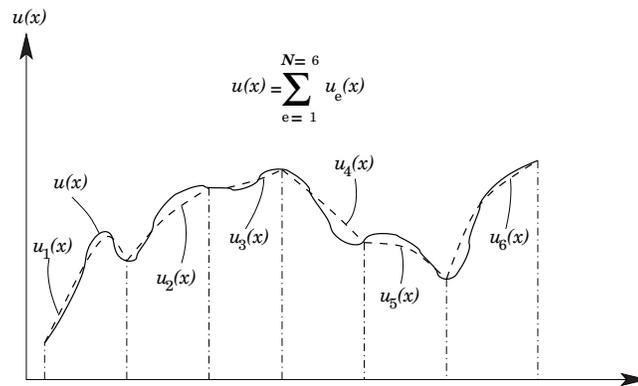


Figura 1.1: Representação da aproximação de uma função.

A análise de um problema pelo MEF envolve três fases distintas: pré-processamento, resolução numérica e pós-processamento. Na fase de pré-processamento, geram-se todos os dados de entrada do problema, entre eles os dados relativos à malha, coordenadas dos nós e incidência dos elementos. Na segunda fase, todas as informações de entrada são interpretadas, geram-se as matrizes e o problema é resolvido. Em geral, deseja-se obter a solução de uma equação matricial da forma $[H]\hat{u} = f$, a qual é resolvida invertendo-se $[H]$, tal que $\hat{u} = [H]^{-1}f$, onde $[H]$ é uma matriz, \hat{u} o vetor solução e f o vetor dos termos independentes. Finalmente, no pós-processamento os resultados são organizados e criadas todas as estruturas para visualização da solução.

Segundo Szabó e Babuska, 1991 o começo do desenvolvimento do MEF na década de 1960 foi baseado em razões intuitivas, analogias com sistemas discretos naturais e experimentação numérica. Os erros de discretização eram controlados por um refinamento uniforme

ou quase uniforme das malhas de elementos. Análises matemáticas do MEF começaram no final da década de 1960. Técnicas para estimativa de erros foram investigadas durante os anos 1970. O refinamento adaptável das malhas, objetivando a redução dos erros de discretização para aumento de eficiência, recebeu uma grande atenção durante esse período (Babuška e Rheinboldt, 1979).

Experimentos numéricos indicaram que o aumento progressivo do grau dos polinômios sobre uma malha fixa de elementos poderia ser muito mais vantajoso que o refinamento uniforme ou quase uniforme (Szabó e Mehta, 1978). Para distinguir entre os dois esquemas, ou seja, a redução de erros pelo refinamento da malha da redução de erros baseado no aumento do grau das funções de base, os nomes versão h e versão p foram popularizados.

O símbolo h normalmente é utilizado para representar o tamanho dos elementos finitos. Por isso, denomina-se *versão h* quando a convergência se dá com a redução progressiva do tamanho do elemento. Por sua vez, a ordem polinomial dos elementos é denotada pelo símbolo p . Assim, denomina-se *versão p* quando a convergência ocorre com o aumento do grau polinomial p dos elementos. As versões h e p são apenas aplicações do MEF, o qual, em princípio, permite mudanças da malha simultaneamente com o aumento da ordem polinomial dos elementos. Esta aproximação geral é usualmente chamada de versão hp do MEF. As bases teóricas para a versão p foram estabelecidas em (Babuska et al., 1981). O entendimento de como combinar efetivamente o refinamento da malha h com o refinamento p foi conseguido durante os anos 1980 (Babuska, 1988). Em trabalhos de Dinâmica de Fluidos, a versão hp normalmente é conhecida como método espectral. Karniadakis e Sherwin, 1999 apresentaram uma unificação entre a versão hp e os métodos espectrais, sendo o método obtido denominado MEF hp Espectral.

Nos últimos anos esse método tem recebido grande atenção, principalmente por apresentar convergência exponencial (também chamada espectral) da solução. Na prática, esse tipo de convergência implica que, para soluções suaves, o erro na solução numérica decai no mínimo de duas ordens ao se duplicar o número de pontos de colocação ou número de modos, diferentemente dos métodos de baixa ordem onde o erro diminui com um fator fixo (Karniadakis e Sherwin, 1999).

Conforme Karniadakis e Sherwin, 1999 e Shen e Tang, 2006, o MEF de Alta Ordem faz

parte dos chamados métodos espectrais, os quais dividem-se em duas categorias: os pseudo-espectrais ou método de colocação e os modais ou métodos de Galerkin. A primeira categoria está associada com a malha, isto é, um conjunto de nós, e é por isso referido como *nodal*. Os coeficientes desconhecidos são obtidos impondo-se que a função resíduo seja exatamente zero em um conjunto de nós. A segunda categoria está associada ao método dos resíduos ponderados, onde a função resíduo é ponderada com um conjunto de funções testes que, depois de integradas, são igualadas a zero.

A eficiência do MEF de Alta Ordem tem sido comprovada e o método aplicado aos mais diversos problemas ao longo dos últimos anos. Entretanto, o aumento da ordem da expansão polinomial implica necessariamente em um aumento da ordem das matrizes dos elementos. Esse aumento torna-se crítico, principalmente em problemas $3D$, resultando em um maior custo computacional para solução do problema. A base de funções tem um papel importante na eficiência do método. Por isso, a identificação de funções que resultem em matrizes esparsas e mais bem condicionadas tem recebido grande atenção. Outro ponto de interesse tem sido o desenvolvimento de ferramentas computacionais de alto desempenho, cuja tendência é a utilização de soluções baseadas em computação paralela e distribuída.

Sistemas distribuídos permitem que processos independentes sejam simultaneamente processados. O MEF de Alta Ordem é relativamente paralelizável, ou seja, a maioria dos seus procedimentos são independentes. Por isso, mesmo com matrizes maiores, ainda é possível o desenvolvimento de códigos extremamente eficientes, permitindo que análises cada vez mais sofisticadas e precisas sejam realizadas.

Nesse trabalho, discutem-se alguns aspectos da implementação de um código de MEF de Alta Ordem. Os códigos desenvolvidos ainda não são paralelos e foram desenvolvidos em uma base de programas, já existente, escrita em Matlab.

1.1 Revisão Bibliográfica

A denominação Método dos Elementos Finitos consolida-se com o trabalho de Clough em 1960 (Clough, 1960) (ver Oden, 1987; Clough, 2001). Atribui-se a (Turner, 1956) o marco inicial do desenvolvimento sistemático do MEF para uso como ferramenta analítica

em processos de tomada de decisão em engenharia (Szabó e Babuska, 1991). A demanda por métodos numéricos eficientes e seguros foi um fator motivador chave para o desenvolvimento do MEF. Com o programa espacial nos Estados Unidos, durante a década de 1960, a demanda aumentou significativamente. Nesse período, muito foi investido no desenvolvimento dessa tecnologia (Szabó e Babuska, 1991).

Atualmente, um grande avanço é observado. Muitos códigos de MEF foram desenvolvidos. Mackerle, 2004 apresenta, em um artigo de revisão bibliográfica, uma vasta lista de publicações, entre os anos de 1990 e 2003, sobre códigos de elementos finitos. São listados mais de duzentos artigos apenas sobre códigos orientados a objetos. Existem ainda diversos *softwares* comerciais, nas mais diversas áreas de engenharia.

Os trabalhos voltados para MEF de Alta Ordem podem ser divididos em áreas diferentes, como por exemplo:

- Espaço de funções. Trabalhos voltados ao desenvolvimento e caracterização de bases polinomiais para os elementos finitos uni, bi e tridimensionais, seja em domínios ortogonais (quadrados, hexaedros) ou não ortogonais (triângulos, tetraedros, prismas, etc). Em geral, o interesse está nas características de condicionamento e esparsidade das matrizes de massa e rigidez dos elementos.
- Desenvolvimento de códigos computacionais. São tratados aspectos de eficiência computacional, como por exemplo, aplicação de métodos diretos ou iterativos para resolução das equações, aplicação de pré-condicionadores, computação paralela e distribuída, etc.
- Aplicação em problemas físicos. Pode-se listar inúmeros trabalhos em áreas como, mecânica estrutural, dinâmica de fluidos, transferência de calor, eletromagnetismo, etc.

A seguir, faz-se uma revisão de alguns trabalhos nessas áreas.

1.1.1 Espaço de Funções

A qualidade das soluções obtidas com o MEF depende de vários fatores, dentre 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, quanto à estabilidade e eficiência dos procedimentos envolvidos na discretização (Nogueira Jr., 2002). Por simplicidade, o espaço de aproximação geralmente é formado por funções polinomiais. As funções empregadas na versão p do MEF são comumente associadas às entidades topológicas dos elementos (vértices, arestas, faces e corpo). Vários conjuntos de funções de forma estão apresentados na literatura.

A seleção de funções de forma é de extrema importância para o desempenho do algoritmo de resolução (Babuska e Helman, 1989). As funções hierárquicas clássicas para quadrados e hexaedros introduzidas em Szabó e Babuska, 1991 apresentam excelente esparsidade e bom condicionamento devido ao uso de polinômios de Legendre e sua natureza tensorial (Edgar e Surana, 1996; Maitre e Pourquier, 1995). Entretanto, as funções para triângulos e tetraedros não tem propriedades similares. Apresentam um aumento exponencial do número de condição com o aumento do grau p do elemento (Carnevali et al., 1993; Adjerid et al., 1995; Nogueira Jr., 2002).

Em (Carnevali et al., 1993) foram introduzidas funções de forma hierárquicas para triângulos e tetraedros, construídas de maneira que as funções de aresta, face e corpo de ordem p fossem ortogonais, no sentido do operador laplaciano, para essas mesmas funções com ordens não superiores a $p - 2$, $p - 3$ e $p - 4$, respectivamente. Esse fato resultou em matrizes locais com melhor condicionamento e esparsidade quando comparado às funções definidas em (Szabó e Babuska, 1991).

Karniadakis e Sherwin, 1999 apresentaram funções de forma hierárquicas para triângulos e tetraedros baseadas em sistemas de coordenadas cartesianas colapsadas. Usaram produto tensorial dos polinômios ortogonais de Jacobi unidimensionais e integração numérica exata através do produto tensorial da quadratura unidimensional de Gauss-Jacobi. Os sistemas colapsados para triângulos e tetraedros são obtidos a partir do sistemas de coordenadas cartesianas definidos sobre quadrados e hexaedros, respectivamente (Sherwin e Karniadakis, 1995).

Bittencourt, 2005 apresentou funções de forma modais e nodais para triângulos e tetraedros baseadas no produto tensorial de funções unidimensionais expressas em coordenadas baricêntricas. As funções nodais utilizam polinômios de Lagrange e são as mesmas funções

de forma h padrão apresentadas na literatura (Zienkiewicz e Taylor, 1989; Cook et al., 1991). As bases modais utilizam polinômios de Jacobi e o procedimento para construção de funções modais e nodais é simples.

Os polinômios de Jacobi têm sido empregados na versão hp do MEF dada a sua ortogonalidade, resultando em matrizes de rigidez unidimensionais quase diagonais, também ditas espectrais (Beuchler e Schöberl, 2006). Bittencourt et al., 2007 mostraram que a construção das funções de forma através de polinômios de Jacobi resulta em um aumento da esparsidade das matrizes de massa e rigidez, e que o número de condição depende da escolha dos pesos do polinômio. Observam ainda que se pode obter números de condição mais favoráveis quando utilizados polinômios de Jacobi com pesos apropriados.

Algumas citações sobre espaços de funções feitas aqui podem ser encontradas com mais detalhes em outros trabalhos realizados anteriormente sobre esse tema (Nogueira Jr., 2002; Vazquez, 2004; Vazquez, 2008).

1.1.2 Códigos Computacionais

O emprego do MEF de Alta Ordem em problemas práticos requer o uso de algoritmos eficientes para solução de grandes sistemas de equações. Essa necessidade se dá devido ao rápido aumento da ordem das matrizes dos elementos, principalmente em problemas $3D$. Na maioria dos problemas $2D$, os métodos diretos são mais eficientes. Entretanto, em casos $3D$, a capacidade de armazenamento requerida pelos métodos diretos pode tornar sua aplicação inviável. Nesses casos, os métodos iterativos se mostram mais eficientes e o método de Gradiente Conjugado Pré-condicionado (GCP) tem sido aplicado com sucesso (Nogueira Jr., 2002).

Encontra-se na literatura uma vasta lista de trabalhos sobre pré-condicionadores para o MEF de Alta Ordem, baseados nas entidades topológicas dos elementos e relacionados às técnicas de decomposição de domínios (Babuska et al., 1991; Mandel, 1990c; Mandel, 1990a; Mandel, 1990b; Ainsworth, 1996; Guo e W, 1996; Jensen e Korneev, 1997; Casarin, 1997). O objetivo básico é condensar os termos referentes aos modos internos dos elementos. A condensação normalmente é feita computando-se o complemento de Schur para cada elemento,

fato que favorece a paralelização do código (Babuska e Helman, 1989; Korneev e Jensen, 1997; Guo e W, 1997).

Aspectos da implementação do MEF usando a arquitetura MPI (*Message Passing Interface*) são discutidos em (Babuska e Helman, 1989). Oden e Patra, 1995 apresentaram técnicas de paralelização para o MEF *hp* adaptável para sistemas com memória distribuída. Confirmaram a convergência exponencial do método sendo que singularidades na solução não apresentaram dificuldades adicionais para a estratégia adotada.

Patra e Oden, 1997 mostraram novas técnicas de computação que diminuem o custo computacional associado à versão *hp* adaptável. É apresentado um algoritmo de resolução direta que reordena as incógnitas do problema, com o objetivo de reduzir o trabalho para a solução do sistema, e também um estimador de erro que pode ser combinado a esse algoritmo.

Ghosh e Basu, 1998 apresentaram um ambiente de programação paralela para a solução de problemas de larga escala usando a versão *p* do MEF baseado em decomposição de domínio. Dong e Karniadakis, 2003 mostraram, através de exemplos de Fluido Dinâmica, que, empregando MPI e fazendo um gerenciamento dinâmico do número de processadores, é possível contra balançar o custo associado à versão *p*. Dong e Karniadakis ainda apresentaram um modelo de paralelismo híbrido de dois níveis baseado em MPI/OpenMP (*Open Multi-Processing*), visando obter vantagens das estruturas hierárquicas que surgem nos problemas de Fluido Dinâmica Computacional (Dong e Karniadakis, 2004).

Um paralelismo eficiente nos atuais computadores paralelos de larga escala, os quais normalmente são equipados com 1000-10000 processadores, é um problema desafiador. Dong e Yosibashi, 2009 apresentaram uma paralelização através da decomposição do domínio, usando MPI, escalado para mais de 2000 processadores. Mostraram a eficiência do código através da medida do escalonamento, razão entre o tempo de execução sequencial e o tempo de execução do algoritmo paralelo em *p* processadores. O escalonamento do método apresentado ficou bem próximo do ideal.

1.1.3 Aplicações

A versão p do MEF foi intensamente investigada durante os últimos 20 anos, mostrando-se superior à versão h em um número significativo de problemas de importância prática (Duster e Rank, 2001). Em Mecânica dos Sólidos, por exemplo, mostrou-se eficiente fornecendo resultados robustos e confiáveis. Inicialmente, em problemas de elasticidade linear (Szabó, 1990), estendendo-se à problemas estruturais não lineares, como a elastoplasticidade (Holzer e Yosibashi, 1996; Jeremic e Xenophontos, 1999).

Duster e Rank, 2001 apresentaram uma comparação entre uma *versão* h adaptável e a versão p para um problema de plasticidade. Observaram que a versão p mostrou-se significativamente mais precisa e que a versão h pode apresentar vantagens somente quando a complexidade da estrutura requerer uma malha muito fina.

Düster et al., 2007 apresentaram uma aproximação para o estudo de placas e cascas, baseado em hexaedros, com funções de forma de alta ordem. A ordem polinomial nas três direções espaciais é ajustada por um procedimento adaptável. Para isso, um identificador de erro hierárquico é aplicado para ajustar o grau do polinômio em cada direção. A técnica mostrou-se eficiente para os problemas estudados.

Apesar do crescimento significativo de áreas de aplicabilidade do MEF de Alta Ordem, problemas dinâmicos em Mecânica dos Sólidos têm recebido relativamente menos atenção. No trabalho desenvolvido por Dong e Yosibashi, 2009, também citado na sessão anterior, a formulação hp do MEF foi aplicada em problemas dinâmicos de elasticidade não linear tridimensional. Ao invés dos polinômios de Legendre, comumente usados nesse tipo de aplicação, foram empregados os polinômios de Jacobi, devido à sua maior generalidade e às vantagens apresentadas, como matrizes de massa e rigidez com um número de condição mais favorável, conforme apresentado em (Bittencourt et al., 2007). Pela comparação com soluções analíticas, demonstrou-se que, para os problemas apresentados, o método tem convergência espacial exponencial e precisão de segunda ordem no tempo.

Encontram-se ainda aplicações do MEF de Alta Ordem em problemas de Dinâmica de Fluidos e Transferência de Calor (Lomtev et al., 1998; Karniadakis e Sherwin, 1999; Schwab, 1999; Beskok e Warburton, 2001; Karniadakis e Sherwin, 2005), e em problemas

de eletromagnetismo (Demkowicz e Vardapetyan, 1998; Ledgera et al., 2003).

1.2 Objetivos

Esse trabalho tem como objetivo central a implementação, em ambiente MatLab, de códigos para o MEF de alta ordem em malhas estruturadas e não estruturadas para aplicação em problemas 2D e 3D. Procura-se dar continuidade aos trabalhos desenvolvidos em (Vazquez, 2004; Bittencourt et al., 2007; Vazquez, 2008). Os procedimentos desenvolvidos foram implementados no código hp^2 FEM, o qual vem sendo desenvolvido nos últimos anos na FEM pelo grupo do Prof. Dr. Marco Lúcio Bittencourt. Inicialmente, apresenta-se um resumo dos procedimentos para construção das bases para quadrados, triângulos, hexaedros e tetraedros através do produto tensorial conforme (Szabó e Babuska, 1991; Karniadakis e Sherwin, 1999; Bittencourt et al., 2007).

Nos últimos anos, o grupo tem realizado um trabalho cuidadoso e criterioso visando a capacitação para desenvolvimento de códigos em elementos finitos de alta ordem. Os trabalhos desenvolvidos até o momento estão voltados para a construção e caracterização das funções de base, bem como das propriedades de condicionamento e esparsidade das matrizes de massa e rigidez locais dos elementos. Como geralmente os problemas práticos de elementos finitos envolvem vários elementos, surge a necessidade da análise de aspectos globais do MEF de Alta Ordem, como por exemplo, a construção dos elementos e a continuidade global entre eles. Assim, um dos objetivos está na geração e compatibilização dos nós (bases nodais) ou modos (bases modais) para o MEF de Alta Ordem, além da aplicação das condições de contorno para esses nós e modos gerados internamente nas expansões com polinômio de grau maior que um.

Uma contribuição original desse trabalho é mostrar que a continuidade da aproximação entre os elementos para bases modais pode ser atingida através da numeração adequada das funções.

Por fim, procura-se apresentar alguns resultados obtidos para problemas 2D e 3D, os quais têm sido usados para validação do código.

1.3 Organização do texto

O texto está organizado da seguinte maneira. No Capítulo 2, apresentam-se procedimentos uniformes para a construção de bases de funções para o MEF de Alta Ordem em linhas, quadrados, triângulos, hexaedros e tetraedros baseados no produto tensorial unidimensional usando polinômios de Lagrange e Jacobi (Karniadakis e Sherwin, 1999; Vazquez, 2004; Bittencourt et al., 2007; Vazquez, 2008). São usados índices apropriados para denotar os polinômios unidimensionais em cada direção da tensorização. A manipulação apropriada dos índices permite obter bases hierárquicas ou não hierárquicas e com continuidade C^0 ou não entre elementos.

No capítulo 3, apresentam-se detalhadamente algumas estruturas do código hp^2 FEM, o módulo de alta ordem e todos os procedimentos envolvidos.

O código desenvolvido foi usado para a resolução de problemas $2D$ e $3D$ e os resultados obtidos são mostrados no Capítulo 4. Apresentam-se os resultados para malhas uniformes e distorcidas de quadrados, triângulos e hexaedros, para os problemas de projeção e Poisson.

Por fim, no Capítulo 5, apresentam-se as considerações finais e as perspectivas para trabalhos futuros.

Capítulo 2

Construção das Funções de Forma para o MEF de Alta Ordem usando Produto Tensorial

O produto tensorial de funções unidimensionais é uma forma eficiente para a construção de bases de funções para elementos bi e tridimensionais em códigos de MEF de alta ordem. Este capítulo apresenta inicialmente um resumo dos procedimentos para a construção das bases de funções para linhas, quadrados, hexaedros, triângulos e tetraedros, baseados no produto tensorial unidimensional usando polinômios de Jacobi e Lagrange conforme (Szabó e Babuska, 1991; Karniadakis e Sherwin, 1999; Bittencourt et al., 2007).

Um conjunto de índices é utilizado para denotar os polinômios unidimensionais em cada direção da tensorização. Com uma manipulação adequada dos índices e dos polinômios, pode-se obter bases nodais e modais, hierárquicas e não-hierárquicas. Por meio dos gráficos das funções locais de aresta, obtidos em rotinas escritas em Matlab, discute-se detalhadamente aspectos da continuidade global C^0 da expansão para quadrados. Mostra-se que a continuidade pode ser obtida apenas com a manipulação dos índices.

Todas as bases apresentadas nesse trabalho podem ser expressas em termos do produto tensorial das funções unidimensionais, inclusive para elementos não estruturados, triângulos e tetraedros, como mostrado em (Bittencourt, 2005).

2.1 Construção das Funções de Base Unidimensionais

2.1.1 Base Nodal

As funções nodais de interpolação dos elementos unidimensionais são definidas em termos dos polinômios de Lagrange. Seja um elemento unidimensional definido em Ω , tal que, $\Omega = \{\xi_1 | -1 \leq \xi_1 \leq 1\}$, como mostrado na Figura 2.1. Dado um conjunto de $P + 1$ pontos nodais desse domínio, denotado por ξ_{1q} ($0 \leq q \leq P$), os polinômios de Lagrange $h_p^P(\xi)$ são os únicos polinômios de grau P , os quais valem um em ξ_{1p} e zero em ξ_{1q} ($p \neq q$). Escreve-se, portanto, $h_p^P(\xi_{1q}) = \delta_{pq}$, onde δ_{pq} é o delta de Kronecker.

Os polinômios de Lagrange são expressos na forma de um quociente de produtos, ou seja,

$$h_p^P(\xi_1) = \frac{\prod_{q=0, q \neq p}^P (\xi_1 - \xi_{1q})}{\prod_{q=0, q \neq p}^P (\xi_{1p} - \xi_{1q})}. \quad (2.1)$$

sendo $h_p^0(\xi_1) = 1$.

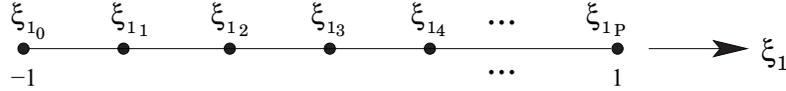


Figura 2.1: Sistema local de coordenadas ξ_1 .

As funções nodais de interpolação $N_p(\xi_1)$ associadas aos nós p ($p = 0, 1, 2, \dots, P$) dos elementos unidimensionais são os próprios polinômios de Lagrange, ou seja, $N_p(\xi_1) = h_p^P(\xi_1)$. É comum associar as funções de interpolação às entidades topológicas do elemento que, no caso do elemento unidimensional são vértices ($p = 0$ e $p = P$) e corpo ($0 < p < P$). Usualmente, fatora-se a expressão geral (2.1) dos polinômios de Lagrange e emprega-se a seguinte expressão para denotar as funções de forma de vértice e de corpo

$$N_p(\xi_1) \mapsto \phi_p(\xi_1) = \begin{cases} \frac{1}{2}(1 - \xi_1)\mathcal{L}_p^{P-1}(\xi_1), & p = 0, \\ \frac{1}{2}(1 + \xi_1)\mathcal{L}_p^{P-1}(\xi_1), & p = P, \\ \frac{1}{4}(1 - \xi_1)(1 + \xi_1)\mathcal{L}_p^{P-2}(\xi_1), & 0 < p < P, \end{cases} \quad (2.2)$$

sendo $\mathcal{L}_p^{P-1}(\xi_1)$ e $\mathcal{L}_p^{P-2}(\xi_1)$ diretamente obtidos da expressão geral (2.1) dos polinômios de

Lagrange e dados por

$$\mathcal{L}_p^{P-1}(\xi_1) = -\frac{\prod_{q=1, q \neq p}^P (\xi_1 - \xi_{1q})}{\prod_{q=1, q \neq p}^P (\xi_{1p} - \xi_{1q})}, \quad (2.3)$$

$$\mathcal{L}_p^{P-2}(\xi_1) = -4 \frac{\prod_{q=1, q \neq p}^{P-1} (\xi_1 - \xi_{1q})}{\prod_{q=1, q \neq p}^{P-1} (\xi_{1p} - \xi_{1q})}. \quad (2.4)$$

As funções para uma expansão de ordem $P = 5$ são mostradas na Figura 2.2. Observe-se que todas as funções são polinômios de ordem P , por isso a base é dita não-hierárquica. Note que todas as funções de corpo $\phi_i(\xi_1)$ ($i = 1, 2, 3, 4$) são iguais a zero nas bordas do elemento.

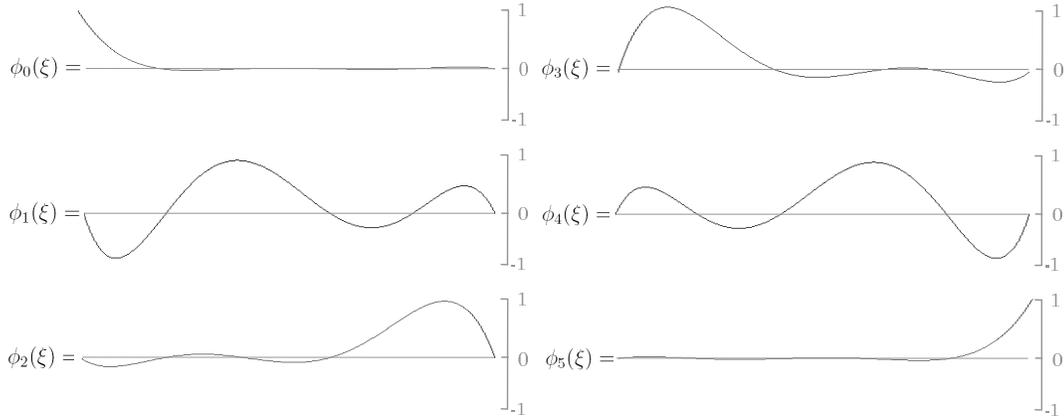


Figura 2.2: Funções de uma expansão *nodal* de ordem $P = 5$.

Pode-se ainda trabalhar com coordenadas naturais adimensionais (Cook et al., 1991). Para isso, considere a reta de comprimento l mostrada na Figura 2.3, sendo $l_1 + l_2 = l$. As coordenadas naturais de qualquer ponto P são definidas como

$$L_1 = \frac{l_1}{l} \quad \text{e} \quad L_2 = \frac{l_2}{l}.$$

Logo,

$$L_1 + L_2 = 1.$$

Nota-se pela expressão anterior que uma das coordenadas é dependente. As coordenadas naturais L_1 e L_2 variam no intervalo $[0, 1]$ e os mapeamentos entre os referenciais locais cartesianos ξ_1 e natural L_1 são dados por

$$L_1(\xi_1) = \frac{1}{2}(1 + \xi_1) \quad \text{e} \quad \xi_1(L_1) = 2L_1 - 1. \quad (2.5)$$

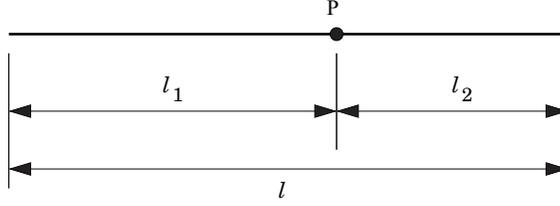


Figura 2.3: Sistema local de coordenadas naturais.

É possível escrever a expressão geral dos polinômios de Lagrange, equação (2.1), em termos das coordenadas naturais. Considerando $P + 1$ pontos na direção L_1 , tem-se

$$h_p^P(L_1) = \frac{\prod_{q=0, q \neq p}^P (L_1 - L_{1q})}{\prod_{q=0, q \neq p}^P (L_{1p} - L_{1q})}. \quad (2.6)$$

Usando as relações anteriores, escreve-se a equação (2.2) em termos das coordenadas naturais

$$N_p(L_1) \mapsto \phi_p(L_1) = \begin{cases} (1 - L_1) \mathcal{L}_p^{P-1}(L_1), & p = 0, \\ L_1 \mathcal{L}_p^{P-1}(L_1), & p = P, \\ (1 - L_1) L_1 \mathcal{L}_p^{P-2}(L_1), & 0 < p < P. \end{cases} \quad (2.7)$$

Nota-se pela definição dos polinômios de Lagrange, expressão (2.1), que as funções (2.2) e (2.7) estão diretamente relacionadas aos nós do elemento. Por isso, a base é dita *nodal*.

2.1.2 Base Modal

Em geral, bases modais são definidas em termos de polinômios ortogonais de Jacobi, sendo Legendre e Chebyshev casos especiais de polinômios dessa família. Para bases modais, apenas as funções de vértice estão associadas às coordenadas nodais.

Seja w uma função definida no intervalo $[a, b]$, $-\infty \leq a < b \leq \infty$, e não identicamente nula em (a, b) . Se

$$u_p = \int_a^b x^p w(x) dx = 0,$$

para $p = 0, 1, 2, \dots$, então w é chamada de função peso. Uma seqüência de polinômios $\{P_p\}_{p=0}^{\infty}$, com P_p de grau exatamente p , que satisfaz

$$\langle P_p, P_q \rangle = \int_a^b P_p(x) P_q(x) w(x) dx = 0, \quad p \neq q$$

é chamada seqüência de polinômios ortogonais com relação à função peso w no intervalo $[a, b]$.

Os polinômios ortogonais de Jacobi de grau p , denotados aqui por $\mathcal{P}_p^{\alpha,\beta}(\xi_1)$, são uma família de soluções polinomiais para o problema singular de Sturm-Liouville, o qual, para $-1 < \xi_1 < 1$, é escrito como (Karniadakis e Sherwin, 1999),

$$\frac{d}{d\xi_1}[(1 - \xi_1)^{1+\alpha}(1 + \xi_1)^{1+\alpha} \frac{d}{d\xi_1} \mathcal{P}_p^{\alpha,\beta}(\xi_1)] = \lambda_p(1 - \xi_1)^\alpha(1 + \xi_1)^\beta \mathcal{P}_p^{\alpha,\beta}(\xi_1), \quad (2.8)$$

sendo $\lambda_p = -p(\alpha + \beta + p + 1)$.

Esses polinômios são ortogonais com relação à função peso $w(\xi_1) = (1 - \xi_1)^\alpha(1 + \xi_1)^\beta$, definida em $[-1, 1]$, com $\alpha, \beta \in \mathfrak{R}, \alpha, \beta > -1$, isto é,

$$\int_{-1}^1 (1 - \xi_1)^\alpha(1 + \xi_1)^\beta \mathcal{P}_p^{\alpha,\beta}(\xi_1) \mathcal{P}_q^{\alpha,\beta}(\xi_1) d\xi_1 = C \delta_{pq}, \quad \xi_1 \in [-1, 1]. \quad (2.9)$$

A constante C depende de α, β e p , isto é,

$$C = \frac{2^{\alpha+\beta+1}}{2p + \alpha + \beta + 1} \frac{\Gamma(p + \alpha + 1)\Gamma(p + \beta + 1)}{p! \Gamma(p + \alpha + \beta + 1)}, \quad (2.10)$$

sendo Γ a função Gama, que pode ser dada como a integral de Euler de segunda espécie por

$$\Gamma(x) = \int_0^\infty e^{-t} t^{x-1} dt, \quad \text{Re}(x) > 0 \quad \text{ou} \quad x > 0.$$

A relação (2.9) implica que $\mathcal{P}_p^{\alpha,\beta}(\xi_1)$ é ortogonal a todos os polinômios de ordem menor que p quando integrada com respeito a $w(\xi_1) = (1 - \xi_1)^\alpha(1 + \xi_1)^\beta$. Uma propriedade desses polinômios é que suas raízes são reais, distintas e estão dentro do intervalo $[-1, 1]$.

Os polinômios de Jacobi e suas derivadas podem ser construídos computacionalmente através de uma relação recursiva, conforme mostrado em (Karniadakis e Sherwin, 1999).

Logo,

$$\begin{aligned} \mathcal{P}_0^{\alpha,\beta}(\xi_1) &= 1, \\ \mathcal{P}_1^{\alpha,\beta}(\xi_1) &= \frac{1}{2}[\alpha - \beta + (\alpha + \beta + 2)\xi_1], \\ a_p^1 \mathcal{P}_{p+1}^{\alpha,\beta}(\xi_1) &= (a_p^2 + a_p^3 \xi_1) \mathcal{P}_p^{\alpha,\beta}(\xi_1) - a_p^4 \mathcal{P}_{p-1}^{\alpha,\beta}(\xi_1), \\ b_p^1(\xi_1) \frac{d}{d\xi_1} \mathcal{P}_p^{\alpha,\beta}(\xi_1) &= b_p^2(\xi_1) \mathcal{P}_p^{\alpha,\beta}(\xi_1) + b_p^3(\xi_1) \mathcal{P}_{p-1}^{\alpha,\beta}(\xi_1), \end{aligned} \quad (2.11)$$

sendo

$$\begin{aligned}
a_p^1 &= 2(p+1)(p+\alpha+\beta+1)(2p+\alpha+\beta), \\
a_p^2 &= (2p+\alpha+\beta+1)(\alpha^2-\beta_2), \\
a_p^3 &= (2p+\alpha+\beta)(2p+\alpha+\beta+1)(2p+\alpha+\beta+2), \\
a_p^4 &= 2(p+\alpha)(p+\beta)(2p+\alpha+\beta+2), \\
b_p^1(\xi_1) &= (2p+\alpha+\beta)(1-\xi_1^2), \\
b_p^2(\xi_1) &= p(\alpha-\beta-(2p+\alpha+\beta)\xi_1), \\
b_p^3(\xi_1) &= 2(p+\alpha)(p+\beta).
\end{aligned} \tag{2.12}$$

Uma base *modal*, definida nos sistemas de coordenadas ξ_1 e L_1 pode ser obtida de forma análoga às equações (2.2) e (2.7). Para isso, substitui-se os polinômios de Lagrange pelos de Jacobi nas funções de corpo mantendo as funções de vértices lineares, ou seja,

$$N_p(\xi_1) \mapsto \psi_p(\xi_1) = \begin{cases} \frac{1}{2}(1-\xi_1), & p=0, \\ \frac{1}{2}(1+\xi_1), & p=P, \\ \frac{1}{4}(1-\xi_1)(1+\xi_1)\mathcal{P}_{p-1}^{\alpha,\beta}(\xi_1), & 0 < p < P. \end{cases} \tag{2.13}$$

e

$$N_p(L_1) \mapsto \psi_p(L_1) = \begin{cases} (1-L_1), & p=0, \\ L_1, & p=P, \\ (1-L_1)L_1\mathcal{P}_{p-1}^{\alpha,\beta}(2L_1-1), & 0 < p < P. \end{cases} \tag{2.14}$$

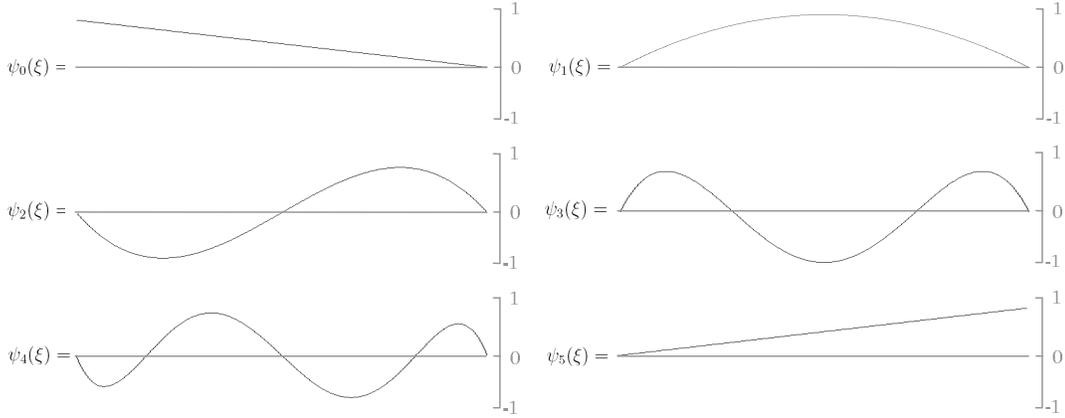


Figura 2.4: Funções de uma expansão *modal* para ordem polinomial $P = 5$.

Os modos para uma expansão *modal* com $P = 5$, normalizados para ter valor máximo igual a um, são mostradas na Figura 2.4. Os modos de vértices, $p = 0$ e $p = P$, são os mesmos do elemento linear de Lagrange. Note que os modos de corpo $p < 0 < P$ são iguais a zero

nas bordas do elemento e aumentam em ordem polinomial. Assim, o conjunto de funções de base que gera o espaço de aproximação para uma dada ordem P contém integralmente o conjunto de funções de base do espaço de aproximação de ordem $P - 1$. Por isso, a base é dita hierárquica.

A hierarquia do espaço de funções permite que as matrizes locais não sejam totalmente reconstruídas com o aumento da ordem polinomial. Esse fato, aliado a propriedade de ortogonalidade entre as funções de base de Jacobi, a qual resulta em matrizes de rigidez acentuatadamente esparsas e bem condicionadas, é bastante desejável do ponto de vista de implementação (Nogueira Jr., 2002).

2.2 Construção Tensorial de Bases para Quadrados e Hexaedros

As bases de funções para quadrados e hexaedros são construídas pelo produto tensorial das funções unidimensionais nas direções ortogonais ξ_1 , ξ_2 e ξ_3 , através dos índices p , q e r definidos na Figura 2.5. São dadas respectivamente por

$$N_{pq}(\xi_1, \xi_2) = N_p(\xi_1)N_q(\xi_2), \quad 0 \leq p, q, \quad p \leq P_1 \quad \text{e} \quad q \leq P_2, \quad (2.15)$$

$$N_{pqr}(\xi_1, \xi_2, \xi_3) = N_p(\xi_1)N_q(\xi_2)N_r(\xi_3), \quad 0 \leq p, q, r, \quad p \leq P_1, \quad q \leq P_2 \quad \text{e} \quad r \leq P_3, \quad (2.16)$$

onde ξ_1 , ξ_2 e ξ_3 são coordenadas cartesianas locais padrão.

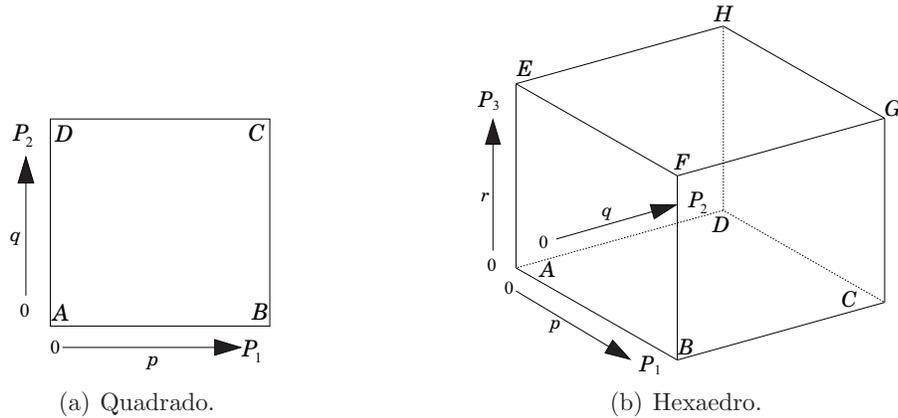


Figura 2.5: Definição dos índices p , q e r .

Nota-se que a ordem polinomial da expansão em cada direção no quadrado e hexaedro podem ser diferentes, como denotado pelo uso de P_1 , P_2 e P_3 . Para que a expansão seja completa no espaço definido por $\xi_1^{P_1}\xi_2^{P_2}\xi_3^{P_3}$ requer-se $(P_1+1)(P_2+1)$ modos em duas dimensões (quadrados) e $(P_1+1)(P_2+1)(P_3+1)$ modos em três dimensões (hexaedros).

2.2.1 Funções de Forma para Quadrados

Seja o quadrado, como mostrado na Figura 2.6, definido na região bidimensional padrão, \mathcal{Q}^2 , tal que

$$\Omega = \mathcal{Q}^2 = \{(\xi_1, \xi_2) \mid -1 \leq \xi_1, \xi_2 \leq 1\}.$$

A Figura 2.6 também ilustra as direções ortogonais ξ_1 e ξ_2 das funções unidimensionais usadas no processo de tensorização.

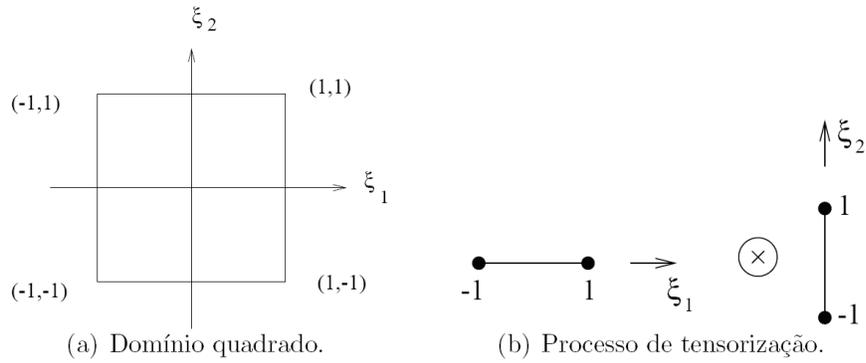


Figura 2.6: Construção tensorial das funções de base para quadrados (Bittencourt, 1991; Vazquez, 2004; Vazquez, 2008).

De forma análoga ao caso unidimensional, associam-se as funções de base às entidades topológicas do elemento. Para o quadrado, mostrado na Figura 2.5(a), escreve-se assim as expressões para as funções dos vértices

$$\begin{aligned}
 \text{vértice } A: \quad N_{00}(\xi_1, \xi_2) &= N_0(\xi_1)N_0(\xi_2), \\
 \text{vértice } B: \quad N_{P_1 0}(\xi_1, \xi_2) &= N_{P_1}(\xi_1)N_0(\xi_2), \\
 \text{vértice } C: \quad N_{P_1 P_2}(\xi_1, \xi_2) &= N_{P_1}(\xi_1)N_{P_2}(\xi_2), \\
 \text{vértice } D: \quad N_{0 P_2}(\xi_1, \xi_2) &= N_0(\xi_1)N_{P_2}(\xi_2).
 \end{aligned} \tag{2.17}$$

Analogamente, as funções de aresta, para $P \geq 2$, são

$$\begin{aligned}
\text{aresta } AB: \quad N_{p0}(\xi_1, \xi_2) &= N_p(\xi_1)N_0(\xi_2), \quad 0 < p < P_1, \\
\text{aresta } BC: \quad N_{P_1q}(\xi_1, \xi_2) &= N_{P_1}(\xi_1)N_q(\xi_2), \quad 0 < q < P_2, \\
\text{aresta } DC: \quad N_{pP_2}(\xi_1, \xi_2) &= N_p(\xi_1)N_{P_2}(\xi_2), \quad 0 < p < P_1, \\
\text{aresta } AD: \quad N_{0q}(\xi_1, \xi_2) &= N_0(\xi_1)N_q(\xi_2), \quad 0 < q < P_2.
\end{aligned} \tag{2.18}$$

Finalmente, as funções de face, para $P \geq 2$, são expressas por

$$N_{pq}(\xi_1, \xi_2) = N_p(\xi_1)N_q(\xi_2) \quad 0 < p, q, \quad p < P_1, \quad q < P_2. \tag{2.19}$$

As Figuras 2.7 e 2.8 mostram as funções do quadrado obtidas com o produto tensorial das funções unidimensionais para casos onde $P_1 = P_2 = P$. A base *nodal*, mostrada na Figura 2.7 foi gerada usando a expansão unidimensional fazendo $N_p(\xi) = \phi_p(\xi)$ (ver equação (2.2)) e a base *modal* foi gerada usando a expansão unidimensional fazendo $N_p(\xi) = \psi_p(\xi)$ (ver equação (2.13)). Para uma melhor visualização, os modos de aresta e face foram escalados por um fator de 4 e 16, respectivamente, para a expansão *modal* mostrada na Figura 2.8. Note que a expansão *modal* mantém uma forma hierárquica, ou seja, as expansões de baixa ordem são um subconjunto das expansões de alta ordem. Em contraste, cada componente da expansão *nodal* mantém a propriedade do delta Kronecker dos polinômios de Lagrange, onde cada modo tem valor unitário em uma posição específica dentro da região.

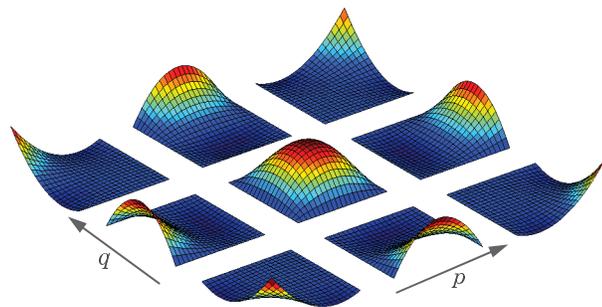
2.2.2 Funções de Forma para Hexaedros

O hexaedro, como mostrado na Figura 2.9, é definido na região tridimensional padrão, \mathcal{Q}^3 , tal que,

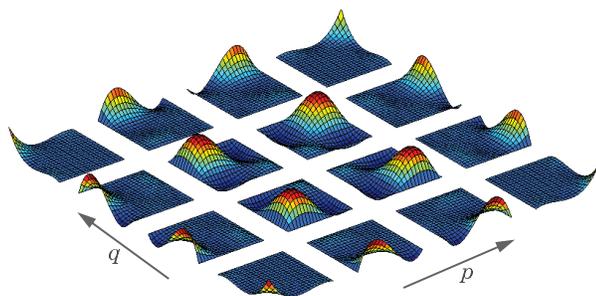
$$\Omega = \mathcal{Q}^3 = \{(\xi_1, \xi_2, \xi_3) \mid -1 \leq \xi_1, \xi_2, \xi_3 \leq 1\}.$$

A Figura 2.9 também ilustra as direções ortogonais ξ_1 , ξ_2 e ξ_3 das funções unidimensionais usadas no processo de tensorização.

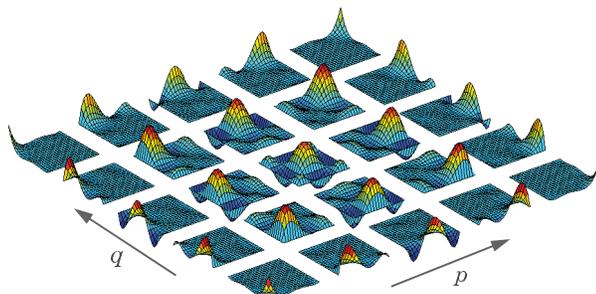
De forma análoga ao quadrado, associam-se as funções de base às entidades topológicas do elemento. Para o hexaedro, mostrado na Figura 2.5(b), escreve-se assim as expressões



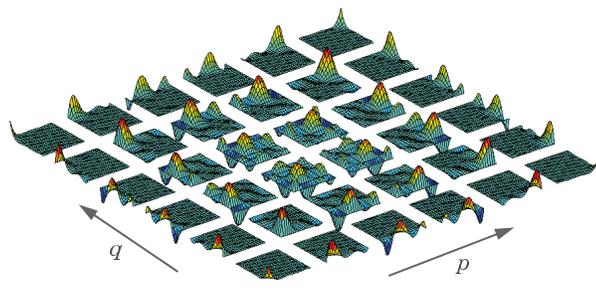
(a) $P = 2$.



(b) $P = 3$.

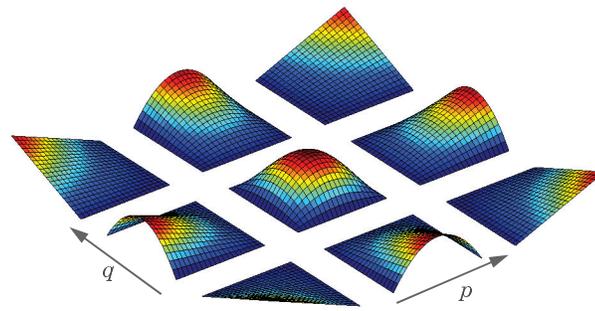


(c) $P = 4$.

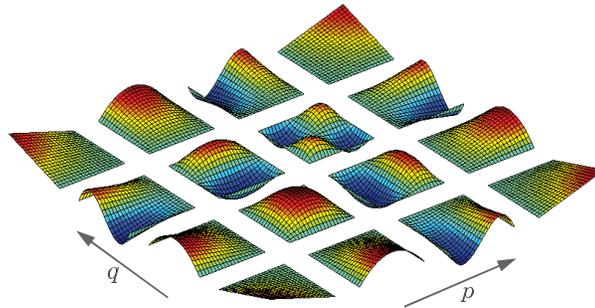


(d) $P = 5$.

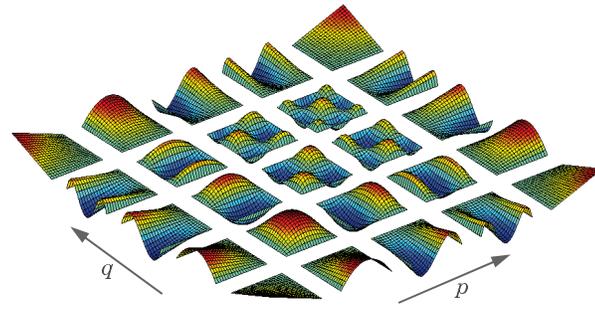
Figura 2.7: Funções para uma expansão *nodal*.



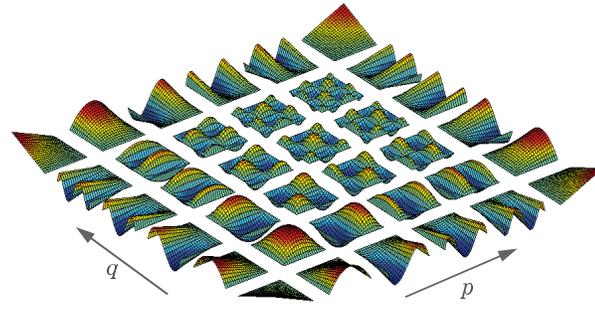
(a) $P = 2$.



(b) $P = 3$.



(c) $P = 4$.



(d) $P = 5$.

Figura 2.8: Funções para uma expansão *modal* (os modos de aresta e face foram escalados por um fator de 4 e 16, respectivamente).

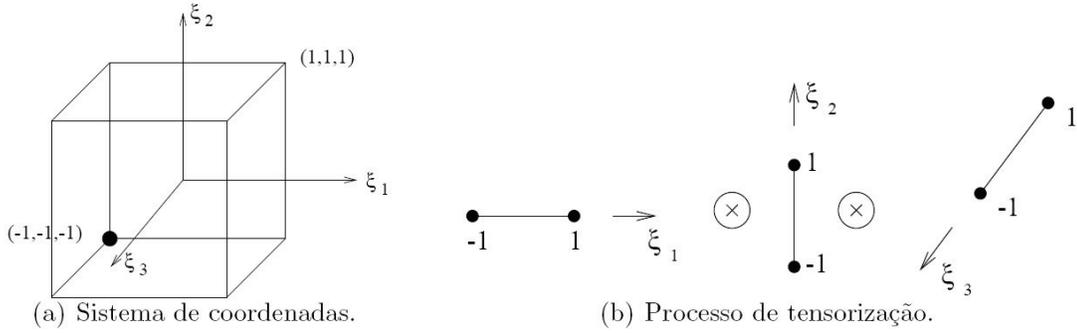


Figura 2.9: Construção tensorial das funções de base para hexaedros (Bittencourt, 1991; Vazquez, 2004; Vazquez, 2008).

para as funções dos vértices

$$\begin{aligned}
\text{vértice } A: \quad N_{000}(\xi_1, \xi_2, \xi_3) &= N_0(\xi_1)N_0(\xi_2)N_0(\xi_3), \\
\text{vértice } B: \quad N_{P_1 00}(\xi_1, \xi_2, \xi_3) &= N_{P_1}(\xi_1)N_0(\xi_2)N_0(\xi_3), \\
\text{vértice } C: \quad N_{P_1 P_2 0}(\xi_1, \xi_2, \xi_3) &= N_{P_1}(\xi_1)N_{P_2}(\xi_2)N_0(\xi_3), \\
\text{vértice } D: \quad N_{0 P_2 0}(\xi_1, \xi_2, \xi_3) &= N_0(\xi_1)N_{P_2}(\xi_2)N_0(\xi_3), \\
\text{vértice } E: \quad N_{00 P_3}(\xi_1, \xi_2, \xi_3) &= N_0(\xi_1)N_0(\xi_2)N_{P_3}(\xi_3), \\
\text{vértice } F: \quad N_{P_1 0 P_3}(\xi_1, \xi_2, \xi_3) &= N_{P_1}(\xi_1)N_0(\xi_2)N_{P_3}(\xi_3), \\
\text{vértice } G: \quad N_{P_1 P_2 P_3}(\xi_1, \xi_2, \xi_3) &= N_{P_1}(\xi_1)N_{P_2}(\xi_2)N_{P_3}(\xi_3), \\
\text{vértice } H: \quad N_{0 P_2 P_3}(\xi_1, \xi_2, \xi_3) &= N_0(\xi_1)N_{P_2}(\xi_2)N_{P_3}(\xi_3).
\end{aligned} \tag{2.20}$$

Analogamente, as funções das arestas, para $P \geq 2$, são

$$\begin{aligned}
\text{aresta } AB: \quad N_{p00}(\xi_1, \xi_2, \xi_3) &= N_p(\xi_1)N_0(\xi_2)N_0(\xi_3), \quad 0 < p < P_1, \\
\text{aresta } BC: \quad N_{P_1 q 0}(\xi_1, \xi_2, \xi_3) &= N_{P_1}(\xi_1)N_q(\xi_2)N_0(\xi_3), \quad 0 < q < P_2, \\
\text{aresta } DC: \quad N_{p P_2 0}(\xi_1, \xi_2, \xi_3) &= N_p(\xi_1)N_{P_2}(\xi_2)N_0(\xi_3), \quad 0 < p < P_1, \\
\text{aresta } AD: \quad N_{0 q 0}(\xi_1, \xi_2, \xi_3) &= N_0(\xi_1)N_q(\xi_2)N_0(\xi_3), \quad 0 < q < P_2, \\
\text{aresta } EF: \quad N_{p 0 P_3}(\xi_1, \xi_2, \xi_3) &= N_p(\xi_1)N_0(\xi_2)N_{P_3}(\xi_3), \quad 0 < p < P_1, \\
\text{aresta } FG: \quad N_{P_1 q P_3}(\xi_1, \xi_2, \xi_3) &= N_{P_1}(\xi_1)N_q(\xi_2)N_{P_3}(\xi_3), \quad 0 < q < P_2, \\
\text{aresta } HG: \quad N_{p P_2 P_3}(\xi_1, \xi_2, \xi_3) &= N_p(\xi_1)N_{P_2}(\xi_2)N_{P_3}(\xi_3), \quad 0 < p < P_1, \\
\text{aresta } EH: \quad N_{0 q P_3}(\xi_1, \xi_2, \xi_3) &= N_0(\xi_1)N_q(\xi_2)N_{P_3}(\xi_3), \quad 0 < q < P_2,
\end{aligned} \tag{2.21}$$

$$\text{aresta } AE: N_{p0r}(\xi_1, \xi_2, \xi_3) = N_p(\xi_1)N_0(\xi_2)N_0(\xi_3), \quad 0 < p < P_1,$$

$$\text{aresta } BF: N_{P_1qr}(\xi_1, \xi_2, \xi_3) = N_{P_1}(\xi_1)N_q(\xi_2)N_0(\xi_3), \quad 0 < q < P_2,$$

$$\text{aresta } CG: N_{pP_2r}(\xi_1, \xi_2, \xi_3) = N_p(\xi_1)N_{P_2}(\xi_2)N_0(\xi_3), \quad 0 < p < P_1,$$

$$\text{aresta } DH: N_{0qr}(\xi_1, \xi_2, \xi_3) = N_0(\xi_1)N_q(\xi_2)N_0(\xi_3), \quad 0 < q < P_2,$$

As funções de face, para $P \geq 2$, são

$$\text{face } ABFE: N_{p0r}(\xi_1, \xi_2, \xi_3) = N_p(\xi_1)N_0(\xi_2)N_r(\xi_3), \quad 0 < p < P_1, \quad 0 < r < P_3,$$

$$\text{face } BCGF: N_{P_1qr}(\xi_1, \xi_2, \xi_3) = N_{P_1}(\xi_1)N_q(\xi_2)N_r(\xi_3), \quad 0 < q < P_2, \quad 0 < r < P_3,$$

$$\text{face } CDHG: N_{pP_2r}(\xi_1, \xi_2, \xi_3) = N_p(\xi_1)N_{P_2}(\xi_2)N_r(\xi_3), \quad 0 < p < P_1, \quad 0 < r < P_3, \quad (2.22)$$

$$\text{face } DAEH: N_{0qr}(\xi_1, \xi_2, \xi_3) = N_0(\xi_1)N_q(\xi_2)N_r(\xi_3), \quad 0 < q < P_2, \quad 0 < r < P_3,$$

$$\text{face } ABCD: N_{pq0}(\xi_1, \xi_2, \xi_3) = N_p(\xi_1)N_q(\xi_2)N_0(\xi_3), \quad 0 < q < P_2, \quad 0 < q < P_2,$$

$$\text{face } EFGH: N_{pqP_3}(\xi_1, \xi_2, \xi_3) = N_0(\xi_1)N_q(\xi_2)N_0(\xi_3), \quad 0 < p < P_1, \quad 0 < p < P_2.$$

Finalmente, as funções de corpo, para $P \geq 2$, são expressas por

$$N_{pqr}(\xi_1, \xi_2, \xi_3) = N_p(\xi_1)N_q(\xi_2)N_0(\xi_3), \quad 0 < p, q, r, \quad p < P_1, \quad q < P_2, \quad r < P_3. \quad (2.23)$$

Bases modais e nodais são facilmente obtidas através das equações anteriores. Obtém-se uma base *nodal* fazendo $N_p(\xi) = \phi_p(\xi)$ (ver equação (2.2)) e a base *modal* fazendo $N_p(\xi) = \psi_p(\xi)$ (ver equação (2.13)).

2.3 Construção Tensorial das Bases para Triângulos e Tetraedros

Coordenadas naturais ou baricêntricas (de área e volume) historicamente têm sido empregadas na construção de base de funções para triângulos e tetraedros, devido principalmente à sua simetria rotacional. Bittencourt, 2005 apresentou um procedimento construtivo baseado no produto tensorial de polinômios unidimensionais expressos em coordenadas naturais L_i ($i = 1, 2, 3, 4$) para construção da base para triângulos e tetraedros, sendo $0 \leq L_i \leq 1$. As funções unidimensionais definidas em (2.7) e (2.14) são utilizadas nesse procedimento. As bases polinomiais para triângulos e tetraedros são dadas, respectivamente, por

$$N_{pqr}(L_1, L_2, L_3) = N_p(L_1)N_q(L_2)N_r(L_3), \quad (2.24)$$

$$N_{pqrs}(L_1, L_2, L_3, L_4) = N_p(L_1)N_q(L_2)N_r(L_3)N_s(L_4), \quad (2.25)$$

onde $0 \leq p, q, r, s, p \leq P_1, q \leq P_2, r \leq P_3$ e $s \leq P_4$, sendo P_i ($i = 1, 2, 3, 4$) o grau do polinômio nas direções L_i ($i = 1, 2, 3, 4$), respectivamente.

2.3.1 Funções de Forma para Triângulos

As coordenadas baricêntricas para triângulos estão ilustradas na Figura 2.10. Dado um triângulo de área A e um ponto qualquer P , definem-se três subtriângulos com áreas A_1, A_2 e A_3 de tal forma que $A = A_1 + A_2 + A_3$. As coordenadas de área $0 \leq L_i \leq 1$ ($i = 1, 2, 3$) são definidas pela razão entre as áreas dos subtriângulos e do triângulo original (Zienkiewicz e Taylor, 1989; Cook et al., 1991)

$$\frac{A_1}{A} + \frac{A_2}{A} + \frac{A_3}{A} = 1 \rightarrow L_1 + L_2 + L_3 = 1. \quad (2.26)$$

As funções de forma nodais para triângulos podem ser escritas como o produto tensorial dos polinômios de Lagrange nas coordenadas L_1, L_2 e L_3 da seguinte maneira

$$N_a(L_1, L_2, L_3) = l_b^{(b-1)}(L_1)l_c^{(c-1)}(L_2)l_d^{(d-1)}(L_3), \quad (2.27)$$

sendo a o número do nó; b, c e d os índices das coordenadas nodais nas direções L_1, L_2 e L_3 , respectivamente, como ilustrado na Figura 2.10. Os valores de b, c e d estão no intervalo fechado $[1, P_i + 1]$ e P_i ($i = 1, 2, 3$) denota o grau do polinômio na direção L_i .

Os polinômios de Lagrange indicados na equação anterior são dados por

$$\begin{aligned} l_b^{(b-1)}(L_1) &= \frac{(L_1 - L_{11})(L_1 - L_{12}) \cdots (L_1 - L_{1_{b-1}})}{(L_{1b} - L_{11})(L_{1b} - L_{12}) \cdots (L_{1b} - L_{1_{b-1}})}, \\ l_c^{(c-1)}(L_2) &= \frac{(L_2 - L_{21})(L_2 - L_{22}) \cdots (L_2 - L_{2_{c-1}})}{(L_{2c} - L_{21})(L_{2c} - L_{22}) \cdots (L_{2c} - L_{1_{c-1}})}, \\ l_d^{(d-1)}(L_3) &= \frac{(L_3 - L_{31})(L_3 - L_{32}) \cdots (L_3 - L_{3_{d-1}})}{(L_{3d} - L_{31})(L_{3d} - L_{32}) \cdots (L_{3d} - L_{3_{d-1}})}. \end{aligned} \quad (2.28)$$

Quando comparada à equação (2.6), deve-se observar que os polinômios de Lagrange em (2.28) são truncados nas coordenadas dos números indicados por b, c e d , respectivamente, ao invés de considerar todas as coordenadas $P_i + 1$ em cada direção L_i . Os graus dos polinômios nas direções L_i não são P_i como seria esperado mas $b-1, c-1$ e $d-1$, respectivamente. Esse é o principal ponto usado para obter as funções de forma nodais padrão para triângulos através de produto tensorial (Bittencourt, 2005). Os mesmos polinômios truncados são usados em (Blyth e Pozrikidis, 2006).

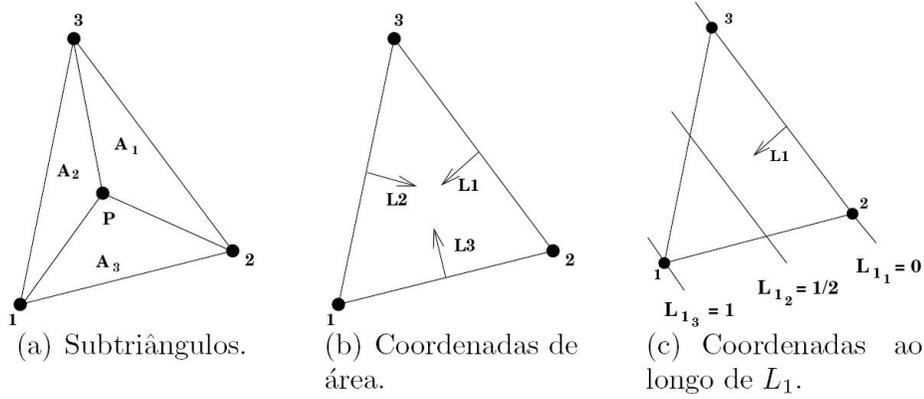


Figura 2.10: Coordenadas baricêntricas para triângulos (Bittencourt, 1991; Vazquez, 2004; Vazquez, 2008).

As funções de vértice são obtidas a partir da equação (2.26) e são dadas por

$$\begin{aligned}
 N_{P_1 00}(L_1, L_2, L_3) &= N_{P_1}(L_1)N_0(L_2)N_0(L_3), \\
 N_{0P_2 0}(L_1, L_2, L_3) &= N_0(L_1)N_{P_2}(L_2)N_0(L_3), \\
 N_{00P_3}(L_1, L_2, L_3) &= N_0(L_1)N_0(L_2)N_{P_3}(L_3).
 \end{aligned} \tag{2.29}$$

Da mesma forma, as funções de aresta são dadas por

$$\begin{aligned}
 N_{pq0}(L_1, L_2, L_3) &= N_p(L_1)N_q(L_2)N_0(L_3), 0 < p < P_1, 0 < q < P_2 \\
 N_{p0r}(L_1, L_2, L_3) &= N_p(L_1)N_0(L_2)N_r(L_3), 0 < p < P_1, 0 < r < P_3 \\
 N_{0qr}(L_1, L_2, L_3) &= N_0(L_1)N_q(L_2)N_r(L_3), 0 < q < P_2, 0 < r < P_3.
 \end{aligned} \tag{2.30}$$

Finalmente, as funções de face, para $0 < p < P_1 - 1$, $0 < q < P_2$, $0 < r < P_3$, são

$$N_{pqr}(L_1, L_2, L_3) = N_p(L_1)N_q(L_2)N_r(L_3). \tag{2.31}$$

Bases nodais e modais são facilmente obtidas através das equações anteriores. Obtém-se uma base *nodal* fazendo $N_p(L_1) = \psi_p(L_1)$ (ver equação (2.7)) e a base *nodal* fazendo $N_p(L_1) = \phi_p(L_1)$ (ver equação (2.14)).

2.3.2 Funções de Forma para Tetraedros

Para os tetraedros, aplica-se o mesmo procedimento usado nos triângulos. Utilizam-se as coordenadas baricêntricas de volume $0 \leq L_i \leq 1$ ($i = 1, 2, 3, 4$) dadas pela relação de volumes dos tetraedros, mostradas na Figura 2.11, de tal forma que

$$L_1 + L_2 + L_3 + L_4 = 1. \tag{2.32}$$

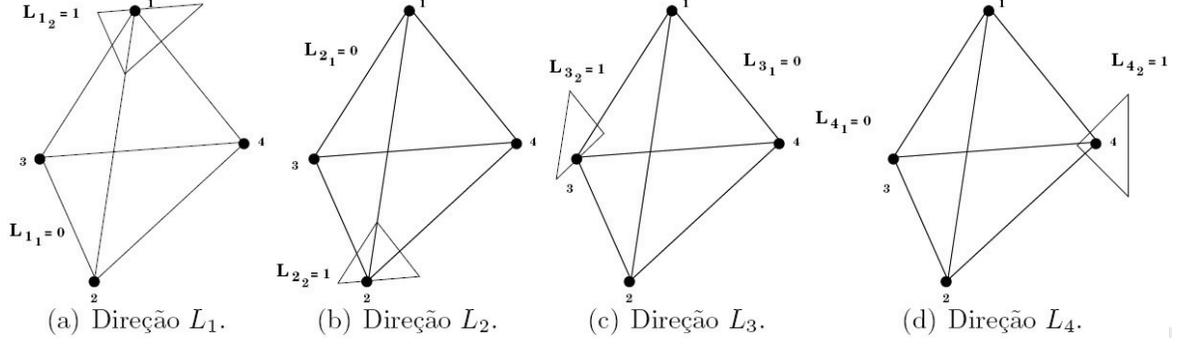


Figura 2.11: Coordenadas baricêntricas para tetraedros (Bittencourt, 1991; Vazquez, 2004; Vazquez, 2008).

Assume-se $P_1 = P_2 = P_3 = P_4 = P$ e portanto têm-se 4 funções de vértice, $6(P - 1)$ de aresta, $\frac{1}{2}(P - 1)(P - 2)$ de face e $\frac{1}{6}(P - 1)(P - 2)(P - 3)$ de corpo. As funções de vértice são obtidas a partir da equação (2.25) e da Figura 2.11 como

$$\begin{aligned}
 N_{P_1 000}(L_1, L_2, L_3, L_4) &= N_{P_1}(L_1)N_0(L_2)N_0(L_3)N_0(L_4), \\
 N_{0P_2 00}(L_1, L_2, L_3, L_4) &= N_0(L_1)N_{P_2}(L_2)N_0(L_3)N_0(L_4), \\
 N_{00P_3 0}(L_1, L_2, L_3, L_4) &= N_0(L_1)N_0(L_2)N_{P_3}(L_3)N_0(L_4), \\
 N_{000P_4}(L_1, L_2, L_3, L_4) &= N_0(L_1)N_0(L_2)N_0(L_3)N_{P_4}(L_4).
 \end{aligned} \tag{2.33}$$

Da mesma forma, as funções de aresta, para $0 < p, q, r < P$, são dadas por

$$\begin{aligned}
 N_{pq00}(L_1, L_2, L_3, L_4) &= N_p(L_1)N_q(L_2)N_0(L_3)N_0(L_4), \\
 N_{p0r0}(L_1, L_2, L_3, L_4) &= N_p(L_1)N_0(L_2)N_r(L_3)N_0(L_4), \\
 N_{p00s}(L_1, L_2, L_3, L_4) &= N_p(L_1)N_0(L_2)N_0(L_3)N_s(L_4), \\
 N_{0qrr}(L_1, L_2, L_3, L_4) &= N_0(L_1)N_q(L_2)N_r(L_3)N_0(L_4), \\
 N_{0q0s}(L_1, L_2, L_3, L_4) &= N_0(L_1)N_q(L_2)N_0(L_3)N_s(L_4), \\
 N_{00rs}(L_1, L_2, L_3, L_4) &= N_0(L_1)N_0(L_2)N_r(L_3)N_s(L_4).
 \end{aligned} \tag{2.34}$$

As funções de face, para $P \geq 3$, são

$$\begin{aligned}
 N_{pqr0}(L_1, L_2, L_3, L_4) &= N_p(L_1)N_q(L_2)N_r(L_3)N_0(L_4), \\
 N_{pq0s}(L_1, L_2, L_3, L_4) &= N_p(L_1)N_q(L_2)N_0(L_3)N_s(L_4), \\
 N_{p0qs}(L_1, L_2, L_3, L_4) &= N_p(L_1)N_0(L_2)N_q(L_3)N_s(L_4), \\
 N_{0qrs}(L_1, L_2, L_3, L_4) &= N_0(L_1)N_q(L_2)N_r(L_3)N_s(L_4).
 \end{aligned} \tag{2.35}$$

Enfim, as funções de corpo, para $P \geq 4$, são

$$N_{pqrs}(L_1, L_2, L_3, L_4) = N_p(L_1)N_q(L_2)N_r(L_3)N_s(L_4). \tag{2.36}$$

Bases nodais e modais são facilmente obtidas através das equações anteriores. Obtém-se uma base *nodal* fazendo $N_p(L_1) = \phi_p(L_1)$ (ver equação (2.7)) e a base *modal* fazendo $N_p(L_1) = \psi_p(L_1)$ (ver equação (2.14)).

2.4 Continuidade Global C^0 para Quadrados

Na resolução de equações diferenciais parciais é requerido uma dada continuidade nas regiões entre os elementos, dependendo da ordem da equação diferencial. Para equações de segunda ordem, deve-se garantir a continuidade C^0 entre os elementos.

Sabe-se que a base *nodal* de Lagrange tem continuidade C^0 natural quando realizado o procedimento de tensorização padrão (Bittencourt et al., 2007). Essa característica vem da construção tensorial a partir do polinômios unidimensionais de Lagrange, os quais apresentam a propriedade de colocação, ou seja, a função terá valor unitário na sua posição de definição (ponto nodal) e zero nos outros pontos. É verdade ainda que para um dado conjunto de $P + 1$ pontos, existe um único polinômio de Lagrange de grau P associado a cada ponto (ver equação (2.1)). Portanto, para um nó de aresta, comum a dois elementos, os polinômios de Lagrange associados naturalmente serão semelhantes.

As bases modais, entretanto, requerem um certo cuidado. Nessas bases, somente as funções de vértice estão associadas aos nós dos elementos. Os modos de aresta, face e corpo são numerados em uma ordem pré-estabelecida, segundo algum critério adotado.

A Figura 2.12 apresenta a numeração local dos nós para um elemento quadrilateral, de acordo com o referencial local (ξ_1, ξ_2) , seguindo o esquema adotado nesse trabalho. Nesse esquema, em ambas as bases (modais e nodais), numera-se primeiro os vértices, seguidos de arestas, faces e corpo, respectivamente. Inicia-se a numeração pelo vértice e segue-se sempre em sentido anti-horário.

Para elementos bidimensionais, os modos de face são nulos nas bordas dos elementos (Figuras 2.7 e 2.8), e por isso não influenciam na continuidade da expansão. Já os modos de aresta requerem um cuidado maior.

Como não são nulos em todas as bordas do domínio do elemento, os modos de aresta

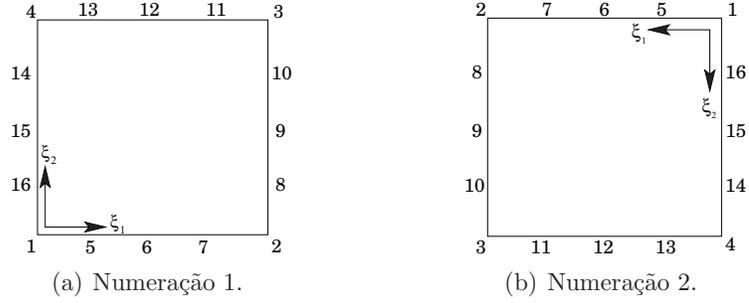


Figura 2.12: Referencial local do elemento e respectiva numeração dos nós (modos).

devem estar relacionados à modos semelhantes dos elementos vizinhos, de forma a manter a continuidade nas regiões de fronteira entre os elementos. Portanto, a distribuição dos modos nas arestas dos elementos é de fundamental importância.

Discuti-se, a seguir, dois aspectos que influenciam a continuidade de uma expansão *modal*. Primeiramente, trata-se da questão da ordem de distribuição das funções nas arestas dos elementos e em seguida da influência da orientação do sistema de coordenadas local dos elementos.

2.4.1 Distribuição das Funções no Elemento

Nas bases modais, as funções não estão associados à uma posição física no elemento. Por convenção aqui adotada, para bases modais hierárquicas, os modos são numerados da menor para a maior ordem polinomial em cada aresta. Logo, para a aresta AB o modo de menor ordem será o mais próximo do vértice A . Assim, a definição da aresta (AB ou BA) interfere na forma como os modos serão distribuídos nessa aresta. As Figuras 2.13 e 2.14 ilustram melhor essa questão.

A distribuição dos modos nas arestas deve ser de tal forma que, em uma aresta compartilhada por elementos vizinhos, as funções estejam numeradas na mesma ordem em ambos os elementos. Para um melhor entendimento dessa questão, mostram-se nas Figuras 2.13 e 2.14 dois esquemas diferentes de distribuição dos modos nas arestas. Note que as funções nas aresta AD e DC da Figura 2.13(a) são distribuídas em ordem inversa em relação às arestas AD e CD da Figura 2.14(a). No esquema 1, Figura 2.13(a), para a aresta 4 (AD) a função

de menor grau está mais próxima do vértice A e no esquema 2, Figura 2.14(a), para a aresta 4 (DA) a função de menor grau está mais próxima do vértice D . O mesmo vale para as arestas 3 (DC) Figura 2.13(a) e (CD) Figura 2.14(a).

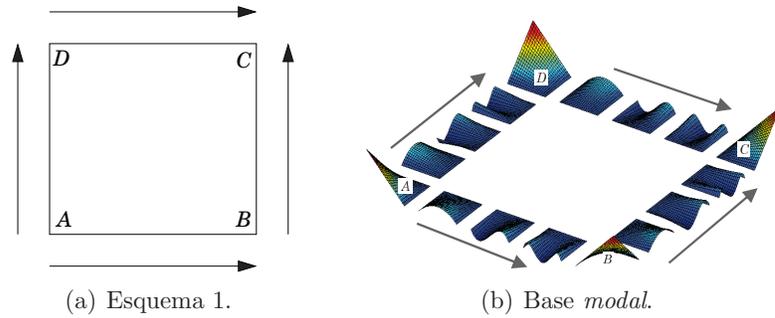


Figura 2.13: Esquema 1 para distribuição das funções de aresta para uma expansão de grau $P = 4$.

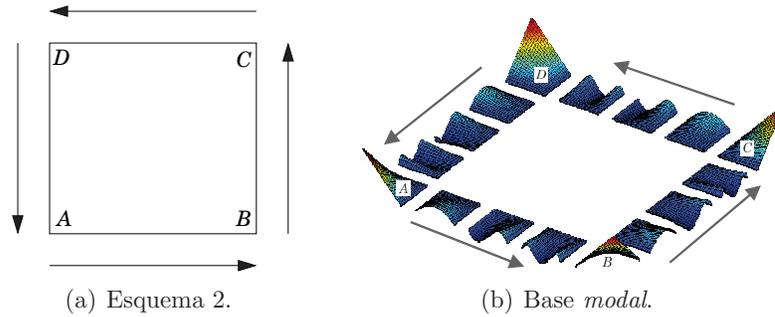
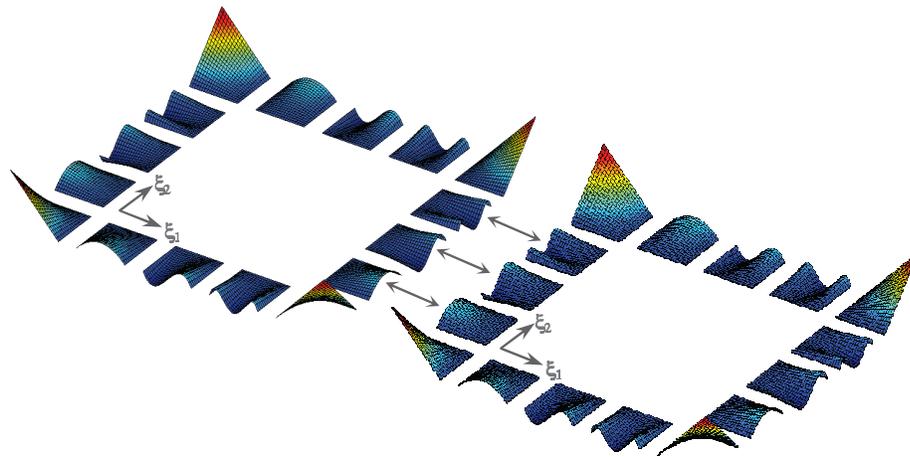


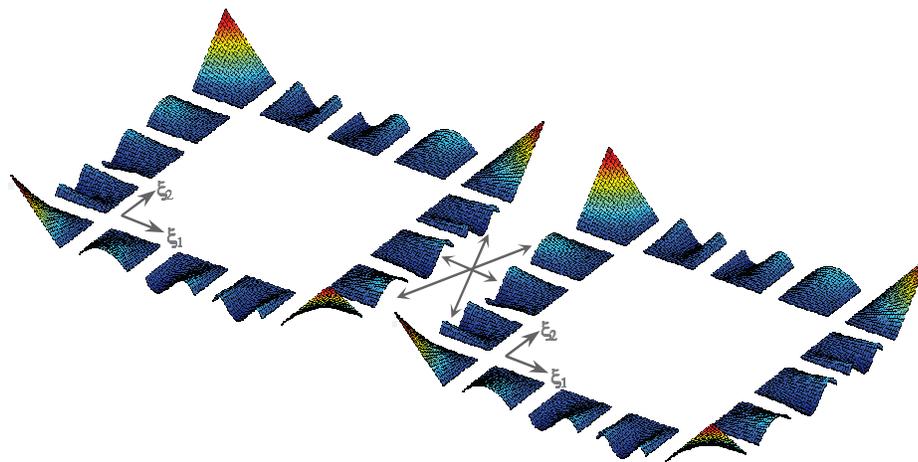
Figura 2.14: Esquema 2 para distribuição das funções de aresta para uma expansão de grau $P = 4$.

A continuidade da expansão entre os elementos é ilustrada através dos gráficos das funções locais de aresta, de forma análoga ao usado em (Karniadakis e Sherwin, 1999; Karniadakis e Sherwin, 1999). A Figura 2.15 ilustra a continuidade da expansão, para uma malha com dois quadrados, nos dois esquemas de distribuição das funções mostrados nas Figuras 2.13 e 2.14. Nota-se que, a continuidade natural da expansão está totalmente vinculada à maneira com que as funções são distribuídas nas arestas. Para o esquema 2, Figura 2.15(b), nota-se que um procedimento específico que reorganize as funções deverá ser desenvolvido para que haja continuidade.

Nesse trabalho, o esquema 1, Figura 2.13(a), resulta naturalmente do processo de tensorização.



(a) Esquema 1.



(b) Esquema 2.

Figura 2.15: Continuidade global C^0 para uma malha com dois quadrados em uma expansão *modal*. Nota-se que a existência ou não da continuidade entre os elementos depende da ordem de distribuição das funções nas arestas.

2.4.2 Orientação dos sistemas de coordenadas

Outro aspecto importante para que exista continuidade é a orientação do sistema de coordenadas local de cada elemento. As Figuras 2.16 a 2.23 ilustram as funções de aresta para uma malha de dois quadrados. As Figuras 2.16 a 2.19 ilustram as funções para uma expansão *modal* e as Figuras 2.20 a 2.23 as funções para uma expansão *nodal*, ambas para $P = 4$.

Nota-se que, para uma base *modal*, o referencial local de cada elemento é fundamental para a continuidade global da expansão (Figuras 2.16 a 2.19). Por sua vez, a continuidade da expansão *nodal* independe do referencial local dos elementos (Figura 2.20 a 2.23).

Portanto, para bases nodais, nenhum procedimento especial é requerido para que a continuidade da expansão seja garantida. A continuidade é natural, ou seja, ela é sempre obtida automaticamente, independente do sistema de coordenadas local dos elementos. Em contra-partida, para as bases modais, caso os referenciais locais de cada elemento não sejam coerentemente definidos, um procedimento que garanta a continuidade da expansão deverá ser implementado (através das Figuras 2.18 a 2.19 nota-se que alguns modos deverão ser trocados de posição (modos pares) e que os modos ímpares deverão ser multiplicados por -1). Karniadakis e Sherwin, 1999 apresentam um algoritmo que identifica para cada elemento os modos que deverão ser alterados.

No código desenvolvido, apresentado no Capítulo 3, os nós de arestas são gerados internamente, possibilitando um total controle sobre a ordem de numeração dos modos. Com isso, a continuidade é sempre obtida automaticamente, sem a necessidade de algoritmos de renumeração, os quais em geral são complexos (Nogueira Jr., 2002). Procedimentos semelhantes são usados em triângulos, hexaedros e tetraedros.

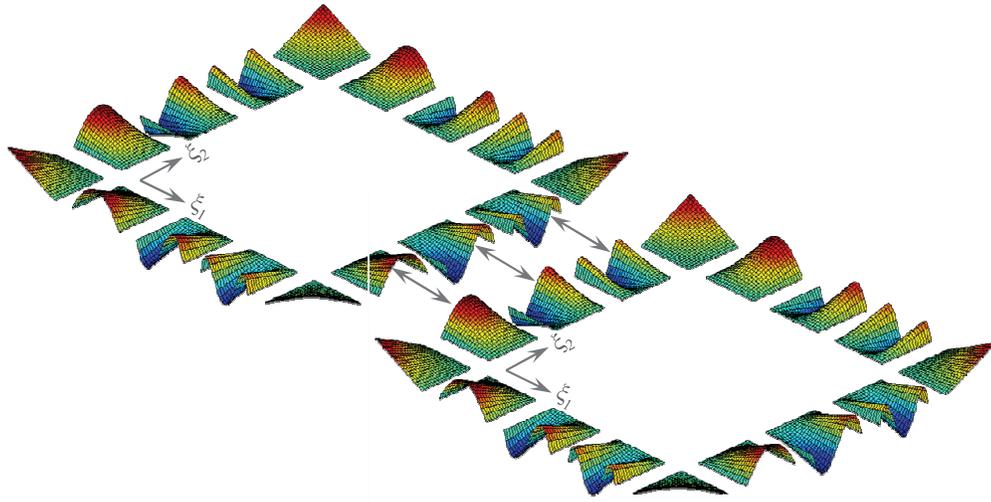


Figura 2.16: Expansão *modal* de grau $P = 4$. Elementos 1 e 2 com referenciais equivalentes.

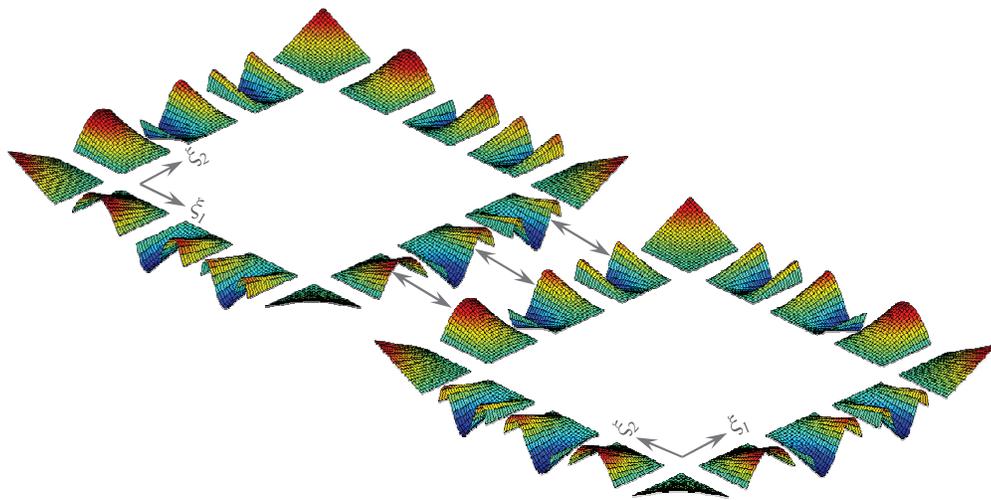


Figura 2.17: Expansão *modal* de grau $P = 4$. Elemento 2 com giro de 90° em relação ao elemento 1.

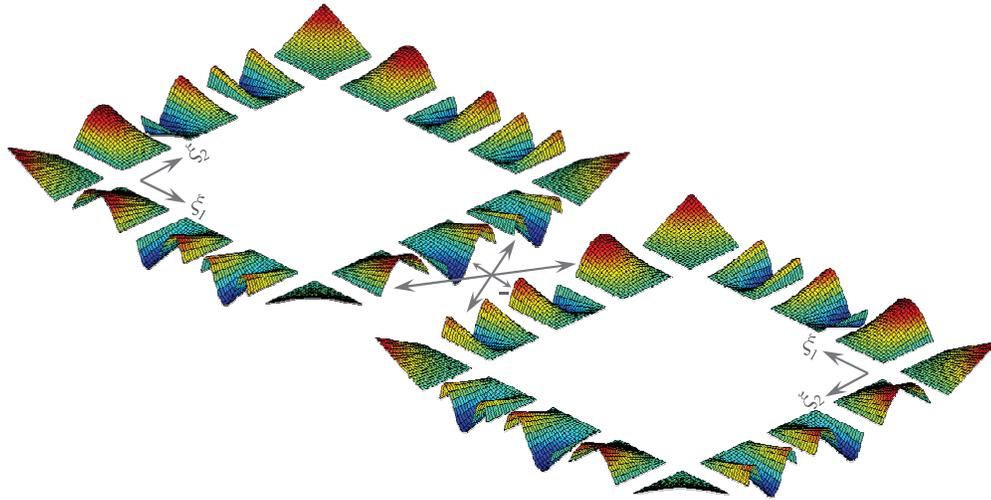


Figura 2.18: Expansão *modal* de grau $P = 4$. Elemento 2 com giro de 180° em relação ao elemento 1.

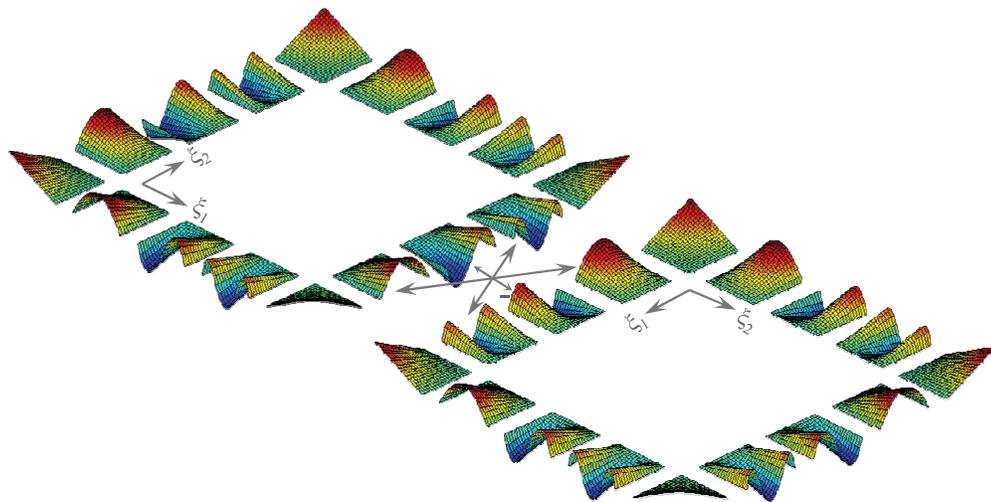


Figura 2.19: Expansão *modal* de grau $P = 4$. Elemento 2 com giro de 270° em relação ao elemento 1.

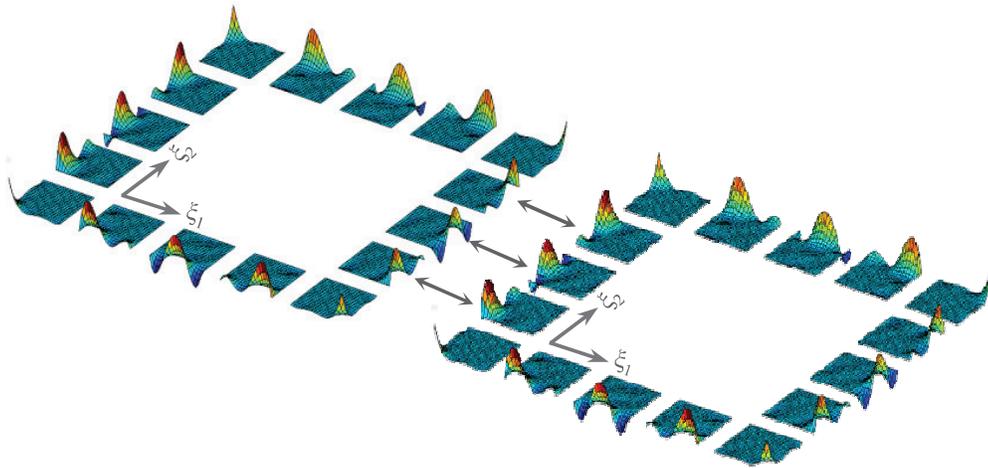


Figura 2.20: Expansão *nodal* de grau $P = 4$. Elementos 1 e 2 com referenciais equivalentes.

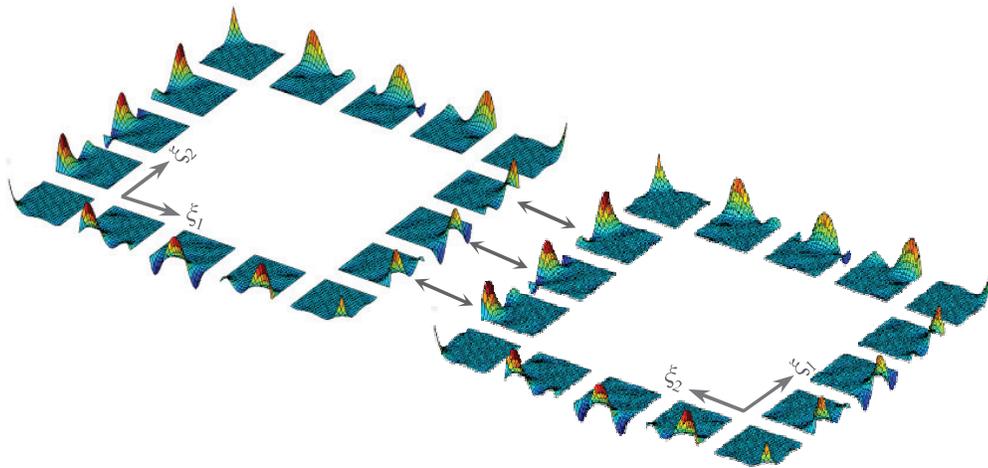


Figura 2.21: Expansão *nodal* de grau $P = 4$. Elemento 2 com giro de 90° em relação ao elemento 1.

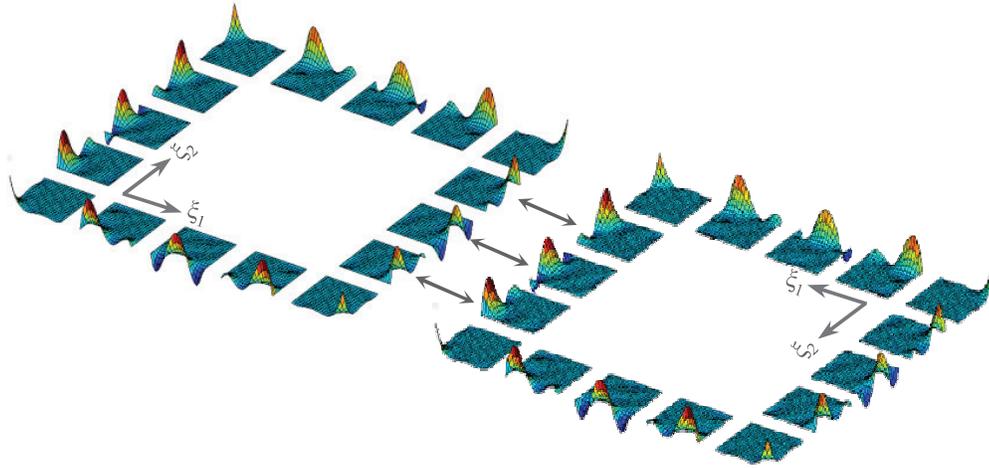


Figura 2.22: Expansão *nodal* de grau $P = 4$. Elemento 2 com giro de 180° em relação ao elemento 1.

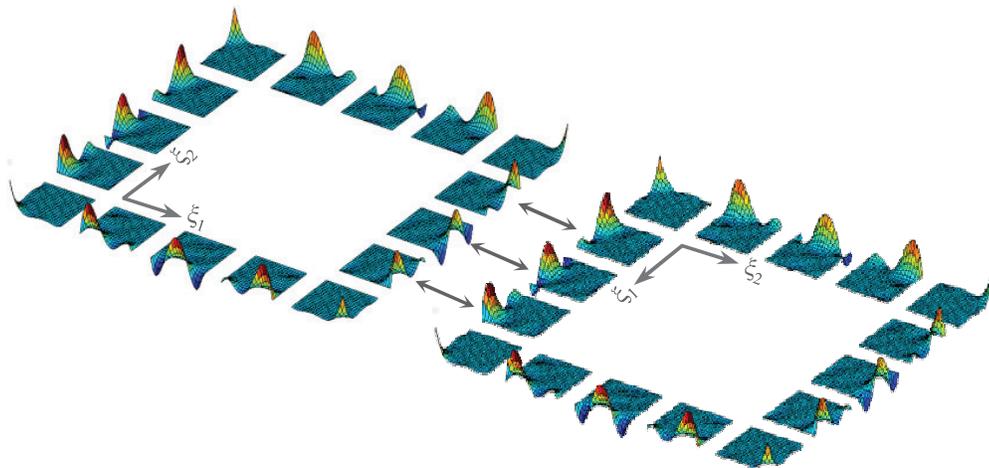


Figura 2.23: Expansão *nodal* de grau $P = 4$. Elemento 2 com giro de 270° em relação ao elemento 1.

Capítulo 3

Estrutura do Código Matlab para o MEF de Alta Ordem

3.1 Programa hp^2 FEM

O hp^2 FEM é um programa de elementos finitos, onde hp representa ao mesmo tempo o uso de funções de forma dos tipos h e p , assim como *high-performance*. O programa é formado por um conjunto de rotinas e estruturas escritas em MatLab (*Matrix Laboratory*), que torna possível a leitura e manipulação de um modelo de elementos finitos, além de permitir a resolução de diversos problemas presentes na engenharia estrutural.

Por ser extremamente versátil, optou-se em utilizar o MatLab como plataforma de desenvolvimento inicial do hp^2 FEM. Posteriormente, objetiva-se migrá-lo para a linguagem C++. O MatLab é um *software* interativo e pode ser utilizado como uma linguagem de programação interpretada. Nele os códigos são testados com mais praticidade e rapidez. Outro ponto importante desse ambiente, como seu próprio nome sugere, é a sua estrutura matricial, a qual se enquadra muito bem com a estrutura também matricial do MEF. Além disso o MatLab possui conjuntos específicos de rotinas para vários tipos de cálculos matemáticos, além de uma estrutura de visualização $2D$ e $3D$, importantes no desenvolvimento inicial.

Uma característica importante do hp^2 FEM, nessa versão em MatLab, é a substituição do comando de repetição *for* por rotinas ditas vetorizadas. Essa substituição, além de deixar

os códigos pequenos e simples, os tornam mais eficientes. Um exemplo são os procedimentos de tensorização das funções, todos feitos de forma vetorizada.

Apresenta-se a seguir uma comparação simples para ilustrar a eficiência do processo de vetorização do código em MatLab. Considere um pequeno código escrito na forma convencional

```
A = magic(Dim);
B = pascal(Dim);
for j = 1:Dim
    for k = 1:Dim;
        X(j,k) = sqrt(A(j,k)) * (B(j,k) - 1);
    end
end
```

Usando o processo de vetorização, é possível realizar o procedimento anterior em apenas três linhas de código, como se segue

```
A = magic(Dim);
B = pascal(Dim);
X = sqrt(A).*(B-1);
```

O tempo médio de execução, em função da dimensão das matrizes, para os dois códigos apresentados, é mostrado na Figura 3.1. Note que inicialmente ambos apresentam desempenho semelhante (até \approx Dim = 200). Entretanto, com o aumento da dimensão das matrizes, o código vetorizado torna-se muito mais eficiente, chegando a apresentar desempenho 500% superior, em relação ao código convencional, para Dim = 1000. Esse é um aspecto importante para um código de alta ordem, onde as matrizes geralmente são grandes e densas.

Até o momento, o código tem apresentado um bom desempenho para os testes realizados. Acredita-se que, com a paralelização de alguns procedimentos, seu desempenho venha a ser bastante razoável mesmo em MatLab. Na versão atual estão implementados elementos unidimensionais, quadrados, triângulos, hexaedros e tetraedros. Consideram-se os problemas de projeção, Poisson, elasticidade linear, grandes deformações, contato e equação de Reynolds.

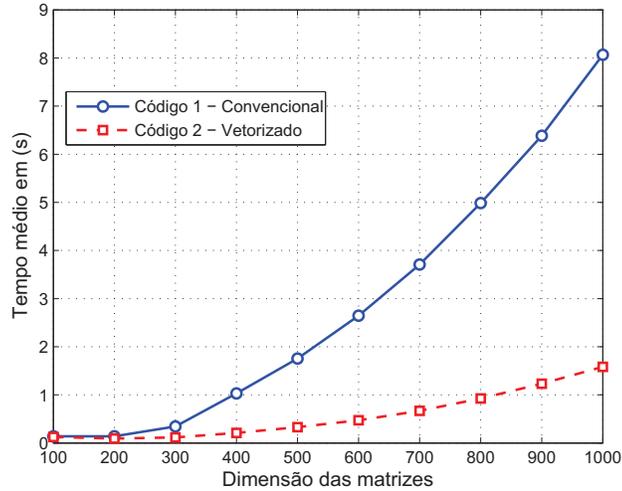


Figura 3.1: Comparação entre o código convencional e vetorizado.

3.1.1 Organização e Funcionamento

O programa funciona através da leitura de dois arquivos ASCII. O primeiro arquivo, com extensão `.fem`, contém as definições da malha de elementos finitos, tais como dimensão do modelo, número de nós, coordenadas do nós, tipo de elementos, número de elementos, grupo de elementos e topologia geometria-malha. O segundo arquivo, de extensão `.def`, contém parâmetros que definem o tipo de aplicação, condições de contorno, carregamentos, regras de integração, funções de interpolação e outros.

As rotinas `ReadFEMFile` e `ReadEFFile` são responsáveis pela leitura desses arquivos e pela interpretação e armazenamento das informações em variáveis pertinentes para a criação do modelo de elementos finitos. Essas informações são armazenadas em estruturas de dados que funcionam como variáveis globais, permitindo assim seu acesso e alteração em qualquer subrotina do programa. Após a leitura dos arquivos de entrada, o programa `hp2FEM` chama as rotinas responsáveis pela criação do modelo de elementos finitos, resolução do problema e pós-processamento dos resultados. A Figura 3.2 mostra o esquema de funcionamento do programa com destaque para o módulo de alta ordem.

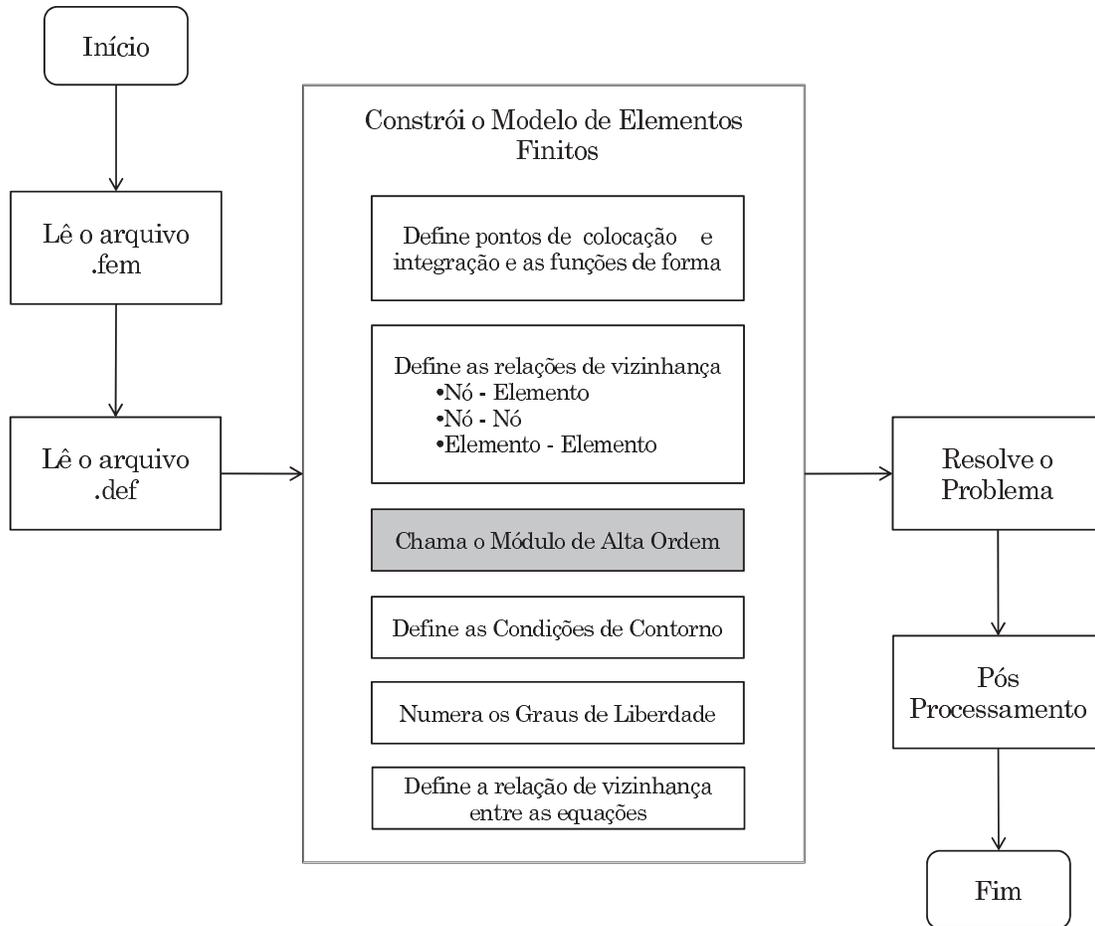


Figura 3.2: Funcionamento geral do hp^2 FEM.

3.1.2 Principais Estruturas

Nessa sessão, apresentam-se as principais estruturas do hp^2 FEM. Associam-se os atributos do modelo de elementos finitos às seguintes estruturas de armazenamento:

Model: armazena informações sobre o modelo de elementos finitos: título, graus de liberdade, quantidade de grupos de elementos, tipo de problema analisado, coordenadas nodais, parâmetros de solução, tipo de malha, número total de nós de vértice, aresta, face e corpo, dentre outros.

- ElementGroups:** armazena informações de cada grupo de elementos finitos. Um grupo consiste em um conjunto de elementos de mesma forma, modelo mecânico, função de interpolação, regra de integração e propriedades geométricas. As informações guardadas nesta estrutura são: incidência dos elementos, número de nós dos elementos, propriedades mecânicas do material, tipo de elemento do grupo, propriedade geométrica, tipo de expansão (modal ou nodal), tipo de base e grau polinomial, número de nós de vértice, aresta, face e corpo, dentre outros.
- MaterialGroups:** armazena o número do material, o seu tipo e os valores das propriedades.
- ElimDOF:** armazena informações sobre as condições de contorno de Dirichlet.
- PrescDOF:** armazena informações sobre as condições de contorno prescritas.
- GeomProp:** armazena informações sobre as propriedades geométricas dos elementos, tais como área da seção e espessura.
- ConcLoad:** armazena dados relativos às cargas nodais aplicadas na malha.
- SurfaceLoad:** armazena dados das cargas de superfície aplicadas sobre as arestas e faces dos elementos. O carregamento pode estar nas direções normal e tangencial e a intensidade ser representada por um valor numérico ou expressão simbólica.
- BodyLoad:** armazena dados das cargas de corpo aplicadas sobre um grupo de elementos. A intensidade pode ser representada por um valor numérico ou expressão simbólica.

As seguintes estruturas armazenam informações sobre como as entidades de malha se relacionam com as entidades geométricas. Essas informações são opcionais para os casos onde se deseja transferir condições de contorno aplicadas ao modelo sólido para nós e elementos da malha.

- Line:** armazena o conjunto de nós que estão contidos nas linhas do modelo geométrico. A estrutura Line é utilizada quando se deseja obter informação topológica, tal como, converter uma carga distribuída em uma linha do modelo geométrico em valores nodais da malha.
- Surface:** armazena o conjunto de nós que pertencem a uma determinada superfície geométrica da malha do modelo tridimensional.
- SurfaceGroups:** armazena para cada superfície geométrica tridimensional o número de faces de elementos finitos geradas nessa superfície, a forma e a incidência das faces.

Algumas estruturas são necessárias durante o processo de solução e geradas internamente logo após a leitura dos dados. São elas:

- ElementNodeTable:** armazena a informação de vizinhança entre elemento/nó, ou seja, dado um nó quais os elementos que o compartilham.
- NodeNodeTable:** armazena a informação de vizinhança entre os nós. Dado um nó quais são os seus nós vizinhos. Essa informação é necessária na montagem da matriz global.
- ElementElementTable:** armazena a informação de vizinhança entre elemento/elemento. Para cada elemento, o número dos elementos vizinhos e a aresta ou face compartilhada pelos vizinhos.
- EquationEquationTable:** tabela de relação de vizinhança equação-equação, isto é, dada uma equação (que corresponde a um grau de liberdade) é possível dizer quais equações (graus-de-liberdade) são influenciados por esta equação.
- Equations:** armazena a numeração dos graus de liberdade dos nós do modelo, levando em conta as condições de contorno.

Interpolation: gerencia o cálculo das coordenadas dos pontos de integração, ponderações e pontos de colocação, funções de forma e suas derivadas e a tensorização dessas entidades para elementos bi e tridimensionais.

Index: armazena os índices utilizados na tensorização das funções, assim como as incidências locais de aresta e face dos elementos.

O programa implementa *solvers* globais e elemento por elemento. Para um solver global, emprega-se a estrutura de matriz esparsa disponível no MatLab. Armazena-se, para cada coeficiente da matriz, os números da linha e coluna, assim como o valor. A matriz esparsa é construída pela função nativa do MatLab chamada `sparse`. A topologia da matriz esparsa é construída pelo programa usando a tabela **EquationEquationTable**.

A seguir, os procedimentos desenvolvidos para o MEF de Alta Ordem são detalhados.

3.2 Construção das Malhas de Alta Ordem

Uma das características do hp^2 FEM é que todas as bases bi e tridimensionais são obtidas pelo produto tensorial das funções unidimensionais, através dos índices p , q , r e s , como apresentado no Capítulo 2. Os índices são definidos no início do programa e são armazenados na tabela **Index**. Estão organizados em relação às entidades topológicas, ou seja, vértices, arestas, faces e corpo.

Para todos os elementos implementados, o programa hp^2 FEM faz somente a leitura da incidência, e das coordenadas, dos nós de vértice. Para expansões de ordem $P > 1$, o módulo de alta ordem é chamado, sendo responsável pela geração de nós adicionais. O número total de nós do elemento deve ser igual ao número de funções necessárias para uma expansão completa, de acordo com a ordem P escolhida. A Tabela 3.1 traz um resumo do número de funções, divididas em vértice, aresta, face e corpo, necessárias para cada tipo de elemento de acordo com a ordem P .

Os nós adicionais são gerados no referencial local do elemento de acordo com o tipo de quadratura escolhida para os pontos de colocação. As coordenadas globais desses nós são

Elemento	Vértice	Aresta	Face	Corpo
Quadrado	4	$4(P - 1)$	$(P - 1)^2$	
Hexaedro	8	$12(P - 1)$	$6(P - 1)^2$	$(P - 1)^3$
Triângulo	3	$3(P - 1)$	$\frac{1}{2}(P - 1)(P - 2)$	
Tetraedro	4	$6(P - 1)$	$2(P - 1)(P - 2)$	$\frac{1}{2}(P - 1)(P - 2)(P - 3)$

Tabela 3.1: Número de funções para cada entidade topológica.

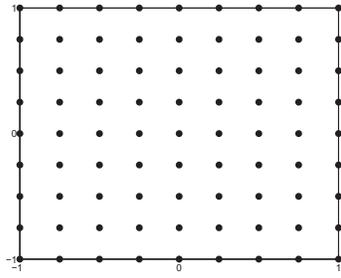
obtidas através de uma interpolação utilizando as funções de forma lineares dos vértices. Os pontos de colocação podem ser determinados a partir das seguintes regras de quadratura implementadas (Karniadakis e Sherwin, 1999):

- Lagrange Iguamente Espaçado;
- Gauss Legendre;
- Gauss Lobatto Legendre;
- Gauss Radau Legendre.
- Gauss Jacobi;
- Gauss Lobatto Jacobi;
- Gauss Radau Jacobi.

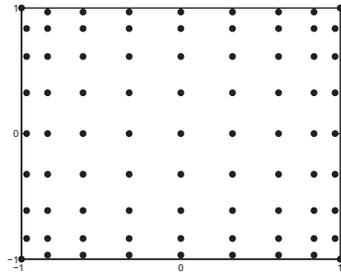
Para as quadraturas que envolvem polinômios de Jacobi são requeridos os pesos α e β desses polinômios. As Figuras 3.3 e 3.4 mostram os pontos de colocação para quadrados e triângulos de acordo com a regra de quadratura para uma expansão de grau $P = 8$. Esses pontos foram obtidos com a tensorização dos pontos unidimensionais.

O funcionamento do módulo de alta ordem é simples. O programa hp^2 FEM, de acordo com a ordem da expansão escolhida, chama o módulo de alta responsável pela geração dos nós adicionais. Os nós são gerados elemento por elemento e a incidência do elementos e dos vizinhos são simultaneamente atualizadas. A numeração dos nós adicionais segue seqüencial a partir do ultimo nó de vértice.

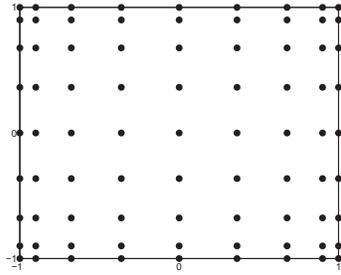
A Figura 3.5 mostra o funcionamento do módulo de alta ordem.



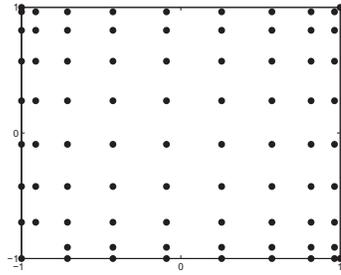
(a) Lagrange Iguamente Espaçado.



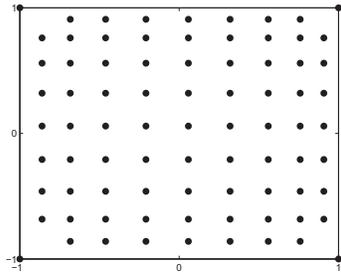
(b) Gauss Legendre.



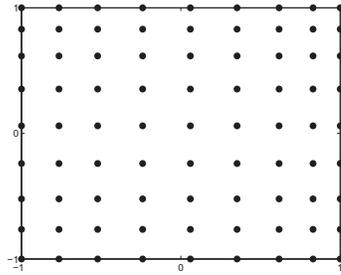
(c) Gauss Lobatto Legendre.



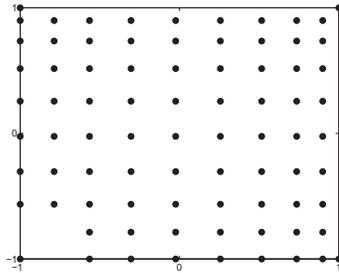
(d) Gauss Radau Legendre.



(e) Gauss Jacobi, $\alpha = 2, \beta = 3$.

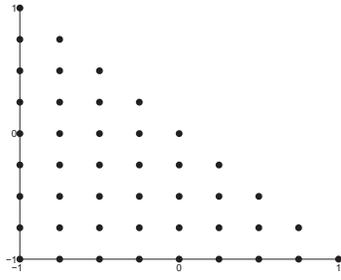


(f) Gauss Lobatto Jacobi, $\alpha = 2, \beta = 3$.

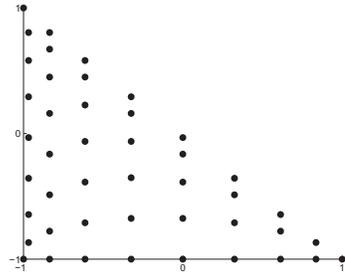


(g) Gauss Radau Jacobi, $\alpha = 2, \beta = 3$.

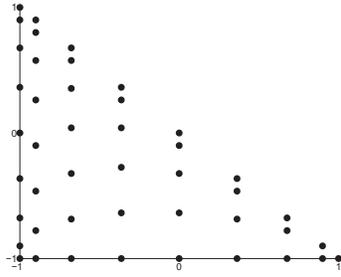
Figura 3.3: Pontos de colocação para o quadrado de acordo com a regra de quadratura para uma expansão de grau $P = 8$.



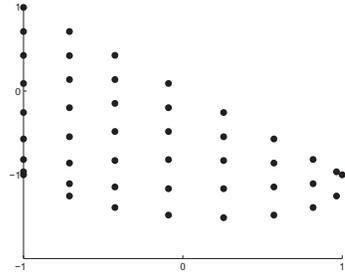
(a) Lagrange Iguamente Espaçado.



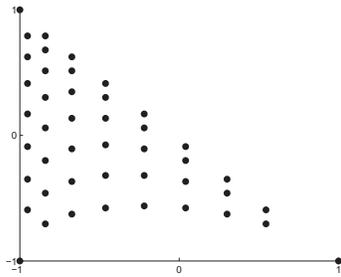
(b) Gauss Legendre.



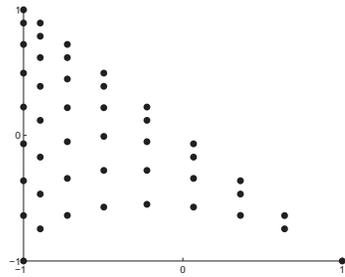
(c) Gauss Lobatto Legendre.



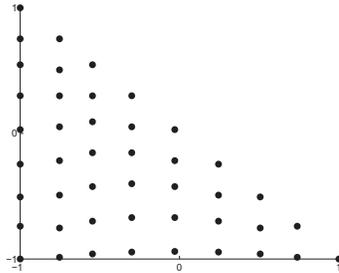
(d) Gauss Radau Legendre.



(e) Gauss Jacobi, $\alpha = 5, \beta = 1$.



(f) Gauss Lobatto Jacobi, $\alpha = 5, \beta = 1$.



(g) Gauss Radau Jacobi, $\alpha = 5, \beta = 1$.

Figura 3.4: Pontos de colocação para o triângulo de acordo com a regra de quadratura para uma expansão de grau $P = 8$.

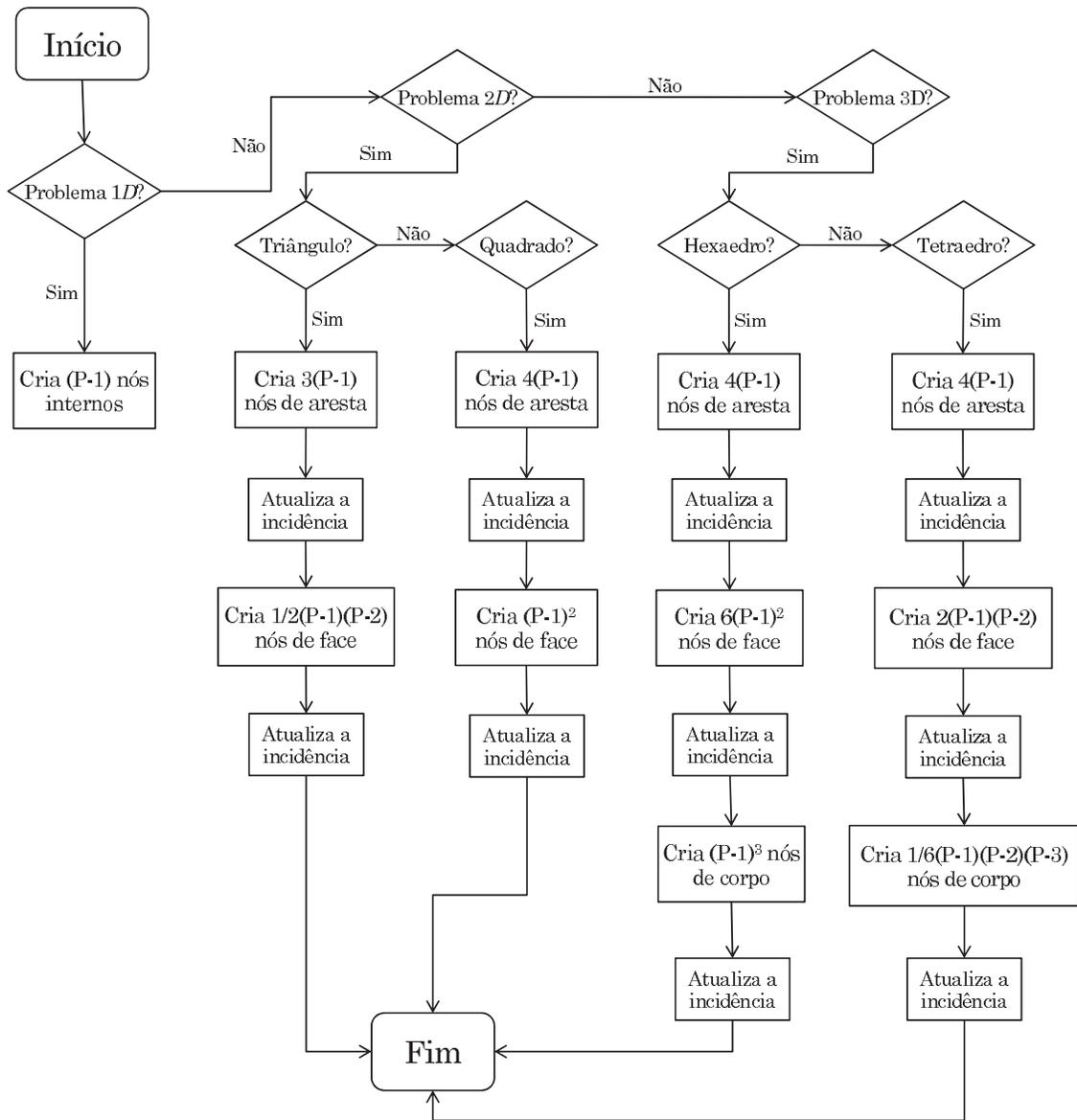


Figura 3.5: Funcionamento do módulo de alta ordem.

3.3 Detalhamento dos Elementos

3.3.1 Elementos Bidimensionais - Quadrados e Triângulos

No hp^2 FEM, quadrados e triângulos são os elementos usados na modelagem de problemas $2D$, identificados, respectivamente, pelas palavras chaves, **SQUARE** e **TRIANGLE**. A Figura 3.6 mostra a numeração local das arestas para os dois elementos. As malhas $2D$ podem ser formadas por um único tipo de elemento, como podem ser também híbridas.

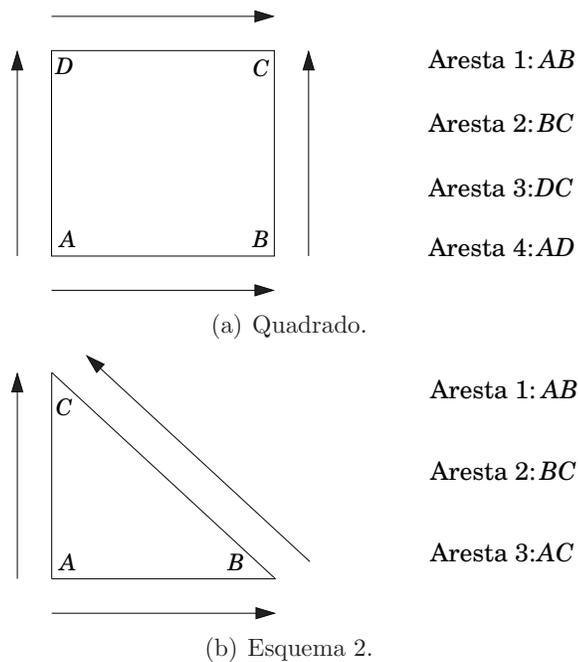
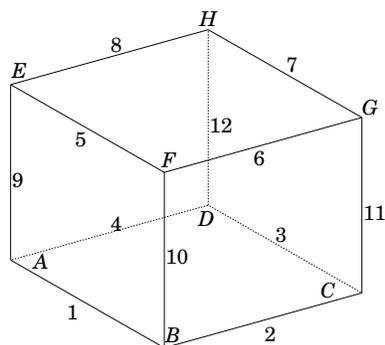


Figura 3.6: Numeração das arestas dos elementos bidimensionais.

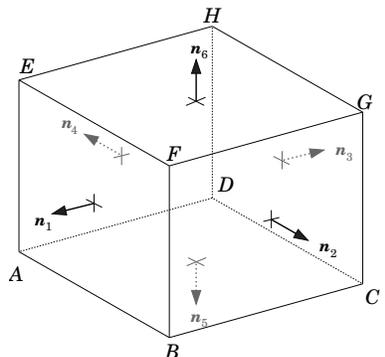
3.3.2 Elementos Tridimensionais - Hexaedros e Tetraedros

O hexaedro e o tetraedro são usados na modelagem de problemas $3D$. No programa hp^2 FEM o hexaedro é identificado pela palavra chave **HEXAHEDRON** e tetraedro por **TETRAHEDRON**. As Figuras 3.7 à 3.10 mostram a numeração das entidades topológicas dos elementos tridimensionais. A incidência das faces é dada de forma que os vetores normais à cada face apontem para fora do elemento.



- | | |
|----------------|-----------------|
| ----- | ----- |
| Aresta 1: AB | Aresta 7: HG |
| ----- | ----- |
| Aresta 2: BC | Aresta 8: EH |
| ----- | ----- |
| Aresta 3: DC | Aresta 9: AE |
| ----- | ----- |
| Aresta 4: AD | Aresta 10: BF |
| ----- | ----- |
| Aresta 5: EF | Aresta 11: CG |
| ----- | ----- |
| Aresta 6: FG | Aresta 12: DH |

Figura 3.7: Hexaedro: numeração local das arestas.



- | | | | | | | | | | |
|--|-----|-----|-----|-----|--|-----|-----|-----|-----|
| <table border="1"> <tr><td>E</td><td>F</td></tr> <tr><td>A</td><td>B</td></tr> </table> <p>Face 1: $ABFE$</p> | E | F | A | B | <table border="1"> <tr><td>F</td><td>G</td></tr> <tr><td>B</td><td>C</td></tr> </table> <p>Face 2: $BCGF$</p> | F | G | B | C |
| E | F | | | | | | | | |
| A | B | | | | | | | | |
| F | G | | | | | | | | |
| B | C | | | | | | | | |
| <table border="1"> <tr><td>G</td><td>H</td></tr> <tr><td>C</td><td>D</td></tr> </table> <p>Face 3: $CDHG$</p> | G | H | C | D | <table border="1"> <tr><td>E</td><td>H</td></tr> <tr><td>A</td><td>D</td></tr> </table> <p>Face 4: $DAEH$</p> | E | H | A | D |
| G | H | | | | | | | | |
| C | D | | | | | | | | |
| E | H | | | | | | | | |
| A | D | | | | | | | | |
| <table border="1"> <tr><td>B</td><td>C</td></tr> <tr><td>A</td><td>D</td></tr> </table> <p>Face 5: $ADCB$</p> | B | C | A | D | <table border="1"> <tr><td>H</td><td>G</td></tr> <tr><td>E</td><td>F</td></tr> </table> <p>Face 6: $EFGH$</p> | H | G | E | F |
| B | C | | | | | | | | |
| A | D | | | | | | | | |
| H | G | | | | | | | | |
| E | F | | | | | | | | |

Figura 3.8: Hexaedro: numeração local das faces.

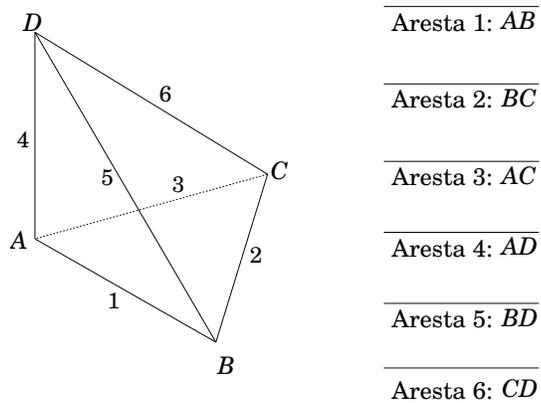


Figura 3.9: Tetraedro: numeração local das arestas.

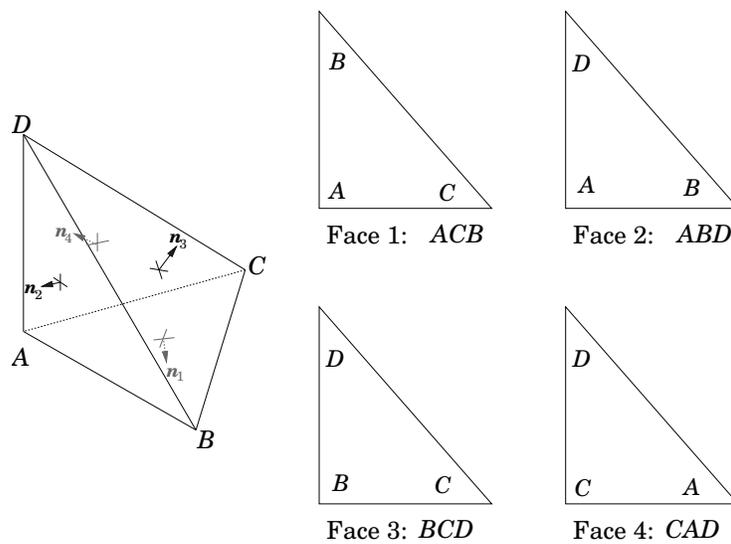


Figura 3.10: Tetraedro: numeração local das faces.

Capítulo 4

Aplicações

A convergência da versão h do MEF foi muito bem estabelecida pela teoria matemática na década de 1970. Sabe-se que o erro na solução numérica decai algebricamente com o refinamento da malha (introdução de mais elementos). No refinamento tipo p , o erro na solução numérica é reduzido com o aumento da ordem da interpolação polinomial mantendo-se fixo o número de elementos. Para soluções suaves, esse tipo de refinamento leva usualmente a um decaimento exponencial do erro (Karniadakis e Sherwin, 1999).

Nesse capítulo, apresentam-se alguns resultados obtidos com o código desenvolvido. Inicialmente apresenta-se o desenvolvimento matemático dos problemas de projeção e Poisson. Em seguida, por meio da aplicação de uma solução analítica, esses problemas são analisados em malhas $2D$ e $3D$ com refinamentos h e p . Esses problemas foram usados durante a fase de desenvolvimento do código, pois permitem a validação das matrizes de massa e rigidez. São definidas seqüências de malhas h de quadrados, triângulos e hexaedros. Os resultados são mostrados por meio de gráficos do erro relativo na norma de energia pelo número de graus de liberdade do problema. Objetiva-se a identificação do tipo de convergência para cada tipo de refinamento.

4.1 Caracterização dos Problemas de Aplicação

4.1.1 Problema de Projeção

No MEF, uma função contínua u é aproximada por uma combinação linear de funções como

$$u \approx u_{ap} = \sum_{i=0}^n u_i \phi_i + \sum_{j=0}^m c_j \psi_j, \quad (4.1)$$

onde u_i são os valores de u_{ap} nos nós dos elementos, ϕ_i são as funções de interpolação nodais, c_j são os coeficientes que não estão associados aos nós e ψ_j são funções que não estão relacionadas aos nós, conhecidos também por modos. A expressão (4.1) é a definição geral para aproximação de uma função no MEF. Em expansões estritamente nodais, onde todas as funções estão diretamente relacionadas aos nós, o último termo da expressão (4.1) é nulo. Para esse caso escreve-se,

$$u \approx u_{ap} = \sum_{i=0}^n u_i N_i, \quad (4.2)$$

onde N_i são as funções de forma.

As funções $\{N_i\}$ definem um espaço funcional no qual se encontra a solução aproximada, visto que u_{ap} é uma combinação linear de $\{N_i\}$ (expressão (4.2)). Esse espaço funcional pode ser ilustrado por um plano (Figura 4.1). O erro na aproximação é dado pela diferença entre u e u_{ap} , ou seja,

$$e = u - u_{ap}. \quad (4.3)$$

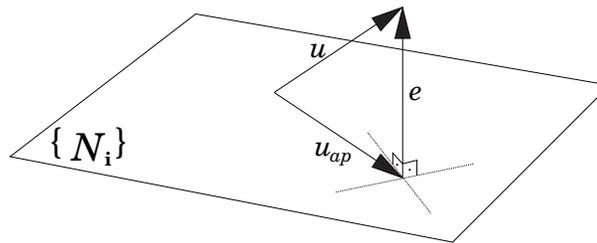


Figura 4.1: Problema de projeção.

O erro pode ser geometricamente entendido como a distância entre a função u e o plano $\{N_i\}$. Sabe-se que a menor distância entre um ponto e um plano é dada pelo vetor que passa pelo ponto e é ortogonal ao plano. Por isso, o erro será mínimo quando e for ortogonal ao

plano $\{N_i\}$, ou seja,

$$\int_{\Omega} e N_i d\Omega = 0. \quad (4.4)$$

Com a substituição da expressão (4.3) na (4.4), obtém-se

$$\int_{\Omega} (u - u_{ap}) N_i d\Omega = 0, \quad (4.5)$$

ou seja,

$$\int_{\Omega} u_{ap} N_i d\Omega = \int_{\Omega} u N_i d\Omega. \quad (4.6)$$

Substituindo (4.2) em (4.6) tem-se,

$$\begin{aligned} \int_{\Omega} \left(\sum_{i=1}^n u_i N_i \right) N_j d\Omega &= \int_{\Omega} u N_j d\Omega, \\ \sum_{i=1}^n u_i \int_{\Omega} N_i N_j d\Omega &= \int_{\Omega} u N_j d\Omega, \quad j = 1, 2, \dots, n. \end{aligned} \quad (4.7)$$

A equação anterior pode ser escrita como

$$\sum_{i=1}^n M_{ij} u_i = b_j, \quad (4.8)$$

sendo

$$M_{ij} = \int_{\Omega} N_i N_j d\Omega$$

e

$$b_j = \int_{\Omega} u N_j d\Omega,$$

onde M_{ij} são dos coeficientes da matriz de massa e b_j os elementos do vetor de carga de corpo.

4.1.2 Problema de Poisson

Considere o Problema de Poisson com condições de contorno de Dirichlet homogêneas definido no domínio Ω

$$\begin{aligned} \nabla^2 u &= f, \quad \Omega, \\ u &= 0, \quad \partial\Omega, \end{aligned} \quad (4.9)$$

sendo f a função de carregamento. Escreve-se a forma fraca, baseada no Método dos Resíduos Ponderados, como

$$\int_{\Omega} \nabla u \cdot \nabla v d\Omega = \int_{\Omega} f v d\Omega + \int_{\Gamma} (\nabla u \cdot \mathbf{n}) v d\Gamma, \quad (4.10)$$

onde v é a função teste e \mathbf{n} o campo vetorial normal à cada ponto do contorno. Como $u = 0$ em todo o contorno Γ , o último termo do lado direito da equação (4.10) é nulo. Aplicando a aproximação de Galerkin, onde $v = N_i$, com n funções de interpolação, tem-se

$$\int_{\Omega} \nabla \left(\sum_{i=1}^n u_i N_i \right) \cdot \nabla N_j d\Omega = \int_{\Omega} f N_j d\Omega, \quad (4.11)$$

ou ainda,

$$\sum_{i=1}^n \left(\int_{\Omega} \nabla N_i \cdot \nabla N_j d\Omega \right) \cdot u_i = \int_{\Omega} f N_j d\Omega, \quad j = 1, 2, \dots, n. \quad (4.12)$$

A equação anterior pode ser escrita como

$$\sum_{i=1}^n K_{ij} u_i = b_j, \quad (4.13)$$

sendo os coeficientes da matriz de rigidez K_{ij} e do vetor de carga de corpo b_j dados por

$$K_{ij} = \int_{\Omega} \nabla N_i \cdot \nabla N_j d\Omega$$

e

$$b_j = \int_{\Omega} f N_j d\Omega.$$

4.1.3 Erro na Norma de Energia

A norma de energia para a solução aproximada u_{ap} pode ser calculada por meio dos valores de u_i , obtidos na solução das equações (4.8) e (4.13).

Para o problema de projeção, a norma de energia pode ser calculada por

$$\|u_{ap}\|^2 = \{u_i\}^T [M] \{u_i\}, \quad (4.14)$$

onde $[M]$ é a matriz de massa global.

Para o problema de Poisson, a norma de energia é calculada por meio da matriz de rigidez global $[K]$. Escreve-se

$$\|u_{ap}\|^2 = \{u_i\}^T [K] \{u_i\}. \quad (4.15)$$

A norma de energia para a solução analítica imposta pode ser calculada da seguinte maneira

$$\|u\|^2 = \int_{\Omega} (\nabla u)^2 d\Omega. \quad (4.16)$$

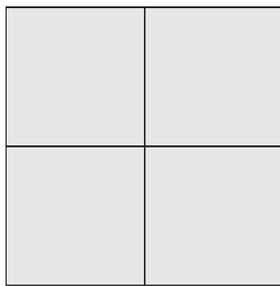
O erro relativo na norma de energia, para ambos os problemas, fica dada por

$$\|e\|^2 = \frac{\|u - u_{ap}\|^2}{\|u\|^2}. \quad (4.17)$$

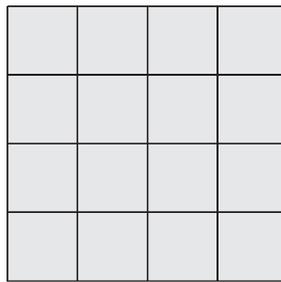
4.2 Resultados

4.2.1 Malhas Consideradas

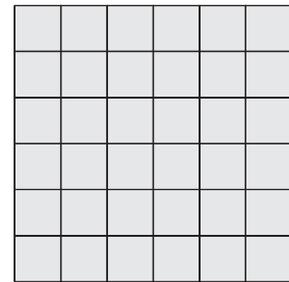
Os problemas $2D$ foram analisados considerando-se um domínio quadrado Ω de lado igual a quatro. O domínio foi discretizado em malhas de quadrados (Figuras 4.2 a 4.4) e triângulos (Figuras 4.5 e 4.6). Para os problemas $3D$ foi considerado o paralelepípedo definido em $\Omega = \{(x, y, z) | 0 \leq x \leq 4, 0 \leq y, z \leq 2\}$, discretizado em malhas de hexaedros uniformes (Figura 4.7).



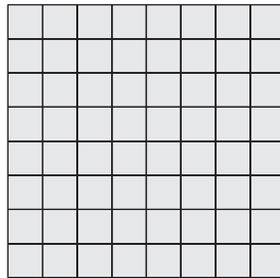
(a) 4 elementos; 9 nós.



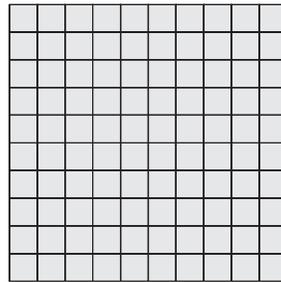
(b) 16 elementos; 25 nós.



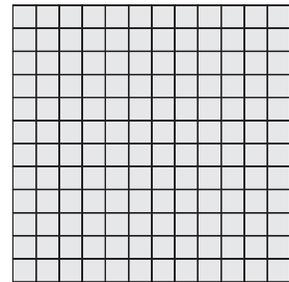
(c) 36 elementos; 49 nós.



(d) 64 elementos; 81 nós.



(e) 100 elementos; 121 nós.



(f) 144 elementos; 169 nós.

Figura 4.2: Malhas de quadrados uniformes.

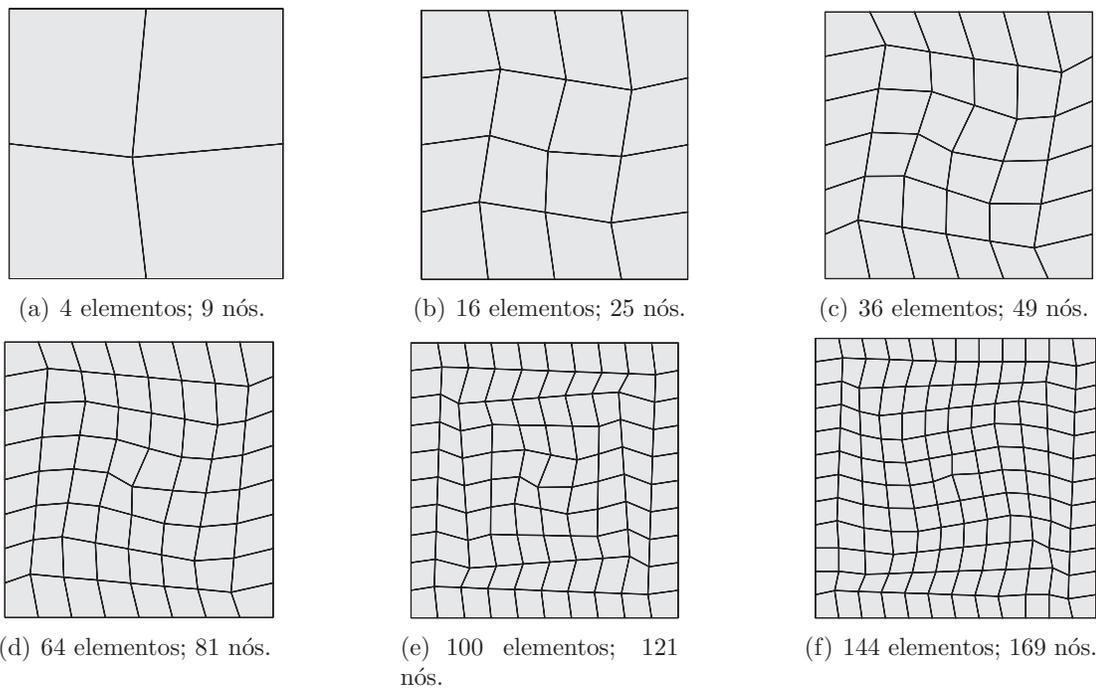


Figura 4.3: Malhas de quadrados distorcidas.

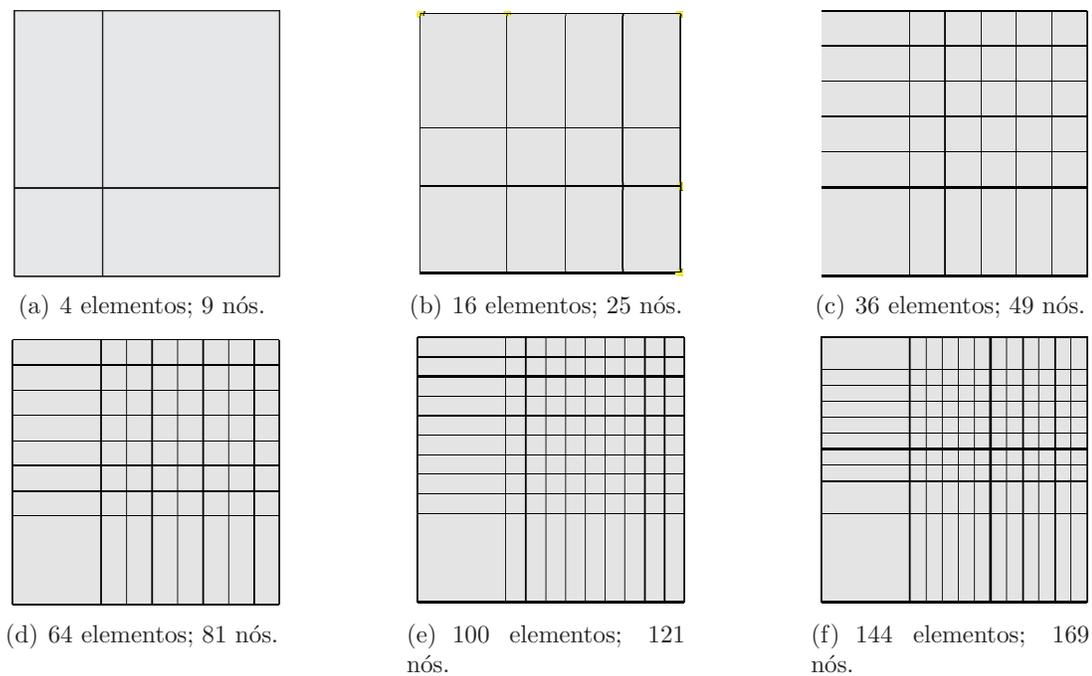
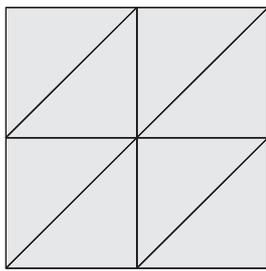
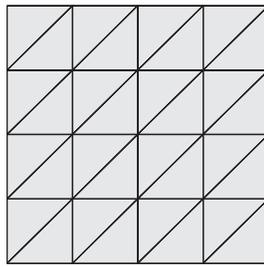


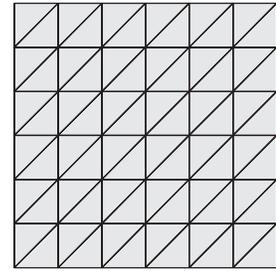
Figura 4.4: Malhas de quadrados com refinamento localizado.



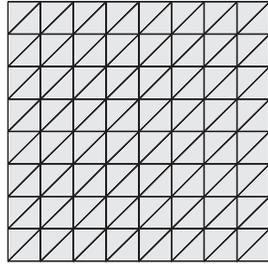
(a) 8 elementos; 9 nós.



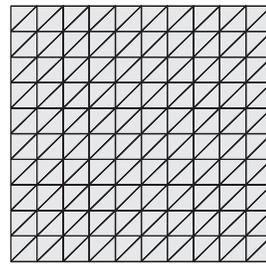
(b) 32 elementos; 25 nós.



(c) 72 elementos; 49 nós.

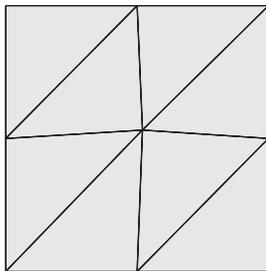


(d) 128 elementos; 81 nós.

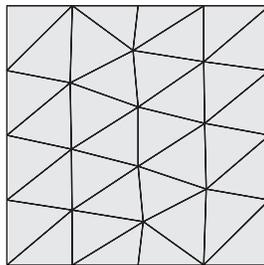


(e) 200 elementos; 121 nós.

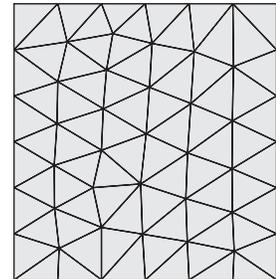
Figura 4.5: Malhas de triângulos aninhadas.



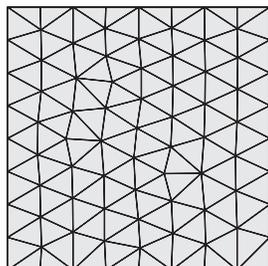
(a) 8 elementos; 9 nós.



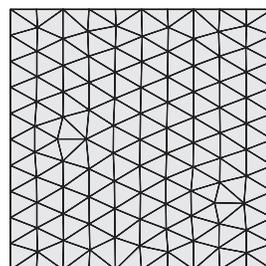
(b) 34 elementos; 26 nós.



(c) 76 elementos; 51 nós.



(d) 136 elementos; 85 nós.



(e) 208 elementos; 125 nós.

Figura 4.6: Malhas de triângulos não-aninhadas.

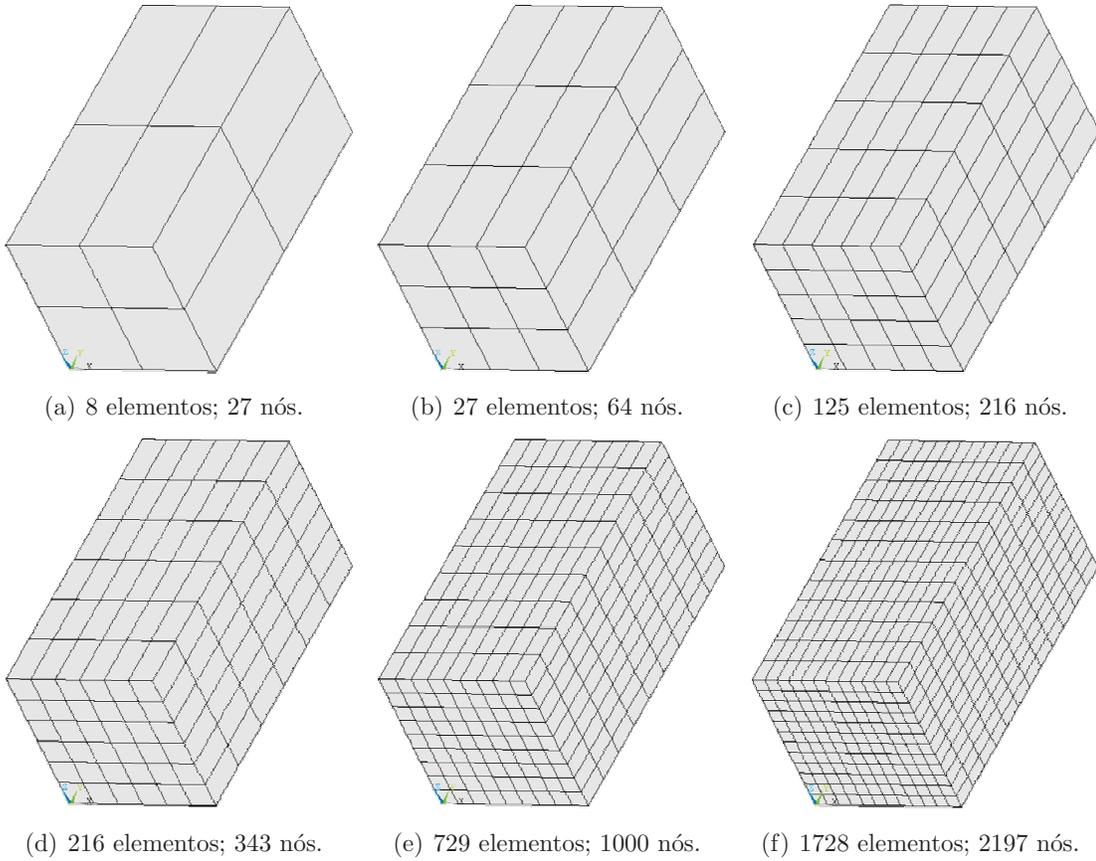


Figura 4.7: Malhas de hexaedros.

4.2.2 Problema de Projeção 2D

Para o problema de projeção 2D, a solução imposta foi $u(x, y) = (x^6 - 4x^5)(y^6 - 4y^5)$ (Figura 4.8). O domínio foi discretizado em quadrados e triângulos, conforme as malhas definidas anteriormente nas Figuras 4.2 a 4.6.

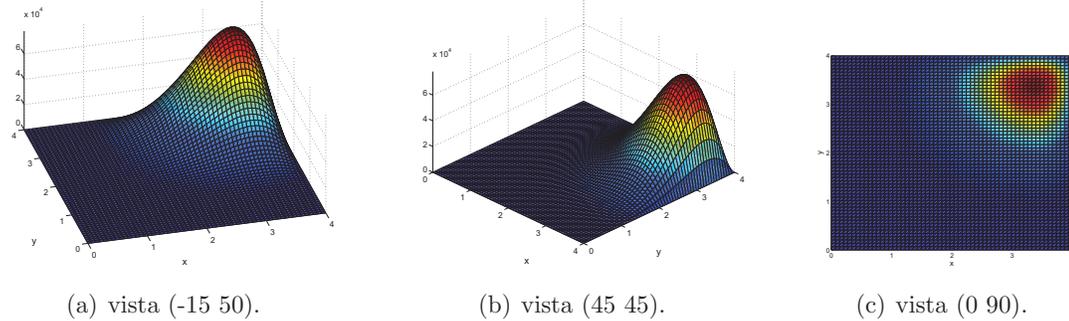


Figura 4.8: Solução imposta para o problema de projeção 2D

O erro na norma de energia, em função do número de graus de liberdade e com refinamentos h e p , é mostrado na Figura 4.9. As curvas relativas ao refinamento h foram obtidas com ordem polinomial fixa $P = 1$ e os refinamentos p utilizam base de Lagrange. As Figuras 4.9(a) e 4.9(b) mostram a convergência da solução para as malhas de quadrados e triângulos, respectivamente. A Figura 4.9(c) apresenta as curvas de convergência para quadrados uniformes e triângulos aninhados.

Pela Figura 4.9(a), nota-se que a distorção da malha interfere na qualidade da solução. A convergência do refinamento p nas malhas uniformes foi melhor. O refinamento h não apresentou diferença.

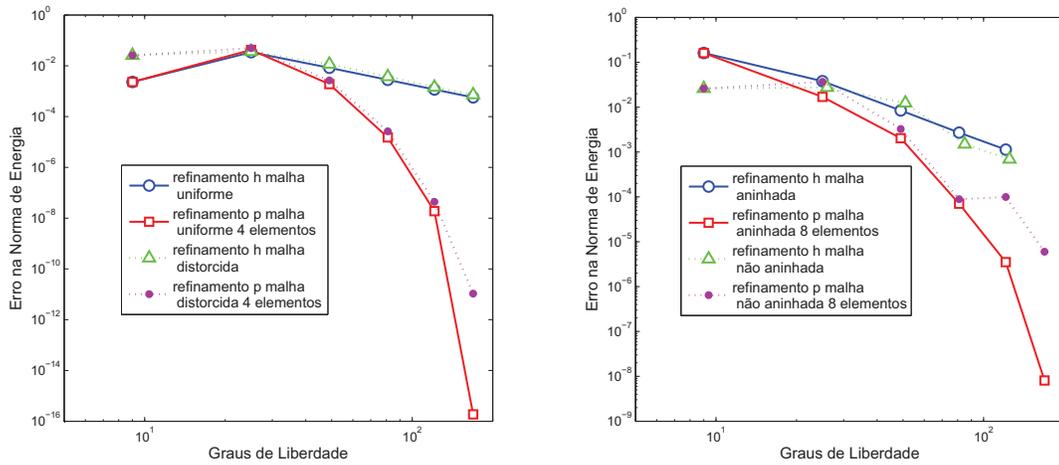
Para triângulos, as malhas aninhadas (Figura 4.5) apresentam um melhor perfil de convergência quando comparadas às malhas não aninhadas (Figura 4.6). Logo, a forma dos elementos interfere na solução.

Nota-se pela Figura 4.9(c) que o refinamento p na malha de quadrados uniformes apresentou-se mais eficiente que o mesmo refinamento em malhas de triângulos.

Visando identificar os efeitos de um refinamento mais localizado, a seqüência de malhas da Figura 4.4 foi empregada. As malhas foram refinadas na região de maior gradiente da

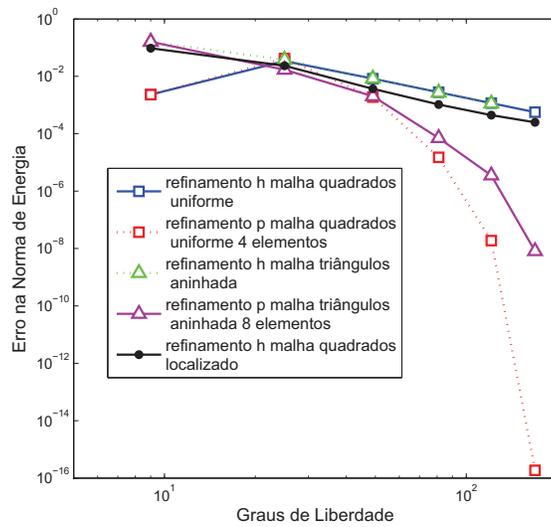
solução imposta. Nota-se pela Figura 4.9(c) que o refinamento localizado h na malha de quadrados foi mais eficiente que os refinamentos h uniformes para quadrados e triângulos. Entretanto, a convergência para o refinamento p continua sendo superior.

Como era esperado, a convergência da solução para os refinamentos p é muito mais rápida que para o refinamento h em todos os casos mostrados na Figura 4.9.



(a) Malhas de quadrados.

(b) Malhas de triângulos.



(c) Malhas de quadrados uniformes e triângulos aninhados.

Figura 4.9: Convergência da solução para o problema de projeção $2D$.

4.2.3 Problema de Projeção 3D

Para o problema de projeção 3D, a solução imposta foi $u(x, y) = (x^{10} - 2x^9)(y^2 - 4y)$. O domínio foi discretizado em hexaedros, conforme as malhas definidas anteriormente na Figura 4.7.

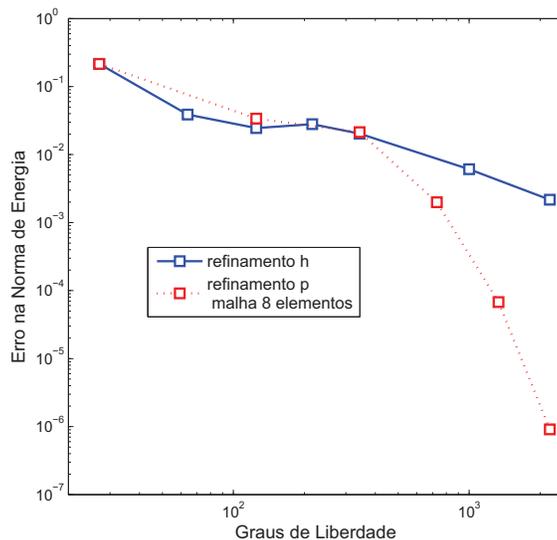


Figura 4.10: Convergência h e p para o problema de projeção 3D com malhas de hexaedros uniformes.

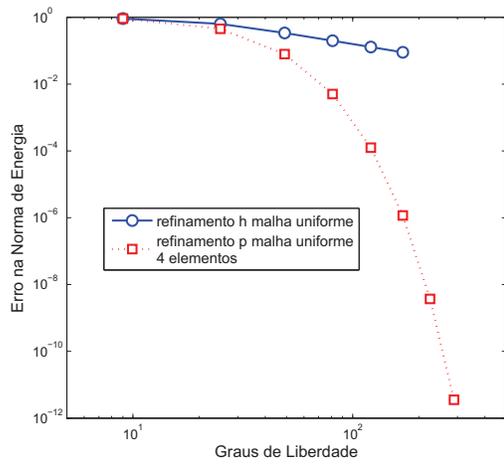
O erro na norma de energia em função do número de graus de liberdade, com refinamentos h e p , é mostrado na Figura 4.10. As curvas relativas ao refinamento h foram obtidas com ordem polinomial fixa $P = 1$ e os refinamentos p utilizam base de Lagrange.

O refinamento h inicialmente apresentou um desempenho melhor que o refinamento p . Entretanto, como o refinamento p , a convergência exponencial assintótica é obtida, superando o refinamento h .

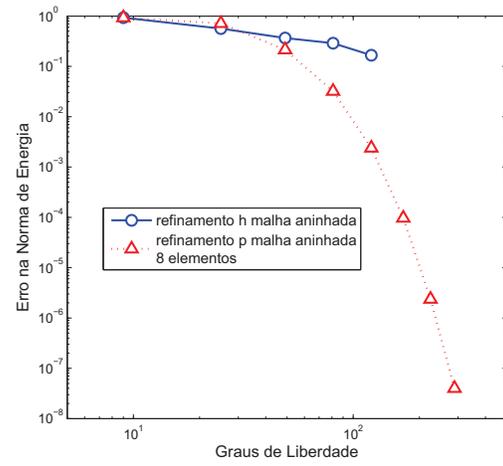
4.2.4 Problema de Poisson 2D

Para o problema de Poisson (4.9), a solução imposta foi $u(x, y) = (x^9 - 4x^8)(y^8 - 4y^7)$ e o termo independente f é dado por

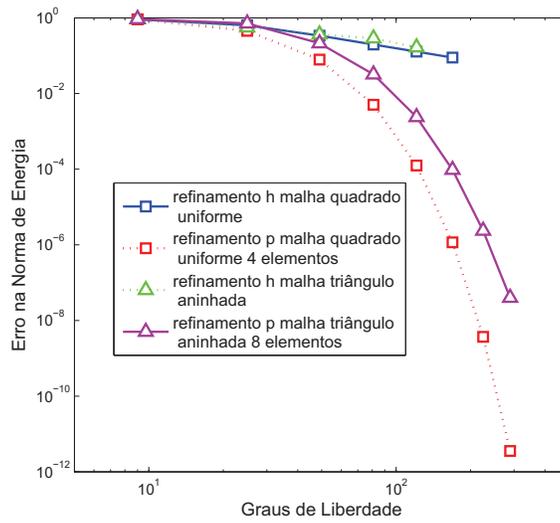
$$f(x, y) = (-72x^7 + 224x^6)(y^8 - 4y^7) - (56y^6 - 168y^5)(x^9 - 4x^8).$$



(a) Malhas de quadrados.



(b) Malhas de triângulos.



(c) Malhas de quadrados uniformes e triângulos aninhados.

Figura 4.11: Convergência da solução para o problema de Poisson 2D.

O erro na norma de energia em função do número de graus de liberdade, com refinamentos h e p , é mostrado na Figura 4.11. As curvas relativas ao refinamento h foram obtidas com ordem polinomial fixa $P = 1$ e os refinamentos p utilizam base de Lagrange. As Figuras 4.11(a) e 4.11(b) mostram a convergência da solução para as malhas de quadrados e triângulos, respectivamente. A Figura 4.11(c) apresenta as curvas de convergência para quadrados uniformes e triângulos aninhados.

Nota-se pela Figura 4.11 que o refinamento p na malha de quadrados uniformes apresentou-se mais eficiente que o mesmo refinamento em malhas de triângulos.

Como era esperado, a convergência da solução para os refinamentos p é muito mais rápida que para o refinamento h tanto em quadrados (Figura 4.11(a)) quanto em triângulos (Figura 4.11(b)).

Capítulo 5

Cosiderações Finais e Perspectivas Futuras

Procedimentos específicos para o MEF de Alta Ordem foram desenvolvidos. Na primeira parte do trabalho foram apresentado os procedimentos para construção das bases de funções baseados na tensorização das funções unidimensionais.

Discutiu-se em detalhes a continuidade global C^0 entre as funções de interpolação para quadrados. De acordo com o Capítulo 2, dois aspectos influenciam na continuidade natural das expansões modais. Primeiramente, foi analisado a influencia da ordem de numeração das funções e em seguida a influencia dos referências locais dos elementos. Concluiu-se que, para uma continuidade natural, as funções devem ser numeradas em uma ordem correta e que os referências locais devem ser coerentemente definidos. Nos códigos desenvolvidos, a ordem correta de numeração das funções é obtida automaticamente no processo de tensorização. Em relação aos referências locais, conclui-se que, deve haver uma certa relação entre os referencias locais de cada elemento, caso contrário um procedimento específico deverá ser implementado para garantia da continuidade. Nota-se pelas Figuras 2.18 e 2.19 que esse procedimento deverá trocar de posição as funções pares multiplicar por -1 as ímpares. Uma característica importante dos códigos desenvolvidos é que os nós de aresta, face e corpo são gerados internamente, permitindo um controle maior sobre a continuidade da aproximação.

No Capítulo 4, através dos problemas de projeção e Poisson $2D$ e $3D$, analisou-se a convergência da solução para os refinamentos h e p . Para o refinamento p a convergência

foi mais rápida que o refinamento h em todas as simulações, mesmo para um refinamento h localizado (Figura 4.9).

Nos capítulos iniciais desse trabalho, apresentou-se uma série de referências destacando vantagens do MEF de Alta Ordem sobre os de baixa ordem. Para uma análise da aplicabilidade do MEF de Alta Ordem em problemas práticos de engenharia, é fundamental o desenvolvimento de códigos computacionais. Com o desenvolvimento do hp^2 FEM a eficiência do método poderá ser analisada na solução de vários problemas em Mecânica dos Sólidos. Como por exemplo, as bases modais, baseadas nos polinômios de Jacobi, apresentam resultados promissores e poderão ser efetivamente aplicadas em problemas reais e sua eficiência poderá ser avaliada. Com o hp^2 FEM será possível ainda o desenvolvimento de outros algoritmos visando um melhor desempenho computacional, como por exemplo, a obtenção das matrizes de elementos 2D e 3D através da tensorização das matrizes unidimensionais, que já vêm sendo testados. Enfim, espera-se que o hp^2 FEM seja um laboratório virtual para todos os tipos de testes em elementos finitos, incluindo os elementos de alta ordem, e também um laboratório de computação de alto desempenho.

O hp^2 FEM está totalmente estruturado com conceitos de engenharia de *software*. Esse aspecto, além de ajudar na organização do código, permitiu que aparentes empecilhos da aplicação do MEF de Alta Ordem fossem superados. No código, por exemplo, todas as estruturas para geração e tensorização das funções unidimensionais e criação das matrizes dos elementos são eficientes tendo um custo computacional relativamente baixo.

Como sugestão para continuidade desse trabalho propõem-se:

- automatizar a geração dos arquivos de entrada para elementos de alta ordem. Para análises de baixa ordem, os dados de entrada do hp^2 FEM são obtidos automaticamente através de um gerador de malhas. Entretanto, para problemas de alta ordem, algumas informações adicionais são necessárias e ainda não são automaticamente obtidas, como por exemplo, a numeração local das arestas e faces de cada elemento para aplicação das condições de contorno e carregamentos;
- implementar as estruturas de visualização da solução;
- reestruturar o código usando conceitos de orientação à objetos;

- analisar a eficiência e otimizar as rotinas com o auxílio da ferramenta *profiler* do MatLab;
- desenvolver uma versão paralela do código MatLab;
- e por fim, realizar a migração para a linguagem $C++$.

Referências Bibliográficas

- Adjerid, S., Aiffa, M. , Flaherty, J. E. Hierarchical finite element bases for triangular and tetrahedral elements. *Computer Methods in Applied Mechanics and Engineering*, v. 190p.2925–2941, 1995.
- Ainsworth, M. A preconditioner based on domain decomposition for h-p finite element approximation on quasi-uniform meshes. *SIAM J. Numer. Anal.*, v. 33p.1358–1376, 1996.
- Babuska, 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.
- Babuska, I., Craig, A., Mandel, J. , Pitkaranta, J. Efficient preconditioning for the p -version finite element method in two dimensions. *SIAM Journal on Numerical Analysis*, v. 28, n.3, p.624–661, 1991.
- Babuska, 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.
- Babuska, 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.
- Babuška, I., Rheinboldt, W. Adaptive approaches and reliability estimations in finite element analysis. *Computer Methods in Applied Mechanics and Engineering.*, v. 17, 1979.
- Beskok, A., Warburton, T. C. An unstructured hp finite-element scheme for fluid flow and heat transfer in moving domains. *Journal of Computational Physics*, v. 174p.492–509, 2001.
- Beuchler, S., Schöberl, J. New shape functions for triangular p-fem using integrated jacobi polynomials. *Numerische Mathematik*, v. 103p.339–366, 2006.
- Bittencourt, M. L. *Introdução ao Método dos Elementos Finitos Aplicado à Análise Estrutural*. Campinas, 1991.
- Bittencourt, M. L. Fully tensorial nodal and modal shape functions for triangles and tetrahedra. *International Journal for Numerical Methods in Engineering*, v. 63p.1530–1558, 2005.

- Bittencourt, M. L., Vazquez, M. G. , Vazquez, T. G. Construction of shape functions dor the h- and p-versions of the fem using tensorial product. *International Journal for Numerical Methods in Engineering*, v. 71p.529–563, 2007.
- Blyth, M. G., Pozrikidis, C. A lobatto interpolation grid over the triangle. *IMA Journal of Applied Mathematics*, v. 71, 2006.
- Carnevali, P., Morris, R. B., Tsuji, Y. , Taylor, G. New basis functions and computational procedures for p -version finite element analysis. *International Journal for Numerical Methods in Engineering*, 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.
- Clough, R. The finite element method in plane stress analysis. *Proceedings 2nd ASCE Conference on Electronic Computation*, 1960.
- Clough, R. W. Thoughts about the origin of the finite element method. *Computers & Structures*, v. 79p.2029–2030, 2001.
- Cook, R., Malkus, D. , Plesha, M. *Concepts and Applications of Finite Element Analysis*. John Wiley & Sons, USA, third ed., 1991.
- Demkowicz, L., Vardapetyan, L. Modeling of electromagnetic absorption / scattering problems using hp-adaptive finite elements. *Computer Methods in Applied Mechanics and Engineering*, v. 152p.103–124, 1998.
- Dong, S., Karniadakis, G. E. Dual-level parallelism for high-order cfd methods. *Parallel Computing*, v. 30p.1–20, 2004.
- Dong, S., Yosibashi, Z. A parallel spectral element method for dynamic three-dimensional nonlinear elasticity problems. *Computers and Structures*, v. 87p.59–72, 2009.
- Dong, S. S., Karniadakis, G. E. P-refinement and p-threads. *Computer Methods in Applied Mechanics and Engineering*, v. 192p.2191–2201, 2003.
- Düster, A., Scholza, D. , Ranka, E. pq-adaptive solid finite elements for three-dimensional plates and shells. *Computer Methods in Applied Mechanics and Engineering*, v. 197p.243–254, 2007.
- Duster, A., Rank, E. The p -version of the finite element method compared to an adaptive h-version for the deformation theory of plasticity. *Computer Methods in Applied Mechanics and Engineering*, 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 & Structures*, v. 60, n.4, p.521–530, 1996.
- Ghosh, D. K., Basu, P. K. A parallel programming environment for adaptive p-version finite element analysis. *Advances in EGINEERING Software*, v. 29p.227–240, 1998.
- Guo, B., W, W. C. A preconditioner for the h-p version of the finite element method in two dimensions. *Numerische Mathematik*, v. 75p.59–77, 1996.
- Guo, B., W, W. C. An iterative and parallel solver based on domain decomposition for the hp-version of the finite element method. *J. Comput. Appl. Math.*, v. 83p.71–85, 1997.
- Holzer, S. M., Yosibashi, Z. The p-version of the finite element method in incremental elasto-plastic analysis. *International Journal for Numerical Methods in Engineering*, v. 39p.1859–1878, 1996.
- Jensen, S., Korneev, V. On domain decomposition preconditioning in the hierarchical p-version of the finite element method. *Computer Methods in Applied Mechanics and Engineering*, v. 150p.215–238, 1997.
- Jeremic, B., Xenophontos, C. Application of the p-version of the finite element method to elastoplasticity with localization of deformation. *Communications in Numerical Methods in Engineering*, v. 15p.867–876, 1999.
- Karniadakis, G. E., Sherwin, S. J. *Spectral/hp Element Methods for CFD*. Oxford University Press, Oxford, 1999.
- Karniadakis, G. E., Sherwin, S. J. *Spectral/hp Element Methods for Computational Fluid Dynamics*. Oxford University Press, Oxford, 2005.
- Korneev, V. G., Jensen, S. Preconditioning of the p-version of the finite element method. *Computer Methods in Applied Mechanics and Engineering*, v. 150p.215–238, 1997.
- Ledgera, P. D., Morgana, K., Peraireb, J., Hassana, O. , Weatherilla, N. The development of an hp-adaptive finite element procedure for electromagnetic scattering problems. *Finite Elements in Analysis and Design*, v. 39p.751–764, 2003.
- Lomtev, I., Quillen, C. B. , Karniadakis, G. E. Spectral/hp methods for viscous compressible flows on unstructured 2d meshes. *Journal of Computational Physics*, v. 144p.325–357, 1998.
- Mackerle, J. Object-oriented programming in fem and bem a bibliography (1990-2003). *Advances in Engineering Software*, v. 35p.325–336, 2004.

- Maitre, J. F., 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. On block diagonal and schur complement preconditioning. *Numer. Math.*, v. 58p.79–93, 1990b.
- Mandel, J. Two-level domain decomposition preconditioning for the p-version finite element method in three dimensions. *International Journal for Numerical Methods in Engineering*, v. 29p.1095–1108, 1990c.
- Nogueira Jr., A. C. *Formulação p do Método de Elementos Finitos em problemas de elasticidade linear e não-linear com malhas 3D não-estruturadas e em métodos multigrid algébricos*. Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, 2002., Tese (Doutorado).
- Oden, J., Patra, A. A parallel adaptive strategy for hp finite element computations. *Computer Methods in Applied Mechanics and Engineering*, v. 121p.449–470, 1995.
- Oden, J. T. Some historic comments on finite elements. *Proceedings of the ACM conference on History of scientific and numeric computation*, v. 79p.125–130, 1987.
- Patra, A., Oden, J. T. Computational techniques for adaptive hp finite element methods. *Finite Elements in Analysis and Design*, v. 25p.27–39, 1997.
- Reddy, J. N. *An Introduction to the Finite Element Method*. McGraw-Hill International Editions, 2006.
- Schwab, C. *P- And Hp- Finite Element Methods: Theory and Applications in Solid and Fluid Mechanics*. Oxford University Press, Oxford, 1999.
- Shen, J., Tang, T. *Spectral and High-Order Methods with Applications*. Science Press, first ed., 2006.
- Sherwin, S. J., Karniadakis, G. A new triangular and tetrahedral basis for high-order (hp) finite element methods. *International Journal for Numerical Methods in Engineering*, v. 38p.3775–3802, 1995.

- Szabó, B. A. The p- and h-p versions of the finite element methods in solid mechanics. *Computer Methods in Applied Mechanics and Engineering*, v. 80p.185–195, 1990.
- Szabó, B., Babuska, I. *Finite Element Analysis*. Wiley Interscience, New York, 1991.
- Szabó, B. A., Mehta, A. K. p-convergent finite element approximations in fracture mechanics. *International Journal for Numerical Methods in Engineering*, v. 12p.551–560, 1978.
- Turner, M. Stiffness and deflection analysis of complex structures. *Journal of the Aeronautical Sciences*, v. 23p.805–823, 1956.
- Vazquez, M. *Construção de funções de interpolação para as versões h e p do MEF através de produto tensorial*. Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, São Paulo, Brasil, 2004., Dissertação (mestrado).
- Vazquez, T. *Funções de Base e Regras de Integração Tensorizáveis para o MEF de Alta Ordem*. Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, 2008., Tese (doutorado), São Paulo, Brasil.
- Zienkiewicz, O. C., Taylor, R. L. *The Finite Element Method*, v. 1. McGraw-Hill International Editions, fourth ed., 1989.