

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito: 20.02.2003

Assinatura: *Am. Paula Lampião*

Método de diferenças finitas generalizadas por mínimos quadrados

Daniel Ricardo Izquierdo Pena

Orientador: Prof. Dr. Luis Gustavo Nonato

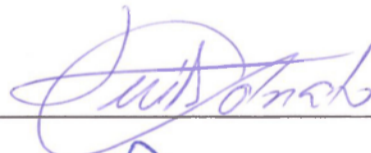
EXISTE UMA VERSÃO REVISADA

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação - ICMC-USP, como parte dos requisitos para obtenção do título de Mestre em Ciências de Computação e Matemática Computacional.

USP – São Carlos
Fevereiro/2003

A Comissão Julgadora:

Prof. Dr. Luis Gustavo Nonato



Prof. Dr. Antonio Castelo Filho



Profa. Dra. Célia Aparecida Zorzo Barcelos

Celia A. Z. Barcelos

Para mis viejos

AGRADECIMENTOS

Gostaria de prestar meus sinceros agradecimentos as seguintes pessoas e instituições:

Ao Prof.Dr. Luis Gustavo Nonato, pela orientação, dedicação, paciência e pela confiança depositada em mim, no decorrer do desenvolvimento deste trabalho.

A Carolina, Carmen e Carine por todo seu apoio.

A minha amiga Glaucia por todas suas sugerencias e pelo tempo compartilhado.

A minha familia e amigos em Colômbia por seu incondicional apoio e confiança.

A meus amigos da colonia que me proporcionaram momentos alegres e que me deram apoio em momentos difíceis neste período.

Aos meus companheiros e professores de laboratório que estiveram prontos a me esclarecer dúvidas e dar boas sugestões no decorrer deste período de minha vida.

Ao CNPq pelo financiamento deste trabalho junto a área de Pós-Graduação do ICMC-USP.

RESUMO

Neste trabalho é apresentado o Método de Diferenças Finitas Generalizadas (MDFG), como uma alternativa de método meshless para solucionar equações diferenciais parciais. Este método está baseado na aproximação local por mínimos quadrados a partir de nós espalhados sobre o domínio. É proposto um critério de seleção dos nós utilizados na aproximação, garantindo que o erro de truncamento gerado pelo MDFG é baixo. Demonstra-se teoricamente a convergência do método em certos problemas elípticos e parabólicos, e alguns exemplos são apresentados para comprovar este fato. A aplicação do método na solução de EDP em domínios de alta complexidade é mostrada num exemplo de escoamento de fluido numa artéria.

ABSTRACT

A Generalized Finite-Difference Method (GFDM) is proposed in this work as a new meshless method to solve partial differential equations. The method is based on moving least square approximations of an arrangement of nodes distributed in a domain. The convergence of some elliptic and parabolic problems is theoretically prove and examples are also presented. The flexibility of ability GFDM to solving PDE in complex domains is illustrated though the simulation of the blood flow into an artery.

Conteúdo

Introdução.	1
1 MDFG por aproximação local de polinômios.	5
1.1 Célula computacional associada a um nó.	6
1.2 Aproximação por mínimos quadrados.	14
1.3 Consistência do método.	21
1.4 Estabilidade do método.	34
2 Implementação do MDFG.	43
2.1 Análise das EDP em duas dimensões.	43
2.2 Malhas não-estruturadas.	45
2.3 Erro de truncamento do MDFG.	48
2.3.1 Dependência do ET com a forma da célula.	49
2.3.2 ET relacionado ao tamanho relativo das células.	53
2.4 ET devido a função peso.	56
2.5 Teste Numérico	58
3 Aplicações do MDFG.	61
3.1 MDFG para problemas elípticos.	61
3.1.1 Representação da equação de Poisson pelo MDFG.	62
3.1.2 Exemplo numérico.	66
3.2 MDFG para problemas parabólicos.	67
3.2.1 Solução da equação de calor pelo MDFG.	69
3.2.2 Exemplo numérico.	71
3.3 MDFG para problemas de Navier-Stokes.	71

3.3.1	Problema da cavidade.	74
3.3.2	Escoamento numa artéria.	76
4	Conclusões.	79
A	Geração de Malhas Não-Estruturadas.	81
A.1	Triangulação de Delaunay.	81
A.2	Algoritmos de Refinamento.	82
A.2.1	Algoritmo de Chew.	82
A.2.2	Algoritmo de Rupert.	84
B	Estrutura de Dados.	89
B.1	A estrutura Singular Handle-Edge.	89
B.1.1	O elemento Malha.	90
B.1.2	O elemento Vértice.	90
B.1.3	O elemento Estrela do Vértice.	91
B.1.4	O elemento Face.	92
B.1.5	Os elementos Curva de Contorno e Arestas de Contorno.	92
B.2	Implementação.	92

Lista de Figuras

1.1	Nós das células (representados pelo círculo vazio) do nó x_i para: (a) diferenças centradas, (b) diferenças progressivas, e (c) diferenças regressivas. . .	7
1.2	Nós das células (representados pelo círculo vazio) do nó $x_{i,j}$ para diferenças centradas bi-dimensionais.	8
1.3	Nós da célula (representados pelo círculo vazio) do nó $x_{i,j,k}$ para diferenças centradas tri-dimensionais.	9
1.4	Seleção da célula: (a) critério da distância, (b) critério dos oito segmentos, e (c) critério dos quatro quadrantes.	10
1.5	Diferentes tipos de célula de nó.	11
1.6	Diferentes tipos de malhas.	13
1.7	Malha irregular hexagonal gerada a partir de uma célula regular, e de perturbações de ordem δs	14
1.8	Função de aproximação \bar{f}_o : (a) uma linha reta com $s = 1$, e (b) uma parábola com $s = 2$	16
1.9	Representação dos monômios base do espaço polinomial \mathcal{P}_s^d como pontos no espaço \mathcal{E}^d , com suas respectivas camadas de grau s	18
1.10	(a) Célula do nó v_o . (b) Representação do espaço \mathcal{E}^2	30
1.11	(a) Curvas dos elementos $e_3^{(1)}$, $e_3^{(2)}$ e $e_3^{(4)}$ para c_3 . (b) Curvas dos elementos $e_5^{(1)}$, $e_5^{(2)}$ e $e_5^{(4)}$ para c_5	32
1.12	Malha triangular regular bi-dimensional (onde todas as arestas tem comprimento h) com uma rotação num ângulo θ	33
2.1	Malhas triangulares num rectângulo com diferentes tamanhos de aresta mínima h , obtidos pelos métodos de refinamento de Chew e Rupert.	46

2.2	Distribuição do número de vértices das células C(1) dos vértices das figuras anteriores.	47
2.3	Valores das estimativas dos termos E_i^1 e E_i^2 em células de 5 nós (a,b), 6 nós (c,d), 7 nós (e,f), e 8 nós (g,h), respeito à ordem de perturbação δs . . .	52
2.4	Valores das estimativas dos termos E_4^1 e E_4^2 em células tipo C(I), C(II), e C(III), possuindo 5, 10 e 15 nós respectivamente.	53
2.5	Células tipo C(I) (a) e do tipo C(III) (b), para um nó no bordo ∂D do domínio D	54
2.6	Valores das estimativas dos termos E_3^1, E_3^2 associados à aproximação do $\partial^2 f / \partial x^2$ (a), e dos termos E_5^1, E_5^2 associados à aproximação do $\partial^2 f / \partial y^2$ (b), para células C(II) e C(III) de um nó $v_o \in \partial D$	55
2.7	Malha com duas células C_1 e C_2 tipo C(I) de diferentes forma e tamanho. .	56
2.8	Células C(I) regulares: C_1 pentagonal e C_2 hexagonal.	56
2.9	Célula C(I) hexagonal irregular C_3 do exemplo 2.2.	57
2.10	Termos de E_3^2 e E_5^2 para duas diferentes funções peso ω	58
2.11	Valores do erro para as aproximações de D_1, D_3 , e D_5 da função (2.9), para $c = 1$ e $a = 0.25$	59
3.1	Resultado do teste de convergência do MDFG num problema elíptico. . . .	68
3.2	Solução analítica e calculada pelo MDFG ($h = 0.117648$)	68
3.3	Resultado do teste de convergência do MDFG num problema parabólico. .	71
3.4	(a) malha não estruturada da cavidade (malha 2), (b) campo de velocidade no regime permanente.	75
3.5	Imagem microscópica de uma artéria.	76
3.6	Malha da artéria	77
3.7	Campo de velocidade de um escoamento dentro da artéria.	77
A.1	(a) Nenhuma das circunferas dos triângulos da TD contém pontos no seu interior. (b) TD e seu diagrama de Voronoi (linhas pontilhadas.)	82

A.2	Algoritmo de Chew. (a) Entrada original. (b) Triangulação de Delaunay do contorno original mais novos vértices na fronteira. (c) Eliminação dos elementos externos da triangulação. O “x” indica o circuncentro do triângulo de baixa qualidade, (d) inserção do ponto steiner no circuncentro e correspondente triangulação. (e) Malha depois de inserir 15 novos pontos e (f) triangulação final.	84
A.3	(a) Aresta violada, (b) duas novas arestas geradas pela inserção do novo vértice no meio da aresta anterior.	85
A.4	(a) Entrada inicial, (b) triangulação de Delaunay, (c) triangulação após o refinamento de Rupert, com um ângulo mínimo de 5° , (d) triangulação com ângulo mínimo de 25° . [30]	86
B.1	Relações do elemento Malha.	90
B.2	Relações do elemento Vértice.	90
B.3	Exemplo de Semi-aresta, Semi-aresta mate e Aresta de Contorno. O ponto preto representa o vértice ao qual a Semi-aresta aponta.	91
B.4	Relações do elemento Semi-aresta.	91
B.5	(a) Exemplo da Estrela do Vértice com uma singularidade, (b) relações do elemento Estrela do Vértice.	92
B.6	(a) Exemplo de Face orientada, (b) relações do elemento Face.	92
B.7	Relações dos elementos Curva de Contorno e Arestas de Contorno.	93

Introdução.

O método de diferenças finitas generalizadas (MDFG) proposto neste trabalho é desenvolvido a partir do método de aproximações por mínimos quadrados localizados (MLS)¹, introduzido por Shepard [31]; o qual fornece uma alternativa às interpolações clássicas de aproximação de funções a partir de seus valores dados em uma série de pontos distribuídos irregularmente. Recentemente, o MLS para a solução numérica de equações diferenciais tem recebido grande destaque, especialmente na literatura de engenharia mecânica, gerando uma série de métodos chamados *meshless*, os quais possibilitam uma aproximação numérica a partir de um conjunto de pontos que podem não ter como suporte uma malha ou triangulação. A classificação de um método como sendo do tipo *meshless* ainda não está estabelecido na literatura. Neste trabalho é utilizada a definição dada por Krysl e Belytschko [4], na qual um método é considerado *meshless* se as bases da aproximação são construídas a partir de um suporte arbitrário gerado por uma coleção de nós distribuídos irregularmente.

Uma característica atrativa dos métodos *meshless* é sua facilidade de adaptação em espaços arbitrários, e a flexibilidade na implementação de esquemas adaptativos.

Os métodos *meshless* podem ser divididos em duas categorias: os métodos baseados sob princípios variacionais e métodos que atuam diretamente nas equações diferenciais governantes. Na primeira categoria pode-se ter os seguintes métodos: *smooth particle hydrodynamics* (SPH), por Gingold e Monaghan [13][23]; *diffuse element methods* (DEM), por Nayroles et al. [24]; *element free Galerkin* (EFG), por Belytschko et al. [5]; *reproducing Kernel particle* (RKP), por Liu et al. [20]; *h-p cloud method* por Duarte et al [9]; *partition of unit finite element method* (PUFEM) por Babuška e Melenk [3]; *meshless local Petro-Galerkin* (MLPG) e *local boundary integral equation* (LBIE) por Atluri e Zhu [2].

¹*moving least square*

Tais métodos têm como característica comum a utilização de uma integração numérica para o estabelecimento das equações discretas do sistema. Integrações numéricas tem efeitos sob a precisão e convergência da solução destes métodos meshless, mas apresentam dificuldades devido às seguintes razões: as integrações numéricas geralmente são complexas devido à complexidade da forma da função; é muito difícil escolher uma ordem de integração com todos os fatores considerados; além disso, uma baixa ordem de integração gera soluções de baixa precisão, em quanto integrações de alta ordem incrementam excessivamente o custo computacional.

Na segunda categoria de métodos meshless é considerado o método de diferenças finitas generalizadas (MDFG), no qual um conjunto de equações discretas são estabelecidas diretamente a partir das equações diferenciais. Uma vantagem deste método está na não utilização de integrações numéricas. A idéia de utilizar nós postos irregularmente num domínio para a obtenção das aproximações das diferenças finitas não é nova, Jensen [16] no final dos anos 60, apresenta um método de diferenças finitas o qual utiliza células (ou estrelas) irregulares com seis pontos. Utilizando a expansão de séries de Taylor ele obtém uma formulação de diferenças finitas a qual aproxima derivadas até de segunda ordem. A principal desvantagem deste método é que apresenta frequentes singularidades ou mau condicionamento da célula. Perrone e Kao [28] sugerem a adição de mais nós na célula e a aplicação de uma média para a geração dos coeficientes das diferenças finitas. Liska e Orkisz [19] tem feito uma interessante contribuição ao desenvolvimento do método no que se refere à seleção de células na tentativa de eliminar os problemas indicados acima, aplicando o método na solução de problemas lineares e não-lineares. Recentemente os trabalhos de Luo e Hüßler-Combe [21], Benito et al. [32], Marshall e Grand [22], e Gossler [14] fazem uso do método na construção de esquemas de diferenças finitas, expandido suas aplicações para a solução de diferentes problemas. A pesar das vantagens oferecidas por este método, como sua facilidade de implementação, flexibilidade e pouco custo, esta categoria de métodos meshless tem recebido pouca atenção comparada com os métodos variacionais da primeira categoria.

Embora o MDFG seja um método meshless (o qual utiliza um conjunto de nós espalhados no domínio, os quais podem não ter nenhuma conectividade entre eles), é interessante utilizar malhas (estruturadas ou não estruturadas) como suporte dos nós, com a qual,

pode-se garantir uma melhor distribuição dos nós sobre o domínio rápida busca de vizinhança. O objetivo principal deste trabalho é estudar a convergência do MDFG, quando aplicado na solução de problemas de equações diferenciais parciais. Uma análise do erro que o método introduz, dependendo das posições dos nós utilizados nas aproximações e um estudo sobre consistência e estabilidade também é apresentado.

O trabalho é organizado da seguinte forma: No primeiro capítulo são definidas diferentes tipos de células baseadas nas propriedades intrínsecas das malhas (por ex: relações entre nós por arestas). Além disso, é definido o MDFG como um método diferencial para solucionar problemas d -dimensionais, e é feito um estudo analítico da consistência e estabilidade do método. No segundo capítulo, se estuda numericamente os erros de truncamento gerados pelo MDFG, para diferentes tipos de células bi-dimensionais, com respeito a suas formas e tamanhos. Também é apresentado um critério de escolha de células. Finalmente, no terceiro capítulo, são apresentadas algumas aplicações do MDFG na solução numérica de equações diferenciais parciais bi-dimensionais, avaliando a convergência do método.

Capítulo 1

MDFG por aproximação local de polinômios.

A natureza dos fenômenos físicos geralmente pode ser modelada mediante relações matemáticas conhecidas como equações diferenciais parciais (EDP). Nas EDP uma ou várias funções sob um domínio contínuo são obrigadas a satisfazer uma série de condicionamentos nos quais estão envolvidas as derivadas das próprias funções, além de condições nas fronteiras do domínio. Algumas vezes, tais equações diferenciais podem ser solucionadas analiticamente, encontrando uma ou um conjunto de funções solução das EDP. Lamentavelmente, na maioria dos problemas é impossível obter uma solução analítica para uma EDP, ou então, tal solução pode ser muito complexa para ser trabalhada. Nestas situações, a solução da EDP geralmente é calculada utilizando métodos numéricos que aproximam a solução a partir de uma discretização do domínio. Os métodos numéricos para a solução de EDP mais conhecidos são: Elementos Finitos (EF), Volumes Finitos (VF) e Diferenças Finitas (DF). Nos métodos de DF, uma EDP é definida sob um domínio contínuo é aproximada por um sistema de equações algébricas geradas a partir de uma discretização de tal domínio. A partir das equações algébricas calcula-se uma solução numérica que aproxima a EDP original.

Este trabalho pretende estudar o método de diferenças finitas generalizadas, utilizando qualquer tipo de discretização regular ou irregular do domínio. As discretizações regulares são utilizadas nos métodos clássicos de DF, e elas são construídas a partir de malhas regulares, nas quais todos os nós possuem uma mesma relação topológica com seus vi-

zinhos (a exceção dos nós da fronteira); comumente são utilizadas malhas retangulares, hexagonais, tetraédricas, etc.

Neste capítulo inicialmente é introduzido o conceito de célula computacional; para posteriormente apresentar o método de aproximações de diferenças finitas generalizadas, que utiliza polinômios de aproximação local, ajustados por mínimos quadrados. Os coeficientes dos polinômios obtidos por este esquema darão informações das derivadas das funções requeridas nas soluções numéricas da EDP. A consistência e a estabilidade do método são apresentadas com o propósito de estudar a convergência do método.

1.1 Célula computacional associada a um nó.

Seja um domínio $D \subset \mathbb{R}^d$ e um conjunto de nós $V = \{v_1, \dots, v_{n_v}\}$ tais que $V \subset D$. A célula computacional (ou simplesmente célula) do nó $v_i \in V$, denotada C_i , é definida como o conjunto de n_i nós $v_k \in V$, que são utilizados no cálculo dos valores aproximados das derivadas de uma função contínua no nó v_i . As células dos nós são também conhecidas na literatura como estrelas de nó [10][19] ou moléculas computacionais quando é incluído o nó principal [11].

A seguir serão apresentados alguns exemplos de células computacionais regulares de esquemas clássicos de DF:

Exemplo 1.1. *Seja um problema uni-dimensional, onde o domínio $D \subset \mathbb{R}$ de uma função f tem um comprimento L . O domínio D pode ser discretizado por uma coleção de N nós associados aos pontos x_i distribuídos uniformemente sob D . Seja o ponto $x_0 \in D$ tal que $x_0 \leq x$, para $\forall x \in D$, chamado ponto inicial. Para qualquer ponto x_i da discretização, tem-se que $x_i = x_0 + i\Delta x$, onde $\Delta x = L/(N - 1)$ e $i = 0, \dots, N$.*

As aproximações das derivadas de primeira ordem da função f podem ser obtidas mediante diferenças progressivas, centradas ou regressivas, como:

$$\text{Diferença progressiva:} \quad \left. \frac{\partial f}{\partial x} \right|_{x_i} = \frac{f(x_{i+1}) - f(x_i)}{\Delta x} + O(\Delta x)$$

$$\text{Diferença centrada:} \quad \left. \frac{\partial f}{\partial x} \right|_{x_i} = \frac{f(x_{i+1}) - f(x_{i-1}))}{2\Delta x} - O(\Delta x^2)$$

$$\text{Diferença regressiva: } \left. \frac{\partial f}{\partial x} \right|_{x_i} = \frac{f(x_i) - f(x_{i-1})}{\Delta x} + O(\Delta x)$$

onde $O(\cdot)$ indica a ordem do erro de truncamento que induz a diferença na aproximação. No caso do cálculo das aproximações da segunda derivada, pode ser utilizado as diferenças centradas:

$$\text{Diferença centrada: } \left. \frac{\partial^2 f}{\partial x^2} \right|_{x_i} = \frac{f(x_{i+1}) - 2f(x_i) + f(x_{i-1}))}{\Delta x^2} + O(\Delta x^2)$$

Outras aproximações de diferenças finitas podem ser empregadas, onde se utiliza mais de dois nós vizinhos, ou só os nós que ficam de um mesmo lado do nó principal, como os esquemas diferenciais de Leonard, Petro-Galerkin, etc [17]. Para o caso das diferenças centradas, a célula do nó x_i é dada por: $C_i = \{x_{i-1}, x_{i+1}\}$, como é apresentado na figura 1.1(a); para a diferença progressiva as células são dadas por $C_i = \{x_{i+1}\}$, ver figura 1.1(b); e para a diferença regressiva, as células são: $C_i = \{x_{i-1}\}$, ver figura 1.1(c).

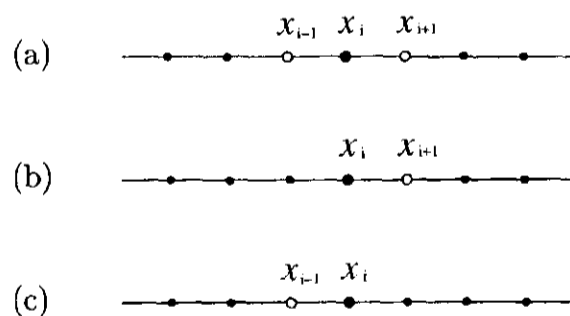


Figura 1.1: Nós das células (representados pelo círculo vazio) do nó x_i para: (a) diferenças centradas, (b) diferenças progressivas, e (c) diferenças regressivas.

Exemplo 1.2. Para um problema bi-dimensional com um domínio $D \subset \mathbb{R}^2$ retangular, pode-se fazer um tratamento análogo ao anterior, discretizando o domínio D por uma malha regular de elementos retangulares, onde cada um dos nós pode ser associado aos vértices dos elementos da malha, tal como se observa na figura 1.2. Para calcular as

derivadas de segunda ordem de uma função sob um nó $x_{i,j}$ pode-se utilizar diferenças centradas como:

$$\left. \frac{\partial^2 f}{\partial x^2} \right|_{x_{i,j}} = \frac{f(x_{i+1,j}) - 2f(x_{i,j}) + f(x_{i-1,j}))}{\Delta x^2} + O(\Delta x^2)$$

$$\left. \frac{\partial^2 f}{\partial y^2} \right|_{x_{i,j}} = \frac{f(x_{i,j+1}) - 2f(x_{i,j}) + f(x_{i,j-1}))}{\Delta y^2} + O(\Delta y^2)$$

Para o cálculo destas diferenças finitas tem-se uma única célula para um nó $x_{i,j}$ dada como $C_{i,j} = \{x_{i+1,j}, x_{i-1,j}, x_{i,j+1}, x_{i,j-1}\}$, tal como é apresentado na figura 1.2.

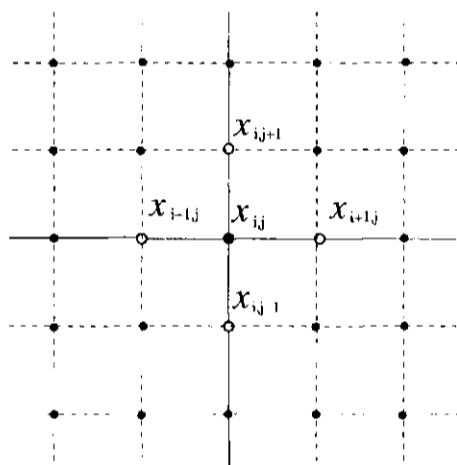


Figura 1.2: Nós das células (representados pelo círculo vazio) do nó $x_{i,j}$ para diferenças centradas bi-dimensionais.

Exemplo 1.3. Analogamente quando são utilizadas diferenças finitas centradas num caso tri-dimensional, num domínio que possa ser discretizado por uma malha regular cúbica, a célula do nó $x_{i,j,k}$ (ver figura 1.3) estará composta por: $C_{i,j,k} = \{x_{i+1,j,k}, x_{i-1,j,k}, x_{i,j+1,k}, x_{i,j-1,k}, x_{i,j,k+1}, x_{i,j,k-1}\}$.

Os métodos de DF mencionados acima, que utilizam discretizações regulares, são muito eficazes pois permitem um controle robusto do erro na aproximação e são muito rápidos, já que a busca das células dos nós é imediata.

Quando a discretização do domínio não pode ser regular, devido a fatores como uma geometria complexa do domínio ou refinamento local da malha, se faz necessário utilizar

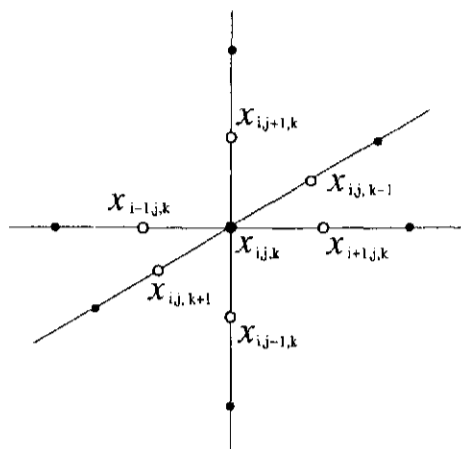


Figura 1.3: Nós da célula (representados pelo círculo vazio) do nó $x_{i,j,k}$ para diferenças centradas tri-dimensionais.

outros métodos de aproximação, como o MDFG que apresentaremos aqui, nos quais a busca e escolha dos nós que compõem uma célula não é tão imediata como no caso regular.

Diferentes critérios para a escolha dos nós de uma célula no caso bi-dimensional tem sido propostos na literatura, um dos primeiros foi dado por Jensen [16], onde os nós da célula são selecionados de acordo com a distância do nó central. Este critério é simples mas tem problemas quando os nós são dispersos irregularmente (Fig. 1.4a). Perrone e Kao [28] propõem o critério “dos oito segmentos”, onde o plano é dividido em oito segmentos iguais, escolhendo só um nó por segmento (Fig. 1.4b), obtendo uma boa escolha da célula, mas tornando-se muito rigoroso, complexo e com grande consumo de tempo computacional. Um critério análogo e mais simples é proposto por Liszka e Orkisz [19], no qual, o plano é dividido em quatro segmentos (quadrantes), e são escolhidos os dois nós mais próximos do nó central de cada segmento (Fig. 1.4c).

O critério utilizado neste trabalho difere dos anteriores por empregar como suporte uma malha não estruturada formada por elementos triangulares num domínio bi-dimensional (se o domínio é tri-dimensional os elementos são tetraedros), assim, as relações de vizinhança podem ser obtidas rapidamente utilizando uma estrutura de dados apropriada.

Na implementação do critério proposto neste trabalho, quando os nós são identificados pelos vértices da malha, utilizamos três tipos de células:

- Célula C(1) : Neste tipo de célula, os nós que compõem a célula de um nó v_i ,

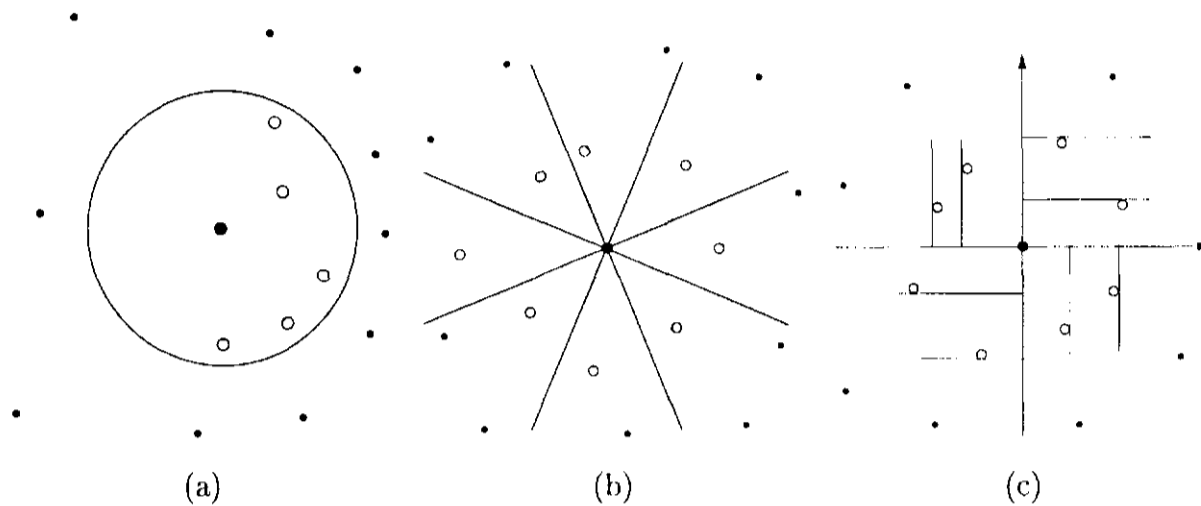


Figura 1.4: Seleção da célula: (a) critério da distância, (b) critério dos oito segmentos, e (c) critério dos quatro quadrantes.

correspondem aos vértices que compartilham uma aresta com v_i , tal como pode ser observado na figura 1.5a. Na implementação do MDFG, deve-se ter muito cuidado com este tipo de célula, já que o método precisa de uma quantidade mínima de nós (tal como será visto na seção 1.2), e podem existir células com uma quantidade de nós inferior ao exigido pelo método. Geralmente, este problema ocorre para células C(I) de nós no bordo do domínio. Não obstante, este tipo de célula é a que gera menos erros ao implementar o MDFG, como será visto na seção 2.3.

- Célula C(II) : Para um nó v_i , este tipo de célula é composto pelos nós da célula C(I) mais os vértices opostos a v_i por arestas cujos extremos estão em C(I); a figura 1.5b apresenta um exemplo destas células. Por possuir mais nós, C(II) por geral evita o problema de possuir menos nós que o mínimo exigido pelo método MDFG, mas apresenta a desvantagem de gerar maiores erros de aproximação.
- Célula C(III) : Este tipo de célula é composto pelos nós da célula C(I) relacionada a v_i , mais os vértices que compartilham uma aresta como os nós em C(I), com exceção de v_i . Um exemplo da célula tipo C(III) é apresentado na figura 1.5c. A utilização destas células será restrita unicamente a casos excepcionais, onde células C(II) não possuam uma quantidade mínima de nós exigida pelo MDFG. Um caso típico onde se apresenta este problema são as células dos nós que estão nos “cantos” das malhas.

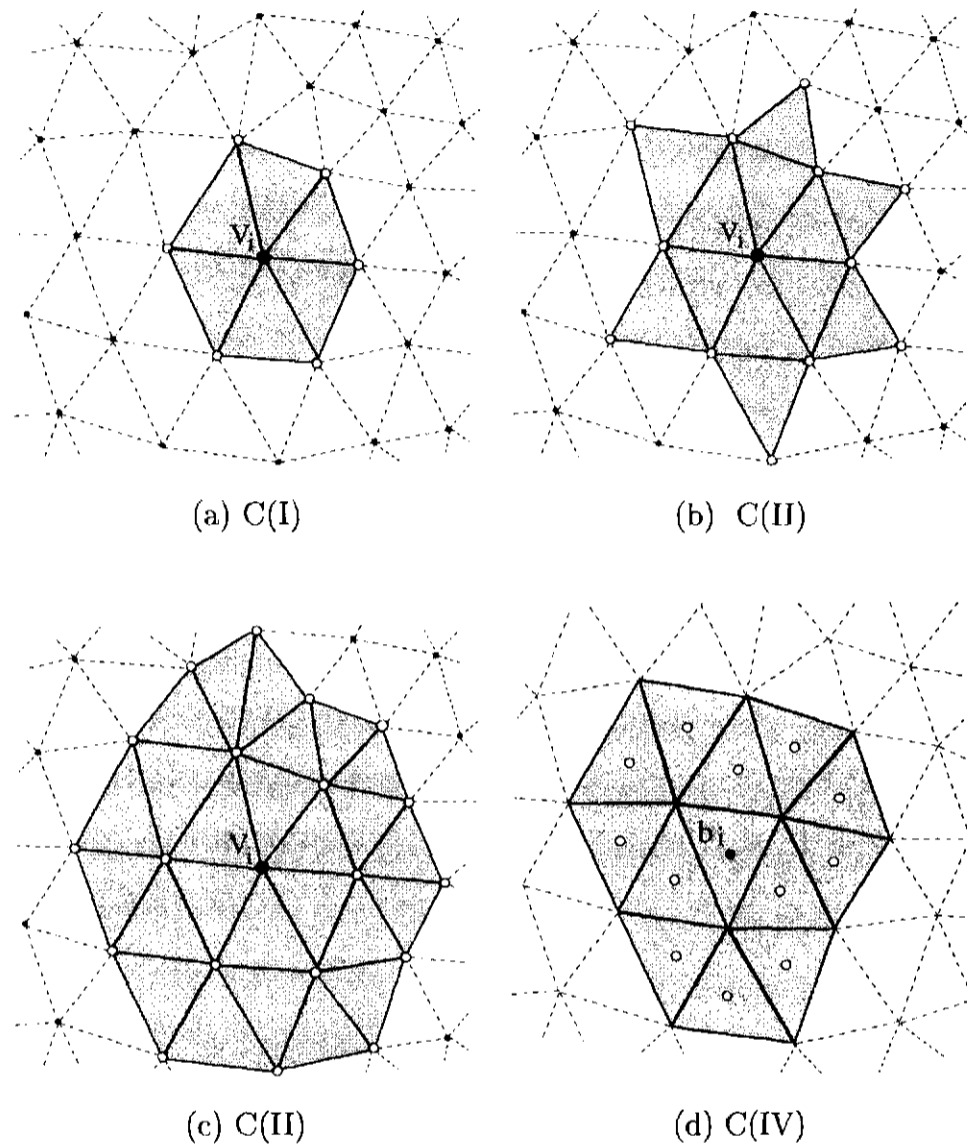


Figura 1.5: Diferentes tipos de célula de nó.

No caso onde os nós devem ser posicionados no baricentro dos elementos triangulares, uma célula tipo C(IV) é utilizada, definida como:

- Célula C(IV) : Os nós que compõem a célula tipo C(IV) para um nó b_i no baricentro de um elemento triangular, são dados pelos nós que estão nos elementos triangulares que intersectam o triângulo do nó b_i . Na figura 1.5d é apresentado um exemplo deste tipo de célula.

Para cada nó $v_i \in V$ situado na posição $\mathbf{r}_i = (x_i^1, \dots, x_i^d)$,¹ pode-se definir um novo sistema de coordenadas cuja origem está situada em \mathbf{r}_i , com o qual, uma posição qualquer

¹ $x_i^j = \langle v_i, e_j \rangle$, onde e_j são os elementos da base canônica.

$\mathbf{r} = (x^1, \dots, x^d)$ é expressa neste novo sistema de coordenadas como $\bar{\mathbf{r}} = (\bar{x}^1, \dots, \bar{x}^d)$ com seus componentes dados por:

$$\bar{x}^j = x^j - x_i^j \quad \text{para } j = 1, \dots, d \quad (1.1)$$

assim, as posições dos nós $v_k \in C_i$, podem ser expressos como $\bar{\mathbf{r}}_k = (\bar{x}_k^1, \dots, \bar{x}_k^d)$.

Em um problema bi ou tri-dimensional, pode ser utilizado por comodidade, a notação por letras (x_k, y_k) ou (x_k, y_k, z_k) , em vez da notação de supra-índices (x_k^1, x_k^2) ou (x_k^1, x_k^2, x_k^3) , para designar posições. Neste trabalho serão utilizadas ambas notações, especialmente a notação por letras para os exemplos.

Como uma medida do tamanho cada célula C_i , se define o raio da célula ρ_i , como $\rho_i = \min_{k \in C_i} \rho_{i,k}$, onde $\rho_{i,k}$ são as distâncias euclidianas de cada um dos nós v_k da célula com o nó principal v_i , assim, $\rho_{i,k} = \|\bar{\mathbf{r}}_k\|_{\text{eucl}}$ (comprimento da aresta $v_i v_k$). Portanto, pode-se introduzir um parâmetro de comprimento global da malha h como:

$$h = \min_{v_i \in V} \rho_i \quad (1.2)$$

o qual simplesmente, é igual ao comprimento da mínima aresta da malha. Com a ajuda deste parâmetro, podemos qualificar uma malha, pelo tamanho dos elementos que ela possui; assim, podemos dizer que uma malha é fina,² se h é muito pequeno. Nos algoritmos de geração de malhas, o usuário inclui um parâmetro relacionado a h , para poder fazer o ajuste da densidade de elementos no domínio, como pode ser observado a partir dos algoritmos dados no apêndice A. É importante esclarecer que a definição dada de h não é única, já que poderia ser utilizado outros parâmetros para indicar o tamanho dos elementos da malha, por exemplo: o comprimento da máxima aresta ou a média dos comprimentos das arestas.

O vetor posição $\bar{\mathbf{r}}_k$ de um nó v_k de uma célula C_i , pode ser normalizado respeito ao parâmetro h , como $\bar{\mathbf{r}}_k/h = \hat{\mathbf{r}}_k$, onde $\hat{\mathbf{r}}_k = \hat{\mathbf{r}}_k(\alpha_k^1, \dots, \alpha_k^d)$. Portanto, as componentes de $\bar{\mathbf{r}}_k$ serão dadas por:

²Na literatura de geometria computacional, é denominada uma malha grossa quando ela tem uma baixa densidade de elementos, e é chamada uma malha fina quando a densidade de elementos é grande.

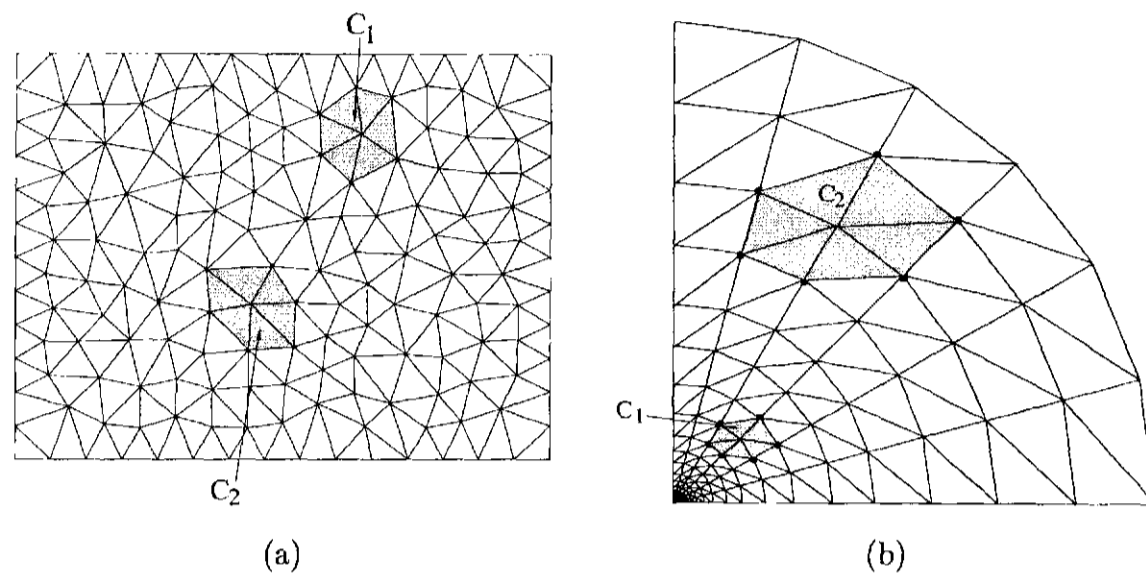


Figura 1.6: Diferentes tipos de malhas.

$$\bar{x}_k^j = h\alpha_k^j \quad \text{para } j = 1, \dots, d \quad (1.3)$$

Uma célula C_i está caracterizada geometricamente por sua forma (dados pelos termos α_k^j , com $j = 1, \dots, d$, e dos $v_k \in C_i$) e seu tamanho (relacionado a h). Nas malhas da figura 1.6 pode-se contemplar este fato, onde células tipo C(I) na malha da esquerda tem aproximadamente o mesmo tamanho, mas com diferente forma (um exemplo são as duas células C_1 e C_2); e na malha da direita são apresentadas células tipo C(I) com a mesma forma (só variando por um ângulo de rotação), mas de tamanhos diferentes, como é o caso das células C_1 e C_2 . Na análise dos erros do MDFG (capítulo 2) serão estudadas com mais detalhe as relações da forma e o tamanho das células e como elas afetam o método.

Finalmente, é definida uma célula irregular C_i com n_i nós como uma célula regular (obviamente com n_i nós), onde perturbações foram aplicadas nas posições de seus nós. Assim, pode-se criar uma célula irregular a partir de uma regular, adicionando uma série de perturbações $(\delta x_k, \delta y_k)$ a cada um dos nós da célula, tal que satisfaçam $(\delta x_k)^2 + (\delta y_k)^2 \leq (\rho_i \delta s)^2$ para $k = 1, \dots, n_i$, onde δs indica a ordem da perturbação. Na figura 1.7 é apresentado um exemplo de uma malha irregular hexagonal C(I) gerada a partir de uma célula regular, e de perturbações de ordem δs . Assim, uma célula irregular tende a ser uma célula regular quando $\delta s \rightarrow 0$.

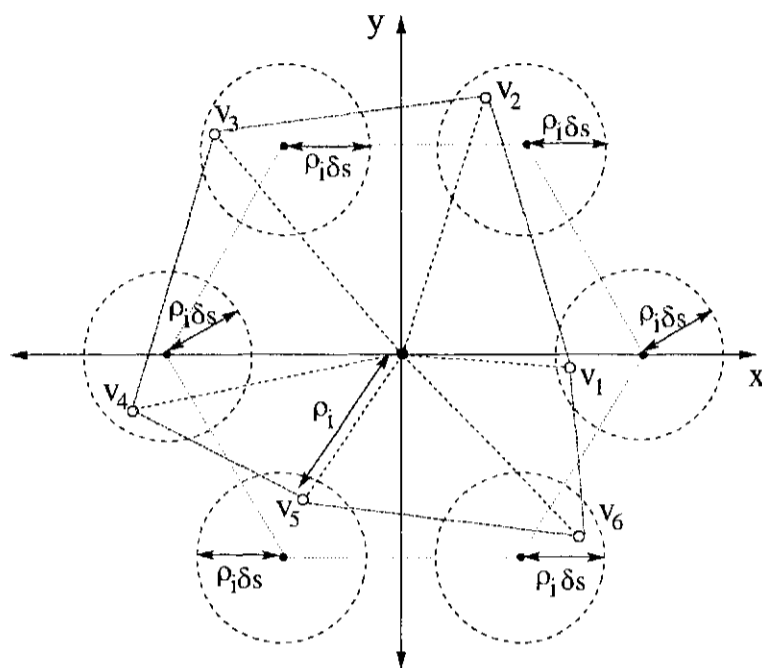


Figura 1.7: Malha irregular hexagonal gerada a partir de uma célula regular, e de perturbações de ordem δs .

1.2 Aproximação por mínimos quadrados.

Seja $f : D \rightarrow \mathbb{R}$ uma função de classe C^q definida em $D \subset \mathbb{R}^d$ e um conjunto de nós $V \subset D$. Suponha que para qualquer nó $v_o \in V$ é definida uma célula C_o , e o valor de $f(\mathbf{r}_o)$ é conhecido.

Procura-se uma função \bar{f}_o que aproxime f na vizinhança de v_o e cujas derivadas de grau menor ou igual a $s < q$ sejam fáceis de calcular. Uma boa alternativa é utilizar um polinômio de grau $s > 0$ ajustado por mínimos quadrados com os valores da função f nos nós de C_o . Suponha que

$$\bar{f}_o(\bar{\mathbf{r}}) = f(\mathbf{r}_o) + W_o(\bar{\mathbf{r}}), \quad (1.4)$$

onde W_o é um polinômio de aproximação de grau s dado por:

$$W_o(\bar{\mathbf{r}}) = \sum_{j=1} c_j P_o^{(j)}(\bar{\mathbf{r}}) \quad (1.5)$$

onde $P_o^{(j)}(\bar{\mathbf{r}})$ expressam os elementos de uma base do espaço polinomial \mathcal{P}_s^d e c_j são os

coeficientes correspondentes. O polinômio de aproximação $W_o(\bar{\mathbf{r}})$ deve ser nulo sob o nó v_o . Cada um dos elementos $P_o^{(j)}(\bar{\mathbf{r}})$ da base polinomial é um monômio de grau menor ou igual a s , com exceção do monômio de grau 0, que pode ser desconsiderado³. Por simplicidade, identificaremos $P_o^{(j)}(\bar{\mathbf{r}})$ como $P^{(j)}$, $W_o(\bar{\mathbf{r}})$ como W_o , e $P_o^{(j)}(\bar{\mathbf{r}}_k)$ como $P_k^{(j)}$ quando $\bar{\mathbf{r}}_k$ é o vetor posição do nó v_k , deixando implícito que estes elementos estão relacionados ao nó v_o .

Exemplo 1.4. Ao utilizar um polinômio de primeiro grau ($s = 1$) numa função de aproximação \bar{f}_o , para um problema uni-dimensional, W_o será dado como:

$$W_o(\bar{x}) = c_1 P^{(1)} = c_1 \bar{x}$$

assim, f é aproximado no ponto x_o por \bar{f}_o que será uma linha reta em \mathbb{R}^2 , (ver figura 1.8a). Para um problema bi-dimensional, W_o será dado por:

$$W_o(\bar{x}, \bar{y}) = c_1 P^{(1)} + c_2 P^{(2)} = c_1 \bar{x} + c_2 \bar{y}$$

com o qual a função de aproximação \bar{f}_o é um plano em \mathbb{R}^3 .

Exemplo 1.5. Utilizando um polinômio de segundo grau ($s = 2$) como uma função de aproximação num problema uni-dimensional, W_o será dado por:

$$W_o(\bar{x}) = c_1 P^{(1)} + c_2 P^{(2)} = c_1 \bar{x} + c_2 \bar{x}^2$$

assim, \bar{f}_o corresponde a uma parábola em \mathbb{R}^2 , (ver figura 1.8b). Para um problema bi-dimensional, W_o será:

$$W_o(\bar{x}, \bar{y}) = c_1 P^{(1)} + c_2 P^{(2)} + c_3 P^{(3)} + c_4 P^{(4)} + c_5 P^{(5)} = c_1 \bar{x} + c_2 \bar{y} + c_3 \bar{x}^2 + c_4 \bar{x} \bar{y} + c_5 \bar{y}^2$$

sendo \bar{f}_o um parabolóide em \mathbb{R}^3 .

³Por definição, o espaço polinomial \mathcal{P}_s^d é composto por todos os polinômios d -dimensionais de grau menor ou igual a s , incluindo os monômios de grau zero. Neste trabalho o espaço \mathcal{P}_s^d corresponde ao conjunto de todos os polinômios d -dimensionais de grau s' , onde $0 < s' \leq s$. Em problemas de interpolação, o espaço \mathcal{P}_s^d deve incluir os polinômios de grau zero [31].

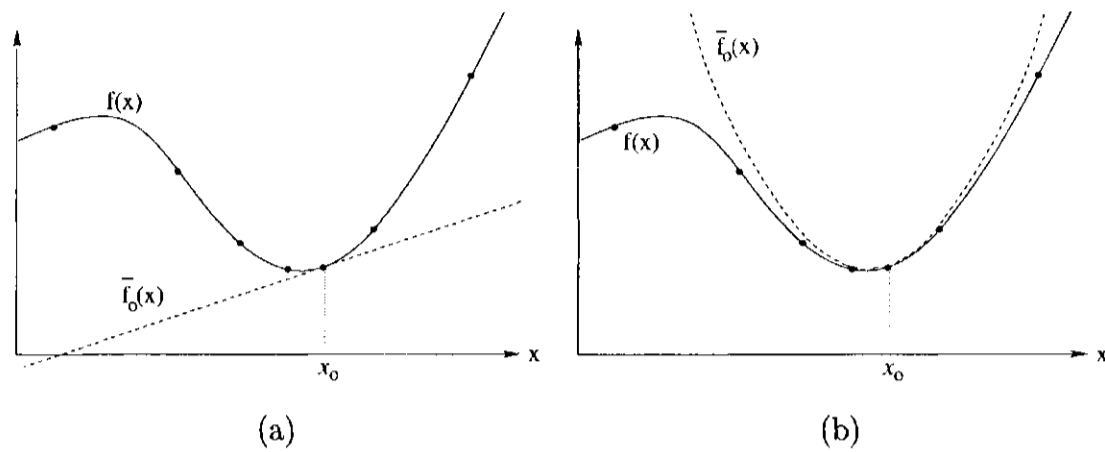


Figura 1.8: Função de aproximação \bar{f}_0 : (a) uma linha reta com $s = 1$, e (b) uma parábola com $s = 2$

Os coeficientes c_i do polinômio W_o aproximam as derivadas de f sob o nó v_o . No seguinte exemplo pode ser apreciado este fato:

Exemplo 1.6. Suponha uma função $f : D \subset \mathbb{R}^2 \rightarrow \mathbb{R}$ a qual é aproximada por \bar{f}_o sobre o nó v_o a partir de um polinômio de aproximação

$$W_o(\bar{x}, \bar{y}) = c_1\bar{x} + c_2\bar{y} + c_3\bar{x}^2 + c_4\bar{x}\bar{y} + c_5\bar{y}^2$$

As derivadas $\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial^2 f}{\partial x^2}, \frac{\partial^2 f}{\partial x \partial y}, \frac{\partial^2 f}{\partial y^2}$ podem ser estimadas de $\frac{\partial \bar{f}_o}{\partial \bar{x}}, \frac{\partial \bar{f}_o}{\partial \bar{y}}, \frac{\partial^2 \bar{f}_o}{\partial \bar{x}^2}, \frac{\partial^2 \bar{f}_o}{\partial \bar{x} \partial \bar{y}}, \frac{\partial^2 \bar{f}_o}{\partial \bar{y}^2}$, devido a que $\bar{\mathbf{r}}_o = (0, 0)$ de (1.4) se obtém:

$$\begin{aligned} \frac{\partial \bar{f}_o}{\partial \bar{x}} &= \frac{\partial W_o}{\partial \bar{x}} = c_1 + 2c_3\bar{x} + c_4\bar{y} = c_1 \\ \frac{\partial \bar{f}_o}{\partial \bar{y}} &= \frac{\partial W_o}{\partial \bar{y}} = c_2 + c_4\bar{x} + 2c_5\bar{y} = c_2 \\ \frac{\partial^2 \bar{f}_o}{\partial \bar{x}^2} &= \frac{\partial^2 W_o}{\partial \bar{x}^2} = 2c_3 \\ \frac{\partial^2 \bar{f}_o}{\partial \bar{x} \partial \bar{y}} &= \frac{\partial^2 W_o}{\partial \bar{x} \partial \bar{y}} = c_4 \\ \frac{\partial^2 \bar{f}_o}{\partial \bar{y}^2} &= \frac{\partial^2 W_o}{\partial \bar{y}^2} = 2c_5 \end{aligned}$$

Cada problema necessita de uma base $\{P^{(1)}, \dots, P^{(n)}\}$ para formar o espaço polinomial \mathcal{P}_s^d (tal como se observa nos exemplos 1.4 e 1.5). Cada um dos elementos da base $P^{(j)}$ é definido como um monômio de grau s' , com $0 < s' \leq s$, que é expresso como:

$$P^{(j)} = (\bar{x}^1)^{\mu_1(j)} (\bar{x}^2)^{\mu_2(j)} \dots (\bar{x}^d)^{\mu_d(j)} \quad (1.6)$$

onde $\mu_k(j)$ é um número inteiro, com $k = 1, \dots, d$, satisfazendo com as seguintes restrições:

$$0 \leq \mu_k(j) \leq s' \quad \text{e} \quad \sum_{k=1}^d \mu_k(j) = s' \quad (1.7)$$

Cada monômio $P^{(j)}$ pode ser representado como uma d -upla $(\mu_1(j), \dots, \mu_d(j))$ no espaço dos inteiros positivos \mathcal{E}^d de dimensão d , definido como:

$$\mathcal{E}^d = \underbrace{\mathbb{E}^+ \times \dots \times \mathbb{E}^+}_d$$

Ao satisfazer com as restrições (1.7) todos os monômios de grau s' , serão representados no espaço \mathcal{E}^d como uma coleção de pontos formando uma espécie de “camada”. Na figura 1.9 são apresentados exemplos destes pontos, suas respectivas camadas (representadas pelas caixas pontilhadas) e o grau que as caracterizam.

A base do espaço polinomial \mathcal{P}_s^d é composta pelos monômios representados por todos os pontos das camadas de grau menor ou igual a s ; assim, os primeiros elementos da base possuem os monômios da camada de grau 1, em seguida os monômios da camada de grau 2, e assim até completar todos os monômios de grau s , tal como é apresentado na tabela 1.1, para os espaços polinomiais \mathcal{P}_3^1 , \mathcal{P}_3^2 e \mathcal{P}_2^3 . A ordenação entre monômios de um mesmo grau é de livre escolha. Neste trabalho isto é feito seguindo o caminho das setas entre pontos de uma mesma camada nas figuras 1.9b e 1.9c. Assim, por notação, um ponto do espaço \mathcal{E}^d é chamado o j -ésimo ponto de \mathcal{E}^d se ele corresponde ao elemento $P^{(j)}$.

O próximo lema nos mostra como obter o grau de um monômio $P^{(j)}$ a partir de j .

Lema 1.1. *Se $P^{(j)}$ o j -ésimo elemento de \mathcal{P}_s^d , então o grau de $P^{(j)}$ é dado por*

$$N_d(j) = \min \left\{ k \mid \sum_{i=1}^k \binom{d}{2} \times (r-1) + d \geq j \right\}$$

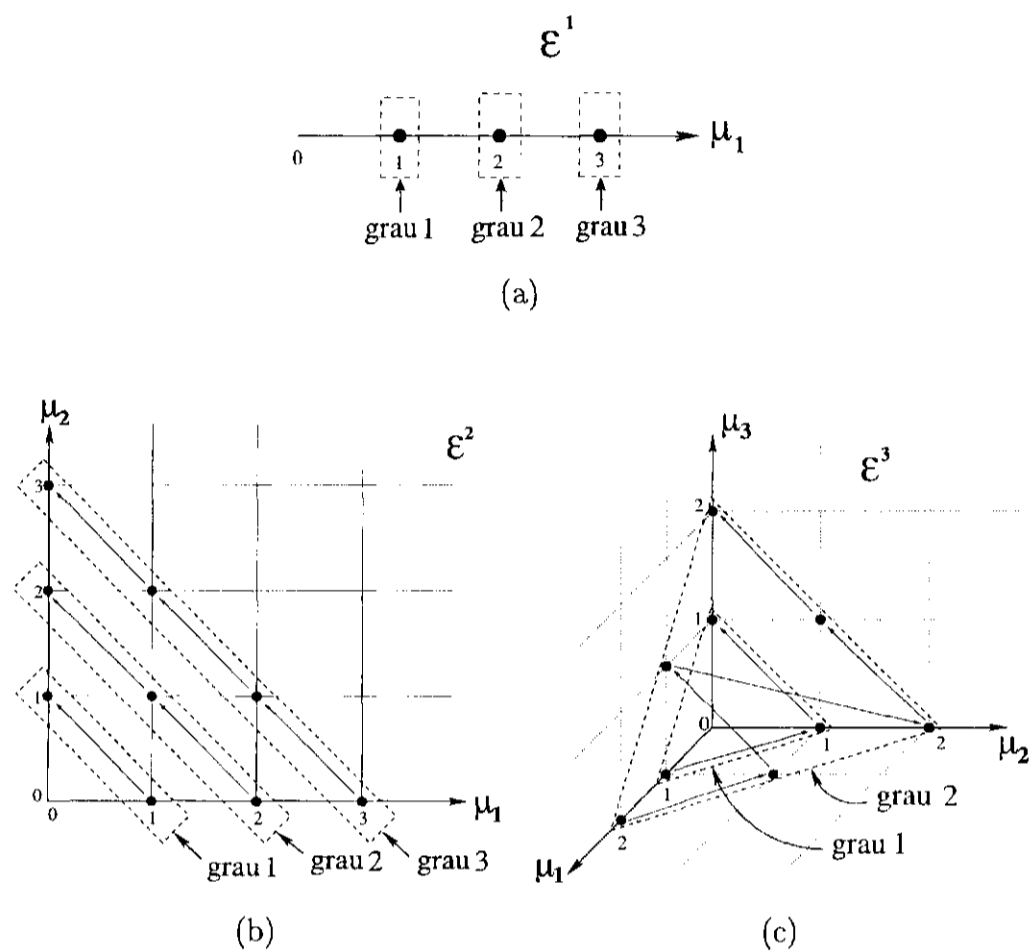


Figura 1.9: Representação dos monômios base do espaço polinomial \mathcal{P}_s^d como pontos no espaço \mathcal{E}^d , com suas respectivas camadas de grau s .

Demonstração. A prova é uma consequência imediata do fato que o número de monômios de grau r no espaço de dimensão d é

$$\binom{d}{r} \times (r - 1) + d$$

□

Corolário 1.1. Se quisermos que W_o seja um polinômio de grau s então a numeração dos monômios vai de 1 até $\sum_{i=1}^s \binom{d}{i} \times (i - 1) + d$, isto é

$$\bar{f}_o(\bar{\mathbf{r}}) = f(\mathbf{r}_o) + W_o = f(\mathbf{r}_o) + \sum_{i=1}^n c_i P^{(i)}$$

$d = 1$		
monômio	μ_1	grau
$P^{(1)} = \bar{x}$	1	1
$P^{(2)} = \bar{x}^2$	2	2
$P^{(3)} = \bar{x}^3$	3	3

(a)

$d = 2$			
monômio	μ_1	μ_2	grau
$P^{(1)} = \bar{x}$	1	0	1
$P^{(2)} = \bar{y}$	0	1	1
$P^{(3)} = \bar{x}^2$	2	0	2
$P^{(4)} = \bar{x}\bar{y}$	1	1	2
$P^{(5)} = \bar{y}^2$	0	2	2
$P^{(6)} = \bar{x}^3$	3	0	3
$P^{(7)} = \bar{x}^2\bar{y}$	2	1	3
$P^{(8)} = \bar{x}\bar{y}^2$	1	2	3
$P^{(9)} = \bar{y}^3$	0	3	3

(b)

$d = 3$				
monômio	μ_1	μ_2	μ_3	grau
$P^{(1)} = \bar{x}$	1	0	0	1
$P^{(2)} = \bar{y}$	0	1	0	1
$P^{(3)} = \bar{z}$	0	0	1	1
$P^{(4)} = \bar{x}^2$	2	0	0	2
$P^{(5)} = \bar{x}\bar{y}$	1	1	0	2
$P^{(6)} = \bar{x}\bar{z}$	1	0	1	2
$P^{(7)} = \bar{y}^2$	0	2	0	2
$P^{(8)} = \bar{y}\bar{z}$	0	1	1	2
$P^{(9)} = \bar{z}^2$	0	0	2	2

(c)

Tabela 1.1: Bases dos espaços polinomiais (a) \mathcal{P}_3^1 , (b) \mathcal{P}_3^2 , e (c) \mathcal{P}_2^3 .

Conhecendo os monômios $P^{(j)}$ o problema de calcular W_o se resume em obter os coeficientes c_1, \dots, c_n , que aproximem a função \bar{f}_o na região da célula C_o . Utilizamos para isto um ajuste de mínimos quadrados para calcular tais coeficientes; obtendo o seguinte sistema linear, onde os coeficientes c_i estarão contidos no vetor das incógnitas:

$$A\mathbf{c} = \mathbf{b} \quad (1.8)$$

onde A é a matriz composta pelos elementos $a_{i,j}$ dados por,

$$a_{j,i} = \sum_{v_k \in C_o} P_k^{(j)} P_k^{(i)} \omega_k \quad \text{com } i, j = 1, \dots, n, \quad (1.9)$$

o vetor de incógnitas $\mathbf{c} = [c_1, \dots, c_n]^T$, e $\mathbf{b} = [b_1, \dots, b_n]^T$ onde b_i denotam os produtos escalares,

$$b_j = \sum_{v_k \in C_o} (f(\mathbf{r}_k) - f(\mathbf{r}_o)) P_k^{(j)} \omega_k \quad (1.10)$$

avaliados sobre os nós $v_k \in C_o$. A função peso ω_k foi introduzida, e geralmente, depende das posições entre os nós v_o e v_k . Algumas vezes, a função peso ω_k é utilizada para privilegiar nós numa dada direção (por exemplo na direção das camadas de um escoamento). Neste trabalho utilizamos funções peso isotrópicas, onde o valor depende unicamente da distância de um ponto qualquer ao nó v_o :

$$\omega_k = \omega_k(\rho_{o,k}) \quad (1.11)$$

A função peso não deve incrementar-se com seu argumento; isto é,

$$\omega_a \geq \omega_b \iff \rho_{o,a} \leq \rho_{o,b}$$

onde $v_a, v_b \in V$; assim, os nós mais distantes do nó central v_o , têm menos influência na aproximação que os nós mais próximos a v_o . Na formulação de aproximação por mínimos quadrados, é assumido que a função peso tem um suporte limitado a uma bola finita de raio R :

$$\omega_a(\rho_{o,a}) = 0 \iff \rho_{o,a} > R$$

Um exemplo de função peso é a dada por Duarte [8], definida mediante a seguinte função contínua:

$$\omega_k(\rho_{o,k}) = \begin{cases} \sqrt{4/\pi} (1.0 - (\rho_{o,k}/R)^2)^4, & \forall \rho_{o,k} < R \\ 0, & \forall \rho_{o,k} \geq R. \end{cases}$$

Para evitar que o sistema (1.8) seja indeterminado, é importante que qualquer célula computacional C_o , tenha uma quantidade de nós igual ou maior a n (número de monômios).

1.3 Consistência do método.

A solução de uma EDP obtida pela utilização do MDFG é uma aproximação da solução real da EDP, devido à aproximação feita para o problema contínuo em um domínio discreto e finito. Esta aproximação é feita a partir de uma representação do domínio contínuo $D \subset \mathbb{R}^d$ por um conjunto finito V de nós espalhados sobre ele. O MDFG será consistente se quando a discretização tende para o domínio contínuo, $V \rightarrow D$, a solução obtida tende à solução real da EDP. Podemos dizer que uma discretização do domínio tende ao domínio contínuo quando para \forall vértice v_i e $\forall \epsilon > 0$ a bola de raio ϵ centrada em v_i possui pelo menos outro vértice $v_j \neq v_i$ em seu interior. Nesta seção é estudada a consistência das aproximações das derivadas obtidas pelo MDFG, analisando a dependência dos termos do erro de truncamento de cada uma delas com o parâmetro de comprimento global da malha h .

Seja um nó qualquer $v_o \in V$. Supondo que o polinômio de aproximação $W_o \in \mathcal{P}_s^d$, a função \bar{f} é dada por (1.4). Supondo que a matriz \mathbf{A} do sistema linear (1.8) utilizada no ajuste por mínimos quadrados de \bar{f} não é singular, então, seu determinante pode ser expresso mediante a regra de Laplace como,

$$\det(A) = \sum_{j=1}^n a_{j,i} \det(A_{[j,i]}) \quad (1.12)$$

onde $\mathbf{A}_{[j,i]}$ é a matriz de ordem $n-1$ obtida ao eliminar a i -ésima coluna e a j -ésima linha da matriz \mathbf{A} . Aplicando a regra de Cramer para solucionar o sistema (1.8), cada um dos coeficientes c_i do vetor \mathbf{c} , será dado por:

$$c_i = \sum_{j=1}^n b_j S_j^{(i)} \quad (1.13)$$

onde o elemento $S_j^{(i)}$ é dado por:

$$S_j^{(i)} = \frac{\det(\mathbf{A}_{[j,i]})}{\det(\mathbf{A})} \quad (1.14)$$

Utilizando a definição dos elementos b_j dada em (1.10) no sistema de coordenadas próprio do nó v_o , e substituindo em (1.13) obtemos:

$$c_i = \sum_{j=1}^n \sum_{v_k \in C_o} [f(\mathbf{r}_k) - f(\mathbf{r}_o)] P_k^{(j)} S_j^{(i)} \omega_k \quad (1.15)$$

onde a função f avaliada em qualquer nó v_k é equivalente a:

$$f(\mathbf{r}_k) = f(x_o^1 + \bar{x}_k^1, x_o^2 + \bar{x}_k^2, \dots, x_o^d + \bar{x}_k^d) = f(x_o^1 + P_k^{(1)}, x_o^2 + P_k^{(2)}, \dots, x_o^d + P_k^{(d)})$$

assim, ao expandir por séries de Taylor a função f ao redor do \mathbf{r}_o , se obtém:

$$f(\mathbf{r}_k) = \sum_{l'=0}^{\infty} \frac{1}{l'!} \left(\sum_{m=1}^d P_k^{(m)} \frac{\partial}{\partial x^m} \right)^{l'} f(\mathbf{r}_o) \quad (1.16)$$

Utilizando o teorema multinomial⁴, expressamos os termos entre parênteses como:

$$\left(\sum_{m=1}^d P_k^{(m)} \frac{\partial}{\partial x^m} \right)^{l'} = \sum \left(\frac{l'!}{\prod_{m=1}^d \mu_m!} \prod_{m=1}^d \left(P_k^{(m)} \frac{\partial}{\partial x^m} \right)^{\mu_m(l')} \right) \quad (1.17)$$

onde o somatório do lado direito da expressão é dado sobre cada uma das combinações dos inteiros não negativos μ_m cuja soma é l' . Cada uma destas combinações pode ser associada às coordenadas de um ponto na camada de grau $l' > 0$ do espaço \mathcal{E}^d , introduzido na seção 1.2. Ao incluir (1.17) em (1.16), os elementos $l'!$ são anulados, e os somatórios sob l' (equivalente à soma das camadas de grau $l' = 1, \dots, \infty$ do espaço \mathcal{E}^d) e o somatório sob as combinações de $\mu_m(l')$ (equivalentes às coordenadas dos pontos na camada l' no espaço \mathcal{E}^d) podem ser simplificados como um único somatório sob um índice l o qual estará associado ao l -ésimo ponto do espaço \mathcal{E}^d com coordenadas $(\mu_1(l), \dots, \mu_d(l))$, assim, o somatório será sobre todos os pontos deste espaço, seguindo a numeração discutida na seção 1.2. Portanto a série de Taylor (1.16) é expressa como:

⁴O teorema multinomial diz que: $\left(\sum_{i=1}^d a_i \right)^n = \sum \left(\frac{n!}{\prod_{i=1}^d n_i!} \prod_{i=1}^d a_i^{n_i} \right)$, onde n_i são inteiros não negativos cuja soma é n

$$f(\mathbf{r}_k) = f(\mathbf{r}_o) + \sum_{l=1}^{\infty} \frac{1}{\pi(l)} \prod_{m=1}^d \left(P_k^{(m)} \frac{\partial}{\partial x^m} \right)^{\mu_m(l)} f(\mathbf{r}_o) \quad (1.18)$$

com $\pi(l) = \prod_{m=1}^d \mu_m(l)!$. Expressando o produtório como:

$$\prod_{m=1}^d \left(P_k^{(m)} \frac{\partial}{\partial x^m} \right)^{\mu_m(l)} = \prod_{m=1}^d \left(P_k^{(m)} \right)^{\mu_m(l)} \prod_{m=1}^d \left(\frac{\partial}{\partial x^m} \right)^{\mu_m(l)}$$

e lembrando que $P_k^{(m)} = \bar{x}_k^m$ para $m = 1, \dots, d$, utilizamos a definição de $P^{(j)}$ dada em (1.6):

$$\prod_{m=1}^d \left(P_k^{(m)} \right)^{\mu_m(l)} = (\bar{x}_k^1)^{\mu_1(l)} \dots (\bar{x}_k^d)^{\mu_d(l)} = P_k^{(l)},$$

e definindo o operador diferencial D_l como:

$$D_l = \prod_{m=1}^d \left(\frac{\partial}{\partial x^m} \right)^{\mu_m(l)} = \frac{\partial^{N_d(l)}}{(\partial x^1)^{\mu_1(l)} \dots (\partial x^d)^{\mu_d(l)}} \quad (1.19)$$

(1.18) pode ser escrito como:

$$f(\mathbf{r}_k) = f(\mathbf{r}_o) + \sum_{l=1}^{\infty} \left(\frac{1}{\pi(l)} P_k^{(l)} D_l \right) f(\mathbf{r}_o) \quad (1.20)$$

Substituindo (1.20) na equação (1.15) obtemos:

$$c_i = \sum_{j=1}^n \sum_{v_k \in C_o} \sum_{l=1}^{\infty} \frac{1}{\pi(l)} P_k^{(l)} P_k^{(j)} S_j^{(i)} \omega_k [D_l f(\mathbf{r}_o)] \quad (1.21)$$

separando em (1.21) os termos com parâmetro l maiores que n e fatorando os termos que são menores ou iguais a n , obtemos:

$$\begin{aligned}
c_i &= \sum_{j=1}^n \sum_{v_k \in C_o} \left(\sum_{l=1}^n \frac{1}{\pi(l)} P_k^{(l)} P_k^{(j)} S_j^{(i)} \omega_k [D_l f(\mathbf{r}_o)] \right) \\
&\quad + \sum_{j=1}^n \sum_{v_k \in C_o} \left(\sum_{l=n+1}^{\infty} \frac{1}{\pi(l)} P_k^{(l)} P_k^{(j)} S_j^{(i)} \omega_k [D_l f(\mathbf{r}_o)] \right) \\
&= \sum_{l=1}^n \left(\frac{1}{\pi(l)} [D_l f(\mathbf{r}_o)] \sum_{j=1}^n \left(S_j^{(i)} \sum_{v_k \in C_o} P_k^{(l)} P_k^{(j)} \omega_k \right) \right) + \bar{T}_{i,n}
\end{aligned} \tag{1.22}$$

onde o termo $\bar{T}_{i,n}$ portanto, contém os elementos de ordem superior ($l > n$). Lembrando o produto definido em (1.9), a expressão (1.22) é simplificada, obtendo:

$$c_i = \sum_{l=1}^n \left(\frac{1}{\pi(l)} [D_l f(\mathbf{r}_o)] \sum_{j=1}^n S_j^{(i)} a_{j,l} \right) + \bar{T}_{i,n} \tag{1.23}$$

onde $a_{j,l}$ são os elementos da matriz \mathbf{A} .

Proposição 1.1.

$$\sum_{j=1}^n S_j^{(i)} a_{j,l} = \delta_{i,l} \quad , \quad \text{com } i, l = 1, \dots, n \tag{1.24}$$

onde $\delta_{i,l}$ é a função delta de Kronecker ($\delta_{i,l} = 1$ se $i = l$ e $\delta_{i,l} = 0$ se $i \neq l$).

Demonstração. Inicialmente tomemos o caso quando os parâmetros i e l são iguais. Utilizando a definição de $S_j^{(i)}$ dada em (1.14), obtemos:

$$\sum_{j=1}^n S_j^{(i)} a_{j,i} = \frac{1}{\det(\mathbf{A})} \sum_{j=1}^n a_{j,i} \det(\mathbf{A}_{[j,i]})$$

aplicando (1.12), a primeira relação é satisfeita:

$$\sum_{j=1}^n S_j^{(i)} a_{j,i} = \frac{\det(\mathbf{A})}{\det(\mathbf{A})} = 1$$

Sejam os parâmetros i e l diferentes. Por tanto temos:

$$\sum_{j=1}^n S_j^{(i)} a_{i,j} = \frac{1}{\det(\mathbf{A})} \sum_{j=1}^n a_{j,i} \det(\mathbf{A}_{[j,i]}) = \frac{\det(\mathbf{A}')}{\det(\mathbf{A})}$$

onde a matriz \mathbf{A}' é formada por todos os elementos matriciais de \mathbf{A} com exceção da i -ésima coluna, que é igual à l -ésima coluna de \mathbf{A} , assim, \mathbf{A}' terá duas colunas iguais, então, $\det(\mathbf{A}') = 0$, portanto:

$$\sum_{j=1}^n S_j^{(i)} a_{i,j} = \frac{\det(\mathbf{A}')}{\det(\mathbf{A})} = 0$$

□

Aplicando (1.24) na equação (1.23), obtemos:

$$c_i = \sum_{l=1}^n \left(\frac{1}{\pi(l)} [D_l f(\mathbf{r}_o)] \delta_{i,l} \right) + \bar{T}_{i,n} = \frac{1}{\pi(i)} [D_i f(\mathbf{r}_o)] + \bar{T}_{i,n} \quad (1.25)$$

Portanto, como já era esperado, obtemos que o coeficiente c_i do polinômio de aproximação W_o é igual ao valor da derivada D_i da função f no ponto \mathbf{r}_o , mais um termo extra $\bar{T}_{i,n}$ (este termo extra é formado por elementos de ordem superior, e dele depende o erro de truncamento do método). Por definição, o termo $\bar{T}_{i,n}$ é dado como:

$$\bar{T}_{i,n} = \sum_{l=1}^{\infty} T_{i,n+l} \quad (1.26)$$

onde,

$$T_{i,n+l} = \frac{1}{\pi(n+l)} [D_{n+l} f(\mathbf{r}_o)] \sum_{j=1}^n \sum_{v_k \in C_o} P_k^{(n+l)} P_k^{(j)} S_j^{(i)} \omega_k.$$

A fim de concluir o estudo da consistência do método, estudaremos a ordem de erro que introduz cada um dos elementos $T_{i,n+l}$ para $l = 1, 2, \dots$.

Lema 1.2. Cada um dos elementos $T_{i,n+l}$ é dado como:

$$T_{i,n+l} = \frac{1}{\pi(n+l)} \frac{\det(\mathbf{K}_i)}{\det(\mathbf{A})} [D_{(n+l)} f(\mathbf{r}_o)] \quad (1.27)$$

onde \mathbf{K}_i é uma matriz com elementos matriciais iguais aos da matriz \mathbf{A} , com exceção dos elementos da i -ésima coluna, os quais são formados pelos elementos β_j dados por:

$$\beta_j = \sum_{v_k \in C_d} P_k^{(n+l)} P_k^{(j)} \omega_k \quad (1.28)$$

Demonstração. Utilizando a definição de $S_j^{(i)}$ em cada um dos elementos $T_{i,n+l}$ de (1.26), obtemos:

$$T_{i,n+l} = \frac{1}{\pi(n+l)} [D_{(n+l)} f(\mathbf{r}_o)] \frac{1}{\det(\mathbf{A})} \sum_{j=1}^n \left(\det(\mathbf{A}_{[j,i]}) \sum_{v_k \in C_o} P_k^{(n+l)} P_k^{(j)} \omega_k \right) \quad (1.29)$$

substituindo o último somatório de (1.29) por β_j e aplicando a regra de Laplace:

$$T_{i,n+l} = \frac{1}{\pi(n+l)} [D_{(n+l)} f(\mathbf{r}_o)] \frac{1}{\det(\mathbf{A})} \sum_{j=1}^n \beta_j \det(\mathbf{A}_{[j,i]}) = \frac{1}{\pi(n+l)} \frac{\det(\mathbf{K}_i)}{\det(\mathbf{A})} [D_{(n+l)} f(\mathbf{r}_o)]$$

□

A matriz quadrada $\mathbf{A}_{[n \times n]}$, foi definida pelo sistema linear (1.8), onde cada um de seus elementos matriciais foi definido pelo produto (1.9). Esta matriz pode ser expressa como o produto das matrizes $\mathbf{P}_{[n \times n_o]}$, (lembrando que n_o é a quantidade de nós na célula C_o), e sua transposta \mathbf{P}^T . Assim, a matriz \mathbf{P} que satisfaz $\mathbf{P}\mathbf{P}^T = \mathbf{A}$, é dada como:

$$\mathbf{P} = \begin{bmatrix} \sqrt{\omega_1} P_1^{(1)} & \sqrt{\omega_2} P_2^{(1)} & \cdots & \cdots & \sqrt{\omega_{n_o-1}} P_{n_o-1}^{(1)} & \sqrt{\omega_{n_o}} P_{n_o}^{(1)} \\ \sqrt{\omega_1} P_1^{(2)} & \sqrt{\omega_2} P_2^{(2)} & \cdots & \cdots & \sqrt{\omega_{n_o-1}} P_{n_o-1}^{(2)} & \sqrt{\omega_{n_o}} P_{n_o}^{(2)} \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ \sqrt{\omega_1} P_1^{(n-1)} & \sqrt{\omega_2} P_2^{(n-1)} & \cdots & \cdots & \sqrt{\omega_{n_o-1}} P_{n_o-1}^{(n-1)} & \sqrt{\omega_{n_o}} P_{n_o}^{(n-1)} \\ \sqrt{\omega_1} P_1^{(n)} & \sqrt{\omega_2} P_2^{(n)} & \cdots & \cdots & \sqrt{\omega_{n_o-1}} P_{n_o-1}^{(n)} & \sqrt{\omega_{n_o}} P_{n_o}^{(n)} \end{bmatrix} \quad (1.30)$$

Cada um dos elementos $\sqrt{\omega_k} P_k^{(j)}$ da matriz anterior pode ser expresso utilizando as relações (1.6) e (1.3) como:

$$\begin{aligned}
\sqrt{\omega_k} P_k^{(j)} &= \sqrt{\omega_k} (\bar{x}_k^1)^{\mu_1(j)} \dots (\bar{x}_k^d)^{\mu_d(j)} = \sqrt{\omega_k} (h\alpha_k^1)^{\mu_1(j)} \dots (h\alpha_k^d)^{\mu_d(j)} \\
&= \sqrt{\omega_k} h^{N_d(j)} (\alpha_k^1)^{\mu_1(j)} \dots (\alpha_k^d)^{\mu_d(j)} = h^{N_d(j)} u_{j,k}
\end{aligned} \tag{1.31}$$

onde $u_{j,k} = \sqrt{\omega_k} (\alpha_k^1)^{\mu_1(j)} \dots (\alpha_k^d)^{\mu_d(j)}$. Podemos observar que a matriz \mathbf{P} se decompõe como o produto de uma matriz diagonal quadrada $\mathbf{H}_{[n \times n]}$ com $\text{diag}(\mathbf{H}) = \{h^{N_d(1)}, \dots, h^{N_d(n)}\}$ e uma matriz $\mathbf{P}'_{[n \times n]}$ composta pelos elementos $u_{j,k}$, ou seja:

$$\mathbf{P} = \mathbf{H}\mathbf{P}'$$

Então, a matriz \mathbf{A} , é expressa como, $\mathbf{A} = \mathbf{P}\mathbf{P}^T = (\mathbf{H}\mathbf{P}')(\mathbf{H}\mathbf{P}')^T = \mathbf{H}\mathbf{P}'\mathbf{P}'^T\mathbf{H}$; portanto o determinante de \mathbf{A} é dado por:

$$\det(\mathbf{A}) = \det(\mathbf{H}\mathbf{P}'\mathbf{P}'^T\mathbf{H}) = \det(\mathbf{H}) \det(\mathbf{P}'\mathbf{P}'^T) \det(\mathbf{H}) = (\det(\mathbf{H}))^2 \det(\mathbf{P}'\mathbf{P}'^T)$$

Como \mathbf{H} é diagonal e quadrada, seu determinante é igual ao produto dos elementos de sua diagonal, assim:

$$\det(\mathbf{A}) = h^\gamma \det(\tilde{\mathbf{A}}) \quad \text{com} \quad \gamma = 2 \sum_{j=1}^n N_d(j) \tag{1.32}$$

onde $\tilde{\mathbf{A}} = \mathbf{P}'\mathbf{P}'^T$.

Um procedimento semelhante é utilizado com a matriz \mathbf{K}_i , expressando-a como:

$$\mathbf{K}_i = \begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,i-1} & \beta_1 & a_{1,i+1} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,i-1} & \beta_2 & a_{2,i+1} & \dots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{n-1,1} & a_{n-1,2} & \dots & a_{n-1,i-1} & \beta_{n-1} & a_{n-1,i+1} & \dots & a_{n-1,n} \\ a_{n,1} & a_{n,2} & \dots & a_{n,i-1} & \beta_n & a_{n,i+1} & \dots & a_{n,n} \end{bmatrix} \tag{1.33}$$

onde os elementos matriciais $a_{j,i}$ estão definidos em (1.9), e os elementos β_j em (1.28). Esta matriz pode ser decomposta como o produto de duas matrizes $\mathbf{P}_{[n \times n]}$ e $\mathbf{R}_i_{[n \times n]}$, onde a matriz \mathbf{P} é igual à utilizada no caso da decomposição da matriz \mathbf{A} , e definida em (1.30). Então, \mathbf{R}_i é a matriz que cumpre a relação $\mathbf{P}\mathbf{R}_i = \mathbf{K}_i$, logo

$$\mathbf{R}_i = \begin{bmatrix} \sqrt{\omega_1} P_1^{(1)} & \cdots & \sqrt{\omega_1} P_1^{(i-1)} & \sqrt{\omega_1} P_1^{(n+l)} & \sqrt{\omega_1} P_1^{(i+1)} & \cdots & \sqrt{\omega_1} P_1^{(n)} \\ \sqrt{\omega_2} P_2^{(1)} & \cdots & \sqrt{\omega_2} P_2^{(i-1)} & \sqrt{\omega_2} P_2^{(n+l)} & \sqrt{\omega_2} P_2^{(i+1)} & \cdots & \sqrt{\omega_2} P_2^{(n)} \\ \sqrt{\omega_3} P_3^{(1)} & \cdots & \sqrt{\omega_3} P_3^{(i-1)} & \sqrt{\omega_3} P_3^{(n+l)} & \sqrt{\omega_3} P_3^{(i+1)} & \cdots & \sqrt{\omega_3} P_3^{(n)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \sqrt{\omega_{n_o}} P_{n_o}^{(1)} & \cdots & \sqrt{\omega_{n_o}} P_{n_o}^{(i-1)} & \sqrt{\omega_{n_o}} P_{n_o}^{(n+l)} & \sqrt{\omega_{n_o}} P_{n_o}^{(i+1)} & \cdots & \sqrt{\omega_{n_o}} P_{n_o}^{(n)} \end{bmatrix} \quad (1.34)$$

Utilizando as relações (1.31), temos que os elementos $\sqrt{\omega_k} P_k^{(j)} = h^{N_d(k)} u_{j,k}$, para $j \neq i$ (ou seja, os elementos em colunas diferentes à i -ésima coluna), e os elementos na i -ésima coluna são expressos como:

$$\sqrt{\omega_k} P_k^{(n+l)} = h^{N(n+l)} \sqrt{\omega_k} (\alpha_k^1)^{\mu_1(n+l)} \cdots (\alpha_k^d)^{\mu_d(n+l)} = h^{N_d(n+l)} u_{n+l,k} \quad (1.35)$$

Então, a matriz \mathbf{R}_i pode ser escrita como o produto da matriz $\mathbf{R}'_{i[n_o \times n]}$, composta pelos elementos $u_{j,k}$ para $j \neq i$, e $u_{n+l,k}$ para $i = j$, com uma matriz diagonal quadrada $\mathbf{H}'_{[n \times n]}$, cujos elementos na diagonal são dados por: $\text{diag}(\mathbf{H}') = \{h^{N_d(1)}, \dots, h^{N_d(i-1)}, h^{N_d(n+l)}, h^{N_d(i+1)}, \dots, h^{N_d(n)}\}$. Assim:

$$\mathbf{R}_i = \mathbf{R}'_i \mathbf{H}'$$

Portanto, $\mathbf{K}_i = \mathbf{P} \mathbf{R}_i = (\mathbf{H} \mathbf{P}') (\mathbf{R}'_i \mathbf{H}') = \mathbf{H} \mathbf{P}' \mathbf{R}'_i \mathbf{H}'$, e calculando seu determinante teremos,

$$\det(\mathbf{K}_i) = \det(\mathbf{H} \mathbf{P}' \mathbf{R}'_i \mathbf{H}') = \det(\mathbf{H}) \det(\mathbf{P}' \mathbf{R}'_i) \det(\mathbf{H}')$$

como \mathbf{H} e \mathbf{H}' são matrizes diagonais quadradas, seus determinantes podem ser facilmente calculados como o produto dos elementos das diagonais. Como a matriz \mathbf{H}' é diferente da matriz \mathbf{H} em 1 só elemento, o determinante da matriz \mathbf{K}_i torna-se

$$\det(\mathbf{K}_i) = h^{\gamma - N_d(i) + N_d(n+l)} \det(\mathbf{P}' \mathbf{R}'_i) \quad \text{com} \quad \gamma = 2 \sum_{p=1}^n N_d(p) \quad (1.36)$$

Portanto temos a seguinte proposição:

Proposição 1.2. *Os elementos $T_{i,n+l}$ podem ser escritos como:*

$$T_{i,n+l} = \frac{h^{N_d(n+l)-N_d(i)}}{\pi(n+l)} e_i^{(n+l)} [D_{(n+l)}f(\mathbf{r}_o)] \quad (1.37)$$

onde $e_i^{(n+l)}$ é definida como:

$$e_i^{(n+l)} = \frac{\det(\mathbf{P}'\mathbf{R}'_i)}{\det(\tilde{\mathbf{A}})}$$

Demonstração. A partir da expressão (1.32) e (1.36), obtemos, ao substituir na equação (1.29) do lema 1.2:

$$\begin{aligned} T_{i,n+l} &= \frac{1}{\pi(n+l)} \frac{h^{\gamma-N_d(i)+N_d(n+l)} \det(\mathbf{P}'\mathbf{R}'_i)}{h^\gamma \det(\tilde{\mathbf{A}})} [D_{(n+l)}f(\mathbf{r}_o)] \\ &= \frac{h^{N_d(n+l)-N_d(i)}}{\pi(n+l)} \frac{\det(\mathbf{P}'\mathbf{R}'_i)}{\det(\tilde{\mathbf{A}})} [D_{(n+l)}f(\mathbf{r}_o)] \end{aligned} \quad (1.38)$$

□

Corolário 1.2. *Devido ao fato que as matrizes $\tilde{\mathbf{A}}$ e $\mathbf{P}'\mathbf{R}'_i$ tem todos seus termos matriciais iguais, com exceção dos termos da i -ésima coluna de $\mathbf{P}'\mathbf{R}'_i$, onde cada um deles é dado como $g_j^{(n+l)} = \sum_{i'=1}^k u_{j,i'} u_{n+l,i'}$, os quais podem compôr um novo vetor $\mathbf{g}^{(n+l)}$. Assim, o vetor $\mathbf{e}^{(n+l)} = [e_1^{(n+l)}, \dots, e_n^{(n+l)}]^T$, pode ser calculado, solucionando o sistema linear:*

$$\tilde{\mathbf{A}}\mathbf{e}^{(n+l)} = \mathbf{g}^{(n+l)}$$

Para que c_i seja uma aproximação consistente, os termos do erro de truncamento $\bar{T}_{i,n}$ devem se anular quando o parâmetro de comprimento h da malha tender a zero, ou seja:

$$\lim_{h \rightarrow 0} \bar{T}_{i,n} = 0 \quad \implies \quad c_i \text{ é consistente.}$$

então, da relação (1.37) na proposição 1.2, deve-se cumprir:

$$N_d(i) < N_d(n+l) \quad \forall l = 1, \dots, \infty \quad (1.39)$$

Exemplo 1.7. *Seja uma malha estruturada bi-dimensional, onde um nó v_o na posição $(0,0)$ tem uma célula $C_o = \{v_N, v_S, v_W, v_E\}$, tal como é apresentado na figura 1.10a, onde $-\pi/2 \leq \theta \leq \pi/2$, $\rho_{o,N} = \rho_{o,W} = \rho_{o,S} = \rho_{o,E} = h$, e o nó v_N está fixo na posição $(0, h)$.*

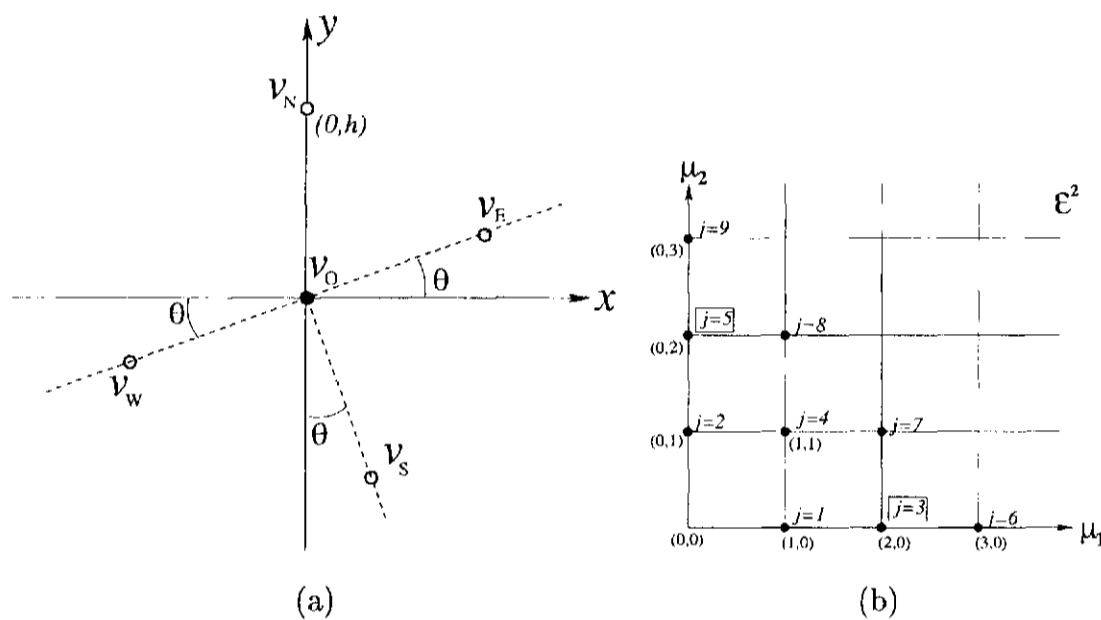


Figura 1.10: (a) Célula do nó v_o . (b) Representação do espaço E^2 .

Suponhamos que estamos interessados em calcular as derivadas de segunda ordem $\partial^2 f / \partial x^2$ e $\partial^2 f / \partial y^2$ de uma função f sob nó v_o (utilizados por exemplo para solucionar a equação de Laplace). Utilizando a numeração dos monômios e dos operadores diferenciais dados pela representação no espaço E^2 , tal como se apresenta na figura 1.10b; assim mediante (1.19) serão definidos os operadores diferenciais:

$$D_3 = \frac{\partial^{N_2(3)}}{(\partial x)^{\mu_1(3)}(\partial y)^{\mu_2(3)}} = \frac{\partial^2}{\partial x^2} \quad e \quad D_5 = \frac{\partial^{N_2(5)}}{(\partial x)^{\mu_1(5)}(\partial y)^{\mu_2(5)}} = \frac{\partial^2}{\partial y^2}$$

onde $(\mu_1(3), \mu_2(3)) = (2, 0)$ e $(\mu_1(5), \mu_2(5)) = (0, 2)$, $N_2(3) = N_2(5) = 2$. Aplicando o MDFG com um polinômio de aproximação W_o ajustado por mínimos quadrados; pela equação (1.25) obtemos:

$$c_3 = \frac{1}{2} D_3 f(0,0) + \bar{T}_{3,n} \quad , e \quad c_5 = \frac{1}{2} D_5 f(0,0) + \bar{T}_{5,n}$$

onde $\frac{1}{\pi(3)} = \frac{1}{2!0!} = \frac{1}{2}$, e $\frac{1}{\pi(5)} = \frac{1}{0!2!} = \frac{1}{2}$. Os erros de truncamento $\bar{T}_{3,n}$ e $\bar{T}_{5,n}$ das aproximações anteriores dependerão do polinômio W_o utilizado. Como a célula é composta por quatro nós, podemos utilizar no máximo quatro monômios $P^{(j)}$ para criar W_o , nos quais devem estar os monômios $P^{(3)}$ e $P^{(5)}$.

a. Seja o polinômio de aproximação dado por:

$$W_o = c_3 P^{(3)} + c_5 P^{(5)}$$

Assim, ao aplicar mínimos quadrados para ajustar a função \bar{f} , o sistema linear que soluciona (1.8), será de 2 ordem. As soluções deste sistema linear são dadas por:

$$c_3 = \frac{1}{2} D_3 f(0,0) + T_{3,1} + T_{3,2} + T_{3,4} + T_{3,6} + T_{3,7} + \dots$$

$$c_5 = \frac{1}{2} D_5 f(0,0) + T_{5,1} + T_{5,2} + T_{5,4} + T_{5,6} + T_{5,7} + \dots$$

da proposição 1.2 obtemos

$$c_3 = \frac{1}{2} D_3 f + \frac{1}{h} e_3^{(1)} D_1 f + \frac{1}{h} e_3^{(2)} D_2 f + e_3^{(4)} D_4 f + \frac{h}{6} e_3^{(6)} D_6 f + \frac{h}{2} e_3^{(7)} D_7 f + O(h)$$

$$c_5 = \frac{1}{2} D_5 f + \frac{1}{h} e_5^{(1)} D_1 f + \frac{1}{h} e_5^{(2)} D_2 f + e_5^{(4)} D_4 f + \frac{h}{6} e_5^{(6)} D_6 f + \frac{h}{2} e_5^{(7)} D_7 f + O(h)$$

onde f é avaliado no ponto $(0,0)$, e os elementos $e_3^{(1)}$, $e_3^{(2)}$, $e_3^{(4)}$, $e_5^{(1)}$, $e_5^{(2)}$ e $e_5^{(4)}$ são apresentados na figura 1.11 como função de θ .

b. Se o polinômio de aproximação é dado por:

$$W_o = c_1 P^{(1)} + c_2 P^{(2)} + c_3 P^{(3)} + c_5 P^{(5)}$$

então aplicando mínimos quadrados para ajustar a função \bar{f} , o sistema linear que soluciona (1.8), é de 4 ordem. As soluções para os termos c_3 e c_5 deste sistema linear são dados por,

$$c_3 = \frac{1}{2} D_3 f(0,0) + T_{3,4} + T_{3,6} + T_{3,7} + T_{3,8} + T_{3,9} + \dots$$

$$c_5 = \frac{1}{2} D_5 f(0,0) + T_{5,4} + T_{5,6} + T_{5,7} + T_{5,8} + T_{5,9} + \dots$$

da proposição 1.2 obtemos

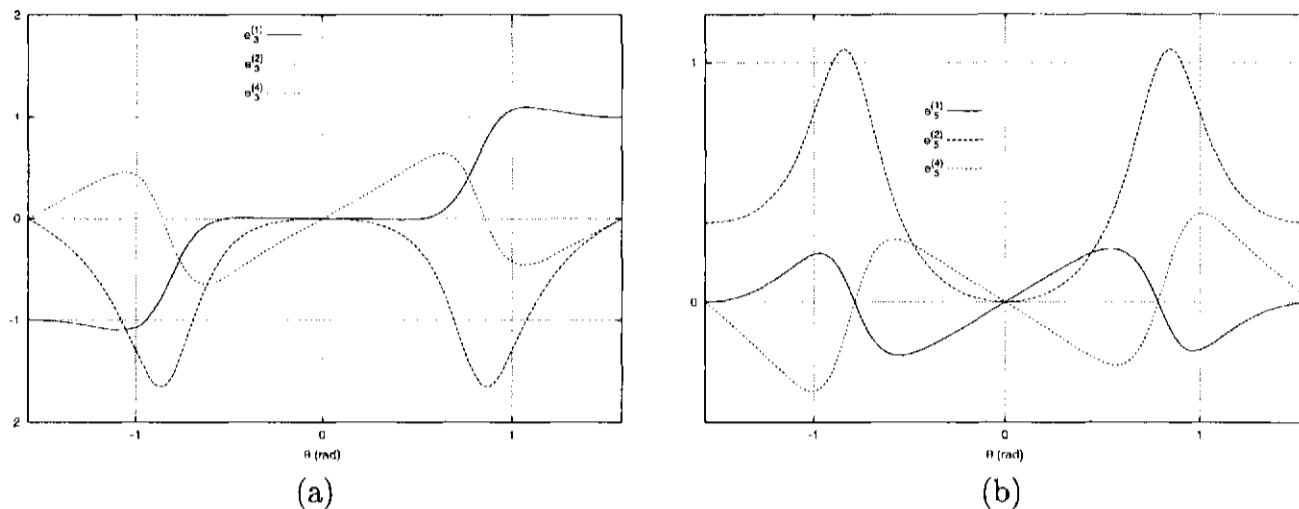


Figura 1.11: (a) Curvas dos elementos $e_3^{(1)}$, $e_3^{(2)}$ e $e_3^{(4)}$ para c_3 . (b) Curvas dos elementos $e_5^{(1)}$, $e_5^{(2)}$ e $e_5^{(4)}$ para c_5 .

$$c_3 = \frac{1}{2}D_3f + e_3^{(4)}D_4f + \frac{h}{6}e_3^{(6)}D_6f + \frac{h}{2}e_3^{(7)}D_7f + \frac{h}{2}e_3^{(8)}D_8f + \frac{h}{6}e_3^{(9)}D_9f + O(h^2)$$

$$c_5 = \frac{1}{2}D_5f + e_5^{(4)}D_4f + \frac{h}{6}e_5^{(6)}D_6f + \frac{h}{2}e_5^{(7)}D_7f + \frac{h}{2}e_5^{(8)}D_8f + \frac{h}{6}e_5^{(9)}D_9f + O(h^2)$$

substituindo os valores de $e_3^{(4)}$ e $e_5^{(4)}$ e aplicando o limite em ambas expressões, obtemos:

$$\lim_{h \rightarrow 0} c_3 = \frac{1}{2}D_3f + e_3^{(4)}D_4f = \frac{1}{2}D_3f - \frac{\sin(\theta)(1 + \cos(\theta))}{\sin^2(\theta) - \cos(\theta) - \cos^2(\theta)}D_4f$$

$$\lim_{h \rightarrow 0} c_5 = \frac{1}{2}D_5f + e_5^{(4)}D_4f = \frac{1}{2}D_5f + \frac{\sin(\theta)\cos(\theta)}{\sin^2(\theta) - \cos(\theta) - \cos^2(\theta)}D_4f$$

portanto, as aproximações c_3 e c_5 têm consistência somente quando $e_3^{(4)} = e_5^{(4)} = 0$, ou seja, para um ângulo $\theta = 0$, com um erro $O(h^2)$ (já que $e_3^{(j)} = e_5^{(j)} = 0$, para $j = 6, 7, 8, 9$).

Os resultados obtidos no exemplo anterior nos indica que para $\theta \neq 0$ a aplicação do método não é boa, devido a que o polinômio de aproximação W_o não utiliza todos os monômios da base do espaço \mathcal{P}_2^2 . No seguinte exemplo é apresentada uma aproximação consistente utilizando um polinômio de aproximação $W_o \in \mathcal{P}_5^d$.

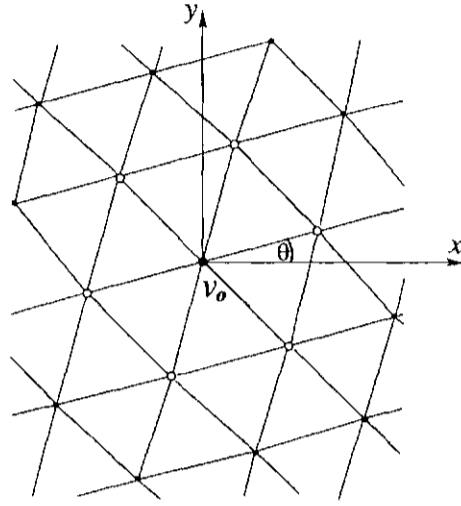


Figura 1.12: Malha triangular regular bi-dimensional (onde todas as arestas tem comprimento h) com uma rotação num ângulo θ .

Exemplo 1.8. *Suponha agora um problema bi-dimensional onde é utilizada uma malha triangular regular (ver figura 1.12) para discretizar o domínio.*

Pelo MDFG pode-se por exemplo, encontrar os valores das derivadas $\partial f/\partial x$, $\partial^2 f/\partial x \partial y$, e $\partial^2 f/\partial y^2$ de uma função f , utilizando uma célula C_o tipo $C(1)$ para um nó qualquer v_o , composta por 6 nós. Utilizando-se um polinômio de aproximação $W_o \in \mathcal{P}_2^2$:

$$W_o = c_1 P^{(1)} + c_2 P^{(2)} + c_3 P^{(3)} + c_4 P^{(4)} + c_5 P^{(5)}$$

A partir da proposição 1.2, obtemos as aproximações das derivadas:

$$c_1 = D_1 f + \frac{h^2}{6} e_1^{(6)} D_6 f + \frac{h^2}{2} e_1^{(7)} D_7 f + \frac{h^2}{2} e_1^{(8)} D_8 f + \frac{h^2}{6} e_1^{(9)} D_9 f + O(h^3)$$

$$c_4 = D_4 f + \frac{h}{6} e_4^{(6)} D_6 f + \frac{h}{2} e_4^{(7)} D_7 f + \frac{h}{2} e_4^{(8)} D_8 f + \frac{h}{6} e_4^{(9)} D_9 f + O(h^2)$$

$$c_5 = \frac{1}{2} D_5 f + \frac{h}{6} e_5^{(6)} D_6 f + \frac{h}{2} e_5^{(7)} D_7 f + \frac{h}{2} e_5^{(8)} D_8 f + \frac{h}{6} e_5^{(9)} D_9 f + O(h^2)$$

onde

$$\mathbf{e}^{(6)} = [3/4, 0, 0, 0, 0]^T \quad \mathbf{e}^{(7)} = [0, 1/4, 0, 0, 0]^T$$

$$\mathbf{e}^{(8)} = [1/4, 0, 0, 0, 0]^T \quad \mathbf{e}^{(9)} = [0, 3/4, 0, 0, 0]^T$$

para qualquer ângulo de rotação θ . Ao aplicar o limite para $h \rightarrow 0$, obtemos:

$$\lim_{h \rightarrow 0} c_1 = \lim_{h \rightarrow 0} \left(D_1 f + \frac{h^2}{8} D_6 f + \frac{h^2}{8} D_8 f + O(h^3) \right) = D_1 f = \frac{\partial f}{\partial x}$$

$$\lim_{h \rightarrow 0} c_4 = \lim_{h \rightarrow 0} (D_4 f + O(h^2)) = D_4 f = \frac{\partial^2 f}{\partial x \partial y}$$

$$\lim_{h \rightarrow 0} c_5 = \lim_{h \rightarrow 0} \left(\frac{1}{2} D_5 f + O(h^2) \right) = \frac{1}{2} D_5 f = \frac{1}{2} \frac{\partial^2 f}{\partial y^2}$$

Portanto, estas aproximações têm consistência, e um erro de ordem $O(h^2)$.

1.4 Estabilidade do método.

Nesta seção será estudada a estabilidade do MDFG através do método de Von Neumann, em EDP lineares de segunda ordem. Por simplicidade vamos trabalhar somente com problemas bi-dimensionais de equações diferenciais hiperbólicas e parabólicas. Seja $f = f(x, y, t)$ uma função, e uma EDP linear homogênea definida como:

$$\varphi \left(\frac{\partial f}{\partial t}, \frac{\partial^2 f}{\partial t^2}, \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial^2 f}{\partial x^2}, \frac{\partial^2 f}{\partial x \partial y}, \frac{\partial^2 f}{\partial y^2} \right) = 0 \quad (1.40)$$

a qual não depende dos termos x, y, t, f . Considerando unicamente as EDP com dependência temporal de primeira ordem, e φ linear, podemos escrever (1.40) como:

$$\frac{\partial f}{\partial t} = \lambda_1 \frac{\partial f}{\partial x} + \lambda_2 \frac{\partial f}{\partial y} + \lambda_3 \frac{\partial^2 f}{\partial x^2} + \lambda_4 \frac{\partial^2 f}{\partial y \partial x} + \lambda_5 \frac{\partial^2 f}{\partial y^2} \quad (1.41)$$

Substituindo as diferenças finitas num nó v_o pelas aproximações do MDFG dadas por (1.15) utilizando polinômios de aproximação $W_o \in \mathcal{P}_2^2$ (compostos por 5 monômios base $P^{(j)}$), e utilizando diferença progressiva para o termo temporal, com um esquema explícito, obtemos:

$$\frac{f_o^{n+1} - f_o^n}{\Delta t} = \sum_{l=1}^5 \lambda_l c_l = \sum_{l=1}^5 \sum_{j=1}^5 \left(\sum_{v_k \in C_o} [f_k^n - f_o^n] P_k^{(j)} \omega_k \right) \lambda_l S_j^{(l)} \quad (1.42)$$

Suponhamos que a função f_o^n é perturbada por um pequeno erro ϵ_o . Portanto, dizemos que o valor da função perturbada \tilde{f}_o^n é composto pela solução exata mais o erro, $\tilde{f}_o^n = f_o^n + \epsilon_o^n$. Substituindo a função \tilde{f}_o^n na equação (1.42), e fatorando obtemos:

$$\begin{aligned} \frac{f_o^{n+1} - f_o^n}{\Delta t} + \frac{\epsilon_o^{n+1} - \epsilon_o^n}{\Delta t} &= \sum_{l=1}^5 \sum_{j=1}^5 \left(\sum_{v_k \in C_o} [f_k^n - f_o^n + \epsilon_k^n - \epsilon_o^n] P_k^{(j)} \omega_k \right) \lambda_l S_j^{(l)} \\ &= \sum_{l=1}^5 \sum_{j=1}^5 \left(\sum_{v_k \in C_o} [f_k^n - f_o^n] P_k^{(j)} \omega_k \right) \lambda_l S_j^{(l)} + \sum_{l=1}^5 \sum_{j=1}^5 \left(\sum_{v_k \in C_o} [\epsilon_k^n - \epsilon_o^n] P_k^{(j)} \omega_k \right) \lambda_l S_j^{(l)} \end{aligned} \quad (1.43)$$

de (1.42), obtemos:

$$\frac{\epsilon_o^{n+1} - \epsilon_o^n}{\Delta t} = \sum_{l=1}^5 \sum_{j=1}^5 \left(\sum_{v_k \in C_o} [\epsilon_k^n - \epsilon_o^n] P_k^{(j)} \omega_k \right) \lambda_l S_j^{(l)} \quad (1.44)$$

Decompondo o erro em modos de Fourier e supondo que o domínio é periódico, obtemos:

$$\epsilon_o^n = \sum_{r_x} \sum_{r_y} \epsilon_o^n(r_x, r_y) e^{i(r_x x_o^1 + r_y x_o^2)}$$

podemos tomar cada modo (r_x, r_y) separadamente já que (1.44) é linear e os modos (r_x, r_y) são independentes uns dos outros. Assim, para o nó v_o podemos escrever:

$$\epsilon_o^n = \epsilon_o^n e^{i(r_x x_o^1 + r_y x_o^2)}$$

onde ϵ_o^n é a amplitude do modo (r_x, r_y) (por simplicidade notamos $\epsilon_o^n(r_x, r_y)$ como ϵ_o^n). Então os erros nos $v_k \in C_o$ são escritos como:

$$\epsilon_k^n = \epsilon_o^n e^{i(r_x x_o^1 + r_y x_o^2)} e^{i(r_x \alpha_k^1 h + r_y \alpha_k^2 h)}$$

sustituindo em (1.44) e eliminando os termos $e^{i(r_x x_o^1 + r_y x_o^2)}$ em ambas partes da expressão, obtemos:

$$\frac{\varepsilon_o^{n+1} - \varepsilon_o^n}{\Delta t} = \varepsilon_o^n \sum_{l=1}^5 \sum_{j=1}^5 \left(\sum_{v_k \in C_o} [e^{i(r_x \alpha_k^1 h + r_y \alpha_k^2 h)} - 1] P_k^{(j)} \omega_k \right) \lambda_l S_j^{(l)} \quad (1.45)$$

expressando a relação dependendo da magnitude dos erros temos,

$$\frac{\varepsilon_o^{n+1}}{\varepsilon_o^n} = 1 + \Delta t \sum_{l=1}^5 \sum_{j=1}^5 \left(\sum_{v_k \in C_o} [e^{i(r_x \alpha_k^1 h + r_y \alpha_k^2 h)} - 1] P_k^{(j)} \omega_k \right) \lambda_l S_j^{(l)} \quad (1.46)$$

O segundo termo da expressão acima pode ser escrito utilizando a regra de Laplace e de Cramer, como:

$$\Delta t \sum_{l=1}^5 \sum_{j=1}^5 \left(\sum_{v_k \in C_o} [e^{i(r_x \alpha_k^1 h + r_y \alpha_k^2 h)} - 1] P_k^{(j)} \omega_k \right) \lambda_l S_j^{(l)} = \frac{\Delta t}{\det(\mathbf{A})} \sum_{l=1}^5 \lambda_l \det(\mathbf{A}^{(l)}) \quad (1.47)$$

onde os elementos da matriz \mathbf{A} são dados em (1.9), e os elementos da matriz $\mathbf{A}^{(l)}$ são iguais aos da matriz \mathbf{A} exceto pela l -ésima coluna, assim, os elementos da matriz $\mathbf{A}^{(l)}$ são dados como:

$$A_{p,q}^{(l)} = \sum_{k \in C_o} P_k^{(p)} (\xi_k \delta_{q,l} + P_k^{(q)} (1 - \delta_{q,l})) \omega_k \quad (1.48)$$

onde $\delta_{q,l}$ é a função delta de Kronecker, e $\xi_k = [e^{i(r_x \alpha_k^1 h + r_y \alpha_k^2 h)} - 1]$. Como todos os termos matriciais de $\mathbf{A}^{(l)}$ são compostos por uma soma sobre os vértices vizinhos a v_o , a matriz pode ser expressa como a soma de diferentes matrizes $\mathbf{A}_k^{(l)}$, cada uma relativa a um vértice v_k da célula C_o ,

$$\mathbf{A}^{(l)} = \sum_{v_k \in C_o} \mathbf{A}_k^{(l)}, \quad (1.49)$$

cada uma das matrizes $\mathbf{A}_k^{(l)}$ pode ser decomposta como o produto $\mathbf{A}_k^{(l)} = \mathbf{M}_k \mathbf{N}_k^{(l)}$, onde a matriz \mathbf{M}_k $[5 \times 5]$ é diagonal, formada por $\text{diag}(\mathbf{M}_k) = \{P_k^{(1)} \sqrt{\omega_k}, \dots, P_k^{(5)} \sqrt{\omega_k}\}$, e a matriz $\mathbf{N}_k^{(l)}$ $[5 \times 5]$ tem como componentes aos elementos $N_{k,p,q}^{(l)} = (\xi_k \delta_{q,l} + P_k^{(q)} (1 - \delta_{q,l})) \sqrt{\omega_k}$, para $p, q = 1, \dots, 5$. Utilizando as relações (1.31) a matriz \mathbf{M}_k pode ser expressa como o

produto de duas matrizes diagonais \mathbf{HM}'_k , formadas por $\text{diag}(\mathbf{H}) = \{h^{N(1)}, \dots, h^{N(5)}\}$, e $\text{diag}(\mathbf{M}'_k) = \{u_{1,k}, \dots, u_{5,k}\}$; e a matriz $\mathbf{N}'_k^{(l)}$ é formada pelo produto $\mathbf{N}'_k^{(l)} \mathbf{H}''^{(l)}$, onde as componentes da matriz $\mathbf{N}'_k^{(l)}$ são dadas como $N'_{k,p,q}{}^{(l)} = (\xi_k \sqrt{\omega_k} \delta_{q,l} + u_{q,k}(1 - \delta_{q,l}))$, e a matriz $\mathbf{H}''^{(l)}$ é diagonal, formada pelos elementos $\text{diag}(\mathbf{H}''^{(l)}) = \{h^{N(1)}, \dots, h^{N(l-1)}, 1, h^{N(l+1)}, \dots, h^{N(5)}\}$. Em resumo, para um dado nó $v_k \in C_o$, temos que a matriz $\mathbf{A}_k^{(l)}$ pode ser decomposta como:

$$\mathbf{A}_k^{(l)} = \mathbf{M}_k \mathbf{N}'_k^{(l)} = (\mathbf{HM}'_k)(\mathbf{N}'_k^{(l)} \mathbf{H}''^{(l)})$$

como as matrizes \mathbf{H} e $\mathbf{H}''^{(l)}$ são independentes do nó da célula, ao somar todas as matrizes $\mathbf{A}_k^{(l)}$ para todos os nós da célula, obtemos:

$$\mathbf{A}^{(l)} = \sum_{v_k \in C_o} \mathbf{HM}'_k \mathbf{N}'_k^{(l)} \mathbf{H}''^{(l)} = \mathbf{H} \left(\sum_{v_k \in C_o} \mathbf{M}'_k \mathbf{N}'_k^{(l)} \right) \mathbf{H}''^{(l)} = \mathbf{H} \Phi^{(l)} \mathbf{H}''^{(l)}$$

onde a matriz

$$\Phi^{(l)} = \sum_{v_k \in C_o} \mathbf{M}'_k \mathbf{N}'_k^{(l)}$$

Portanto, o determinante da matriz $\mathbf{A}^{(l)}$ é dado como:

$$\det(\mathbf{A}^{(l)}) = \det(\mathbf{H} \Phi^{(l)} \mathbf{H}''^{(l)}) = \det(\mathbf{H}) \det(\Phi^{(l)}) \det(\mathbf{H}''^{(l)}) \quad (1.50)$$

onde $\det(\mathbf{H}) = h^{\gamma/2}$, e $\det(\mathbf{H}''^{(l)}) = h^{\gamma/2 - N^{(l)}}$, com γ definida em (1.32). Então, da expressão (1.50) obtemos:

$$\det(\mathbf{A}^{(l)}) = h^{\gamma - N^{(l)}} \det(\Phi^{(l)})$$

utilizando o resultado do determinante da matriz \mathbf{A} dado em (1.32), substituímos em (1.47) e (1.46) obtendo:

$$\frac{\varepsilon_o^{n+1}}{\varepsilon_o^n} = 1 + \Delta t \sum_{l=1}^5 \frac{h^{\gamma - N^{(l)}}}{h^{\gamma}} \lambda_l \frac{\det(\Phi^{(l)})}{\det(\tilde{\mathbf{A}})} = 1 + \Delta t \sum_{l=1}^5 \frac{\lambda_l}{h^{N^{(l)}}} \frac{\det(\Phi^{(l)})}{\det(\tilde{\mathbf{A}})} \quad (1.51)$$

Seja o sistema linear dado a partir da matriz $\tilde{\mathbf{A}}$:

$$\tilde{\mathbf{A}} \mathbf{s} = \mathbf{g}' \quad (1.52)$$

onde $\mathbf{s} = [s_1, \dots, s_5]^T$, e $\mathbf{g}' = [g'_1, \dots, g'_5]^T$, com $g'_j = \sum_{k \in C_o} \xi_k u_{j,k} \sqrt{\omega_k}$ para $j = 1, \dots, 5$. Aplicando a regra de Cramer para solucionar este sistema linear, cada um dos elementos s_j do vetor incógnita \mathbf{s} será dado pela relação entre os determinantes das matrizes $\tilde{\mathbf{A}}$, e a matriz com todos os elementos iguais à matriz $\tilde{\mathbf{A}}$, exceto os elementos da j -ésima coluna, os quais são substituídos pelos elementos do vetor \mathbf{g}' . Esta última matriz corresponde à matriz $\Phi^{(j)}$, portanto o elemento s_j é dado como:

$$s_j = \frac{\det(\Phi^{(j)})}{\det(\tilde{\mathbf{A}})}$$

assim, ao substituir na equação (1.51), teremos:

$$\frac{\varepsilon_o^{n+1}}{\varepsilon_o^n} = 1 + \Delta t \sum_{l=1}^5 \frac{\lambda_l}{h^{N(l)}} s_l \quad (1.53)$$

Quando o método numérico é estável, o erro diminui do passo n ao $n + 1$, ou seja $\varepsilon_o^{n+1} < \varepsilon_o^n$. Definindo como o fator de amplificação G o valor absoluto da relação $G = |\varepsilon_o^{n+1}/\varepsilon_o^n|$, o método numérico é estável se $G < 1$. Podemos escrever (1.53) a partir do fator de amplificação, ficando como:

$$G = \left| 1 + \Delta t \sum_{l=1}^5 \frac{\lambda_l s_l}{h^{N(l)}} \right| \quad (1.54)$$

Exemplo 1.9. Analisar a estabilidade do MDFG num problema advectivo utilizando células hexagonais regulares tipo $C(I)$, numa malha triangular regular (como a dada no exemplo 1.8, com $\theta = 0$), e como uma função peso $\omega_k = 1$.

A equação linear advectiva é dada como:

$$\frac{\partial f}{\partial t} + U_x \frac{\partial f}{\partial x} + U_y \frac{\partial f}{\partial y} = 0$$

assim, temos que $\lambda_1 = -U_x$, $\lambda_2 = -U_y$, $\lambda_3 = 0$, $\lambda_4 = 0$ e $\lambda_5 = 0$. Calculando o vetor \mathbf{g}' numa célula hexagonal regular teremos:

$$\mathbf{g}' = \begin{pmatrix} 2i \left(\sin(r_x) + \sin\left(\frac{1}{2}r_x\right) \cos\left(\frac{\sqrt{3}}{2}r_y\right) \right) \\ 2\sqrt{3} i \cos\left(\frac{1}{2}r_x\right) \sin\left(\frac{\sqrt{3}}{2}r_y\right) \\ 2 \cos(r_x) - 3 + \cos\left(\frac{1}{2}r_x\right) \cos\left(\frac{\sqrt{3}}{2}r_y\right) \\ -\sqrt{3} \sin\left(\frac{1}{2}r_x\right) \sin\left(\frac{\sqrt{3}}{2}r_y\right) \\ 3 \cos\left(\frac{1}{2}r_x\right) \cos\left(\frac{\sqrt{3}}{2}r_y\right) - 3 \end{pmatrix} \quad (1.55)$$

Solucionando o sistema linear (1.52), e pegando os dois termos s_1 e s_2 , dados como:

$$s_1 = \frac{2}{3}i \left(\sin(r_x) + \sin\left(\frac{1}{2}r_x\right) \cos\left(\frac{\sqrt{3}}{2}r_y\right) \right), \quad s_2 = \frac{2}{3}i \cos\left(\frac{1}{2}r_x\right) \sin\left(\frac{\sqrt{3}}{2}r_y\right)$$

o fator de amplificação é:

$$G = \left| 1 - \frac{2}{3}i \frac{\Delta t}{h} \left[U_x \left(\sin(r_x) + \sin\left(\frac{1}{2}r_x\right) \cos\left(\frac{\sqrt{3}}{2}r_y\right) \right) + U_y \cos\left(\frac{1}{2}r_x\right) \sin\left(\frac{\sqrt{3}}{2}r_y\right) \right] \right|$$

do qual podemos concluir, que para qualquer modo (r_x, r_y) , o fator de amplificação G nunca será menor que um. Portanto o sistema é incondicionalmente instável.

Exemplo 1.10. Estudar a estabilidade do MDFG numa equação difusiva, utilizando uma malha como a do exemplo anterior, e uma função peso $\omega_k = 1$.

A equação linear de difusão é dada por:

$$\frac{\partial f}{\partial t} - \nu \left(\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \right) = 0$$

assim, $\lambda_1 = 0$, $\lambda_2 = 0$, $\lambda_3 = 2\nu$, $\lambda_4 = 0$ e $\lambda_5 = 2\nu$. Aproveitando os resultados do exemplo anterior, devido á matriz $\mathbf{P}'\mathbf{P}'^T$ e ao valor g' dependerem unicamente das posições dos nós da célula. Os elementos s_3 e s_5 são dados como:

$$s_3 = \cos(r_x) - 1 \quad , \quad s_5 = \frac{4}{3} \cos\left(\frac{1}{2}r_x\right) \cos\left(\frac{\sqrt{3}}{2}r_y\right) - \frac{1}{3} \cos\left(\frac{\sqrt{3}}{2}r_x\right) - 1$$

então, o fator de amplificação é dado como:

$$G = \left| 1 + \frac{2\Delta t \nu}{h^2} \left(\cos(r_x) + \frac{4}{3} \cos\left(\frac{1}{2}r_x\right) \cos\left(\frac{\sqrt{3}}{2}r_y\right) - \frac{1}{3} \cos\left(\frac{\sqrt{3}}{2}r_x\right) - 2 \right) \right| \quad (1.56)$$

Chamando a função :

$$\eta(r_x, r_y) = \cos(r_x) + \frac{4}{3} \cos\left(\frac{1}{2}r_x\right) \cos\left(\frac{\sqrt{3}}{2}r_y\right) - \frac{1}{3} \cos\left(\frac{\sqrt{3}}{2}r_x\right) - 2$$

com seus valores máximos e mínimos dados como:

$$\eta_{\min} = -3.123 \quad , \quad \eta_{\max} = -0.007$$

para que o MDFG seja estável, o fator de amplificação G deve ser menor que um, portanto, tem que satisfazer as seguintes condições:

$$(1) \quad \frac{2\Delta t \nu}{h^2} \eta_{\max} < 0 \quad (2) \quad -2 < \frac{2\Delta t \nu}{h^2} \eta_{\min}$$

a primeira condição é sempre válida ao ser $\Delta t, \nu, h^2 \geq 0$. Para satisfazer a segunda condição, devemos ter:

$$\frac{\Delta t \nu}{h^2} < -\frac{1}{\eta_{\min}} \approx 0.32$$

assim, a solução da equação é condicionalmente estável para células $C(I)$.

Nestes exemplos foi utilizada a função peso uniforme $\omega = 1$. Acreditamos que utilizando funções peso que não sejam isotrópicas, dando preferencia aos nós que estão em

1.4. ESTABILIDADE DO MÉTODO.

41

direções privilegiadas, o método pode apresentar estabilidade na solução de problemas como os hiperbólicos. Este estudo pretende-se realizar num trabalho posterior.

Capítulo 2

Implementação do MDFG.

Este capítulo tem como objetivo apresentar uma implementação do MDFG como método numérico para solucionar EDP's. A implementação do MDFG é constituída pelos seguintes passos: 1. Análise da EDP a ser solucionada, e a escolha do espaço dos polinômios de aproximação mais conveniente; 2. criação de uma boa distribuição de nós sob o domínio (geração de uma malha), e escolha das células apropriadas para cada nó; 3. obtenção das aproximações e solução dos sistemas de equações discretas. Destes aspectos da implementação, os dois primeiros passos são analisados neste capítulo, e o terceiro será apresentado no capítulo 3, aplicado na solução de equações diferenciais específicas.

2.1 Análise das EDP em duas dimensões.

Por simplicidade vamos considerar problemas bi-dimensionais, assim, as EDP serão equações envolvendo duas variáveis independentes x e y e derivadas parciais de uma função real $f = f(x, y)$. A forma mais geral de uma EDP bi-dimensional de ordem s é:

$$\varphi \left(x, y, f, \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial^2 f}{\partial x^2}, \frac{\partial^2 f}{\partial x \partial y}, \frac{\partial^2 f}{\partial y^2}, \dots, \frac{\partial^s f}{\partial x^s} \right) = 0$$

É bastante comum o caso de problemas práticos importantes onde a equação diferencial parcial é de ordem $s = 2$ e linear nas derivadas de ordem 2, portanto:

$$a \frac{\partial^2 f}{\partial x^2} + b \frac{\partial^2 f}{\partial x \partial y} + c \frac{\partial^2 f}{\partial y^2} = \varphi \left(x, y, f, \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right) \quad (2.1)$$

Além disso, uma equação diferencial que representa um modelo de um problema físico, terá adicionalmente condições de fronteira e condições iniciais (se for um problema de propagação). Geralmente, as condições iniciais e as condições de fronteira são funções que envolvem a função f e suas derivadas de primeira ordem no domínio D e na fronteira ∂D respectivamente.

Ao solucionar numericamente uma equação como a (2.1), o domínio D deve inicialmente ser representado por um conjunto de nós V , sobre tais pontos utilizamos o MDFG para discretizar as EDP, as quais são solucionadas utilizando as condições iniciais e as condições de fronteira.

Para cada nó $v_o \in V$, devemos procurar um polinômio de aproximação W_o a partir do qual, possamos obter os elementos c_i associados às derivadas D_i . Uma boa opção é utilizar um polinômio W_o que pertença ao espaço polinomial $\mathcal{P}_{s'}^2$, onde $s' \geq s = 2$. Por exemplo, considere os polinômios $W_o^2 \in \mathcal{P}_2^2$ e $W_o^3 \in \mathcal{P}_3^2$. No primeiro caso, ao utilizar W_o^2 , teremos de (1.37) que a ordem do erro de truncamento das aproximações das derivadas de primeira e segunda ordens são $O(h^2)$ e $O(h)$ respectivamente (quando $e_i^{(n+l)} \neq 0$), enquanto ao utilizar W_o^3 , a ordem do erro de truncamento das derivadas de primeira e segunda ordens serão de $O(h^3)$ e $O(h^2)$, respectivamente. Portanto as aproximações das derivadas utilizando o polinômio de aproximação W_o^3 são muito melhores que as obtidas pelo polinômio W_o^2 . A ordem de complexidade do MDFG é $O(n^3 n_v)$, onde n_v é o número total de nós do conjunto V , e n é a dimensão do espaço polinomial \mathcal{P}_s^d . Tal complexidade é fácil de ser demonstrada pois devemos solucionar para cada nó de V , o sistema linear dado por (1.8), para o qual pode ser utilizado um método como o LU que tem ordem $O(n^3)$. Assim, as complexidades do método ao utilizar os polinômios de aproximação W_o^2 (com $n = 5$) e W_o^3 (com $n = 9$) em cada um dos nós de V são $O(125n_v)$ e $O(729n_v)$ respectivamente; ou seja, o método terá um custo computacional quase seis vezes maior ao utilizar como polinômio de aproximação W_o^3 que ao utilizar W_o^2 . Portanto, a escolha do espaço dos polinômios de aproximação são sujeitas aos critérios de aproximação da solução vs custo computacional. Ao longo deste trabalho, utilizaremos como polinômios de aproximação, os pertencentes ao espaço \mathcal{P}_2^2 .

2.2 Malhas não-estruturadas.

Na implementação do MDFG, o domínio do problema deve ser discretizado por um conjunto de nós espalhados pelo domínio. Utilizamos malhas triangulares como suporte deste conjunto de nós, os quais têm a vantagem de que as relações entre um nó e seus vizinhos são imediatas, quando uma estrutura de dados apropriada é utilizada .

As malhas são classificadas como estruturadas e não-estruturadas. Quando uma malha estruturada é utilizada, o método clássico de diferenças finitas é recomendado para ser utilizado (por exemplo, diferenças centradas), devido ao fato de obter imediatamente as diferenças finitas. Geralmente, as malhas estruturadas estão formadas por um reticulado representando um domínio com geometria simples. Para domínios com geometria complexa, onde a utilização de malhas estruturadas é muito complexa, a utilização de malhas não-estruturadas se faz mais conveniente.

Uma das malhas não estruturadas mais utilizadas que é dada por elementos triangulares é a triangulação de Delaunay (ver apêndice I). Uma vantagem deste tipo de malha é que existem algoritmos eficientes para construí-la, como por exemplo, o algoritmo incremental. Estas malhas triangulares podem ser refinadas utilizando algoritmos iterativos tais como o algoritmo de Chew e o algoritmo de Rupert (uma descrição destes algoritmos de refinamento é apresentada no apêndice I). Nas malhas geradas pelo algoritmo de Chew, o tamanho dos elementos é praticamente igual, tendendo a triângulos equiláteros, diferentes das malhas geradas pelo algoritmo de Rupert, onde a gama dos tamanhos e variedade de formas dos elementos triangulares é muito amplo. Estas diferenças entre as malhas feitas por estes dois algoritmos podem ser apreciadas na figura 2.1.

Cada um destes algoritmos tem vantagens e desvantagens. Malhas geradas pelo algoritmo de Chew tem como vantagem elementos de excelente qualidade geométrica, embora, em grande quantidade, elevando portanto o custo de memória e o tempo computacional. As malhas geradas pelo algoritmo de Rupert têm a vantagem de possuir menos elementos, com variação de tamanhos em regiões distintas do domínio, tal como pode observar-se na figura (2.7). As malhas geradas pelo algoritmo de Rupert apresentam a desvantagem de possuir elementos triangulares de diferentes formas, o que pode gerar erros maiores na utilização de métodos numéricos, precisando portanto, de uma maior atenção na sua manipulação.

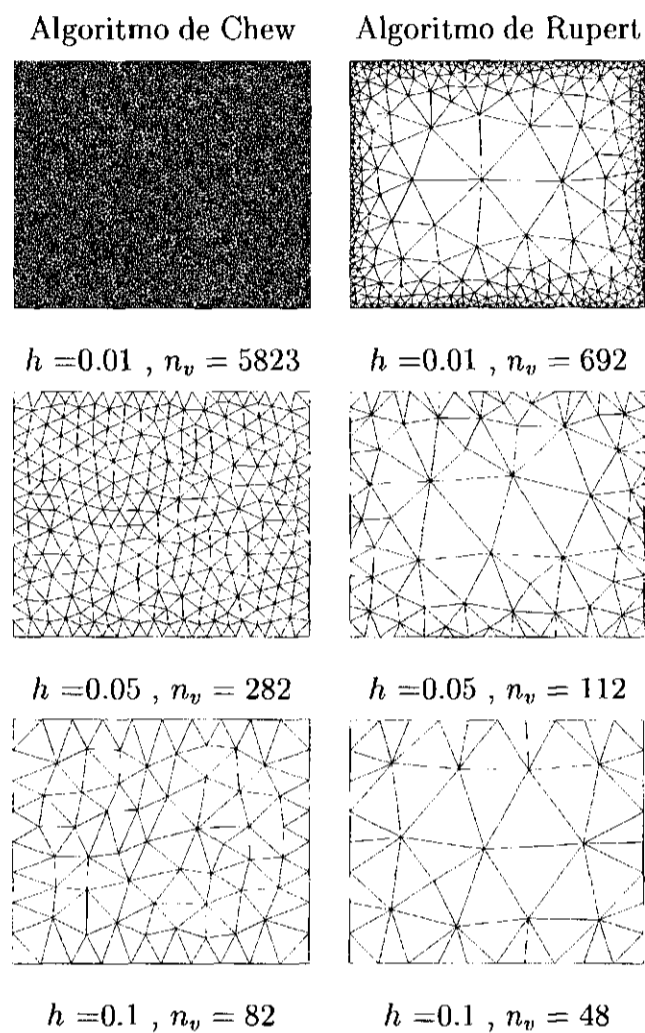


Figura 2.1: Malhas triangulares num retângulo com diferentes tamanhos de aresta mínima h , obtidos pelos métodos de refinamento de Chew e Rupert.

Ao ser utilizado um ajuste por mínimos quadrados no MDFG (tal como foi estudado na seção 1.2, solucionando o sistema linear (1.8) de n equações e n incógnitas), é necessário que todas as células C_i possuam um mínimo de n nós. Utilizando uma malha como suporte dos nós (os quais estão fixos nos vértices da malha) precisamos conhecer a quantidade de nós pertencentes aos distintos tipos de célula C(I), C(II), C(III) e C(IV), para poder criar um critério de escolha das melhores células para cada um dos nós. Iniciaremos estudando a dependência da quantidade de nós das células C(I), utilizando seis malhas com três parâmetros de comprimento h diferentes, geradas pelos algoritmos de refinamento de Chew e Rupert, sob um mesmo domínio retangular. Estas malhas com $h = 0.01, 0.05$ e 0.1 podem ser vistas na figura 2.1. Nas tabelas 2.2 encontra-se a porcentagem de nós que

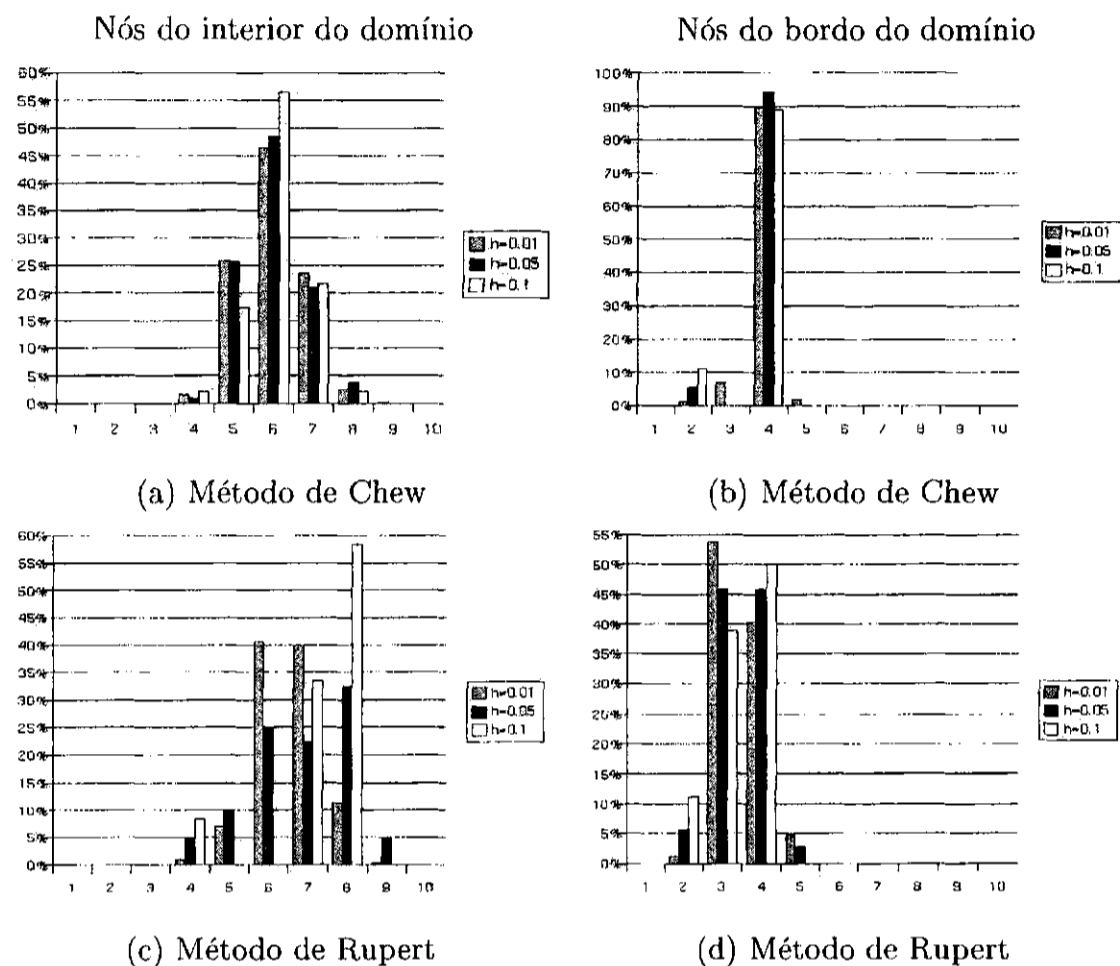


Figura 2.2: Distribuição do número de vértices das células $C(1)$ dos vértices das figuras anteriores.

possuem as células dos nós do interior e do bordo do domínio. Destas tabelas podemos concluir que quase todas as células $C(I)$ do interior do domínio possuem entre 5, 6, 7 ou 8 nós nas malhas geradas por ambos algoritmos; portanto, a princípio podemos implementar as células $C(I)$ na maioria dos nós do interior, e as células $C(II)$ no restantes das células com quantidade de nós inferiores a 5 nós. Além disso, pode ser observado que a maioria das células cujos nós estão sob o bordo do domínio, possuem 4 nós para malhas geradas pelo algoritmo de Chew, e entre 3 e 4 nós para malhas geradas pelo algoritmo de Rupert; então, deverão ser implementadas células tipo $C(II)$ nos nós cujas células $C(I)$ possuem 4 nós, e células $C(III)$ nos nós com $C(I)$ possuindo 3 ou 2 nós. Portanto, ao utilizar o algoritmo de Chew, os nós do bordo terão em sua maioria células $C(II)$ possuindo 6 ou 7 nós. Nestas discretizações do domínio, os nós que estão nos cantos do retângulo

são os únicos que podem contribuir com células C(I) de 2 nós, nos quais tem que ser implementadas células C(III) com 5 nós. É aconselhável utilizar sempre células C(III) nos nós sob o bordo quando a malha foi gerada pelo algoritmo de Rupert.

Dos resultados da seção seguinte podemos concluir que não é aconselhável utilizar células C(I) que possuam somente 5 nós, assim, é preferível fazer a implementação de células tipo C(II) onde 5 nós são encontrados.

2.3 Erro de truncamento do MDFG.

Um dos pontos principais na implementação de um método numérico é a redução dos erros. O principal erro que o MDFG introduz é o erro de truncamento (ET), ao efetuar a discretização das EDPs. As aproximações das derivadas obtidas pelo MDFG para qualquer nó $v_o \in V$ são dadas a partir dos resultados da seção 1.3 como:

$$D_i f \approx c_i \pi(i)$$

onde os termos c_i são expressos a partir da equação (1.25) como

$$c_i = \frac{1}{\pi(i)} [D_i f] + \bar{T}_{i,n}$$

onde $D_i f$ expressa o valor exato da i -ésima derivada de uma função f avaliada em $v_o = v_o(\vec{r}_o)$, n é a dimensão do espaço dos polinômios de aproximação ($i \leq n$), e o termo $\bar{T}_{i,n}$ é o erro truncamento do método, expressado a partir da proposição 1.3 como:

$$\bar{T}_{i,n} = \sum_{l=1}^{\infty} \frac{h^{N(n+l)-N(i)}}{\pi(n+l)} e_i^{(n+l)} [D_{n+l} f]$$

Da expressão acima podemos concluir que o ET depende diretamente das derivadas de ordem superior $D_{n+l} f$ da função f avaliada em v_o e de termos relacionados com a geometria da célula C_o .

Na implementação do MDFG deve-se ter muito cuidado ao discretizar regiões do domínio onde a função solução f apresenta altos valores em suas derivadas de ordem

superior, para controlar o ET. Portanto, neste tipo de funções, o ET deve ser controlado minimizando os termos relacionados com a geometria das células. Uma das opções para solucionar este problema consiste em minimizar o parâmetro global da malha h , inserindo mais nós no domínio (o qual equivale a utilizar uma malha mais fina como suporte dos nós); o que implica um aumento do custo computacional do método. Outra opção é tentar minimizar os termos $e_i^{(n+l)}$ fazendo-se refinamentos locais, deixando fixo o parâmetro h . Por este motivo, devemos estudar melhor o comportamento dos termos $e_i^{(n+l)}$ a partir de fatores tais como o tamanho relativo das células, quantidade de nós que ela possui irregularidades nas posições dos nós e tipo de célula.

2.3.1 Dependência do ET com a forma da célula.

A relação existente entre o ET do MDFG com a forma geométrica das células, será analisado em diferentes tipos de células (regulares e irregulares) de um mesmo tamanho, ou seja, com o raio de cada uma delas é igual a h . Para fazer esta análise, precisamos ter uma estimativa da magnitude dos ET introduzidos nas aproximações das derivadas calculadas pelo MDFG.

No problema bi-dimensional, utilizando polinômios de aproximação do espaço \mathcal{P}_2 , os termos $\bar{T}_{i,5}$ correspondentes aos ET das aproximações das derivadas $D_1f = \partial f/\partial x$ e $D_2f = \partial f/\partial y$, são dadas por:

$$\bar{T}_{i,5} = h^2 \left(\frac{e_i^{(6)}}{\pi(6)} [D_6f] + \frac{e_i^{(7)}}{\pi(7)} [D_7f] + \frac{e_i^{(8)}}{\pi(8)} [D_8f] + \frac{e_i^{(9)}}{\pi(9)} [D_9f] \right) + O(h^3) \quad (2.2)$$

para $i = 1, 2$; e para as aproximações das derivadas $D_3f = \partial^2 f/\partial x^2$, $D_4f = \partial^2 f/\partial x\partial y$ e $D_5f = \partial^2 f/\partial y^2$

$$\begin{aligned} \bar{T}_{i,5} = & h \left(\frac{e_i^{(6)}}{\pi(6)} [D_6f] + \frac{e_i^{(7)}}{\pi(7)} [D_7f] + \frac{e_i^{(8)}}{\pi(8)} [D_8f] + \frac{e_i^{(9)}}{\pi(9)} [D_9f] \right) + \\ & h^2 \left(\frac{e_i^{(10)}}{\pi(10)} [D_{10}f] + \frac{e_i^{(11)}}{\pi(11)} [D_{11}f] + \frac{e_i^{(12)}}{\pi(12)} [D_{12}f] + \frac{e_i^{(13)}}{\pi(13)} [D_{13}f] + \frac{e_i^{(14)}}{\pi(14)} [D_{14}f] \right) + \end{aligned} \quad (2.3)$$

para $i = 3, 4, 5$.

Devido que $\lim_{s \rightarrow \infty} 1/\pi(s) = 0$, e supondo que $h \ll 1$, podem ser desprezados os termos de ordem $O(h^3)$. Definindo $\tilde{D}_i^r f$ como:

$$\tilde{D}_i^r f = \max_{j \in \sigma_{i,r}} \{|D_j f|\}$$

onde o conjunto $\sigma_{i,r} = \{j \mid N_2(j) - N_2(i) = r\}$.

Então, a magnitude dos ET das derivadas de primeira e segunda ordem calculadas pelo MDFG, cumpriram as seguintes relações

$$|\bar{T}_{i,5}| \leq h^2 [\tilde{D}_i^2 f] \left| e_i^{(6)} + e_i^{(7)} + e_i^{(8)} + e_i^{(9)} \right|, \quad \text{para } i = 1, 2 \quad (2.4)$$

e

$$\begin{aligned} |\bar{T}_{i,5}| \leq & h [\tilde{D}_i^1 f] \left| e_i^{(6)} + e_i^{(7)} + e_i^{(8)} + e_i^{(9)} \right| + \\ & h^2 [\tilde{D}_i^2 f] \left| e_i^{(10)} + e_i^{(11)} + e_i^{(12)} + e_i^{(13)} + e_i^{(14)} \right|, \quad \text{para } i = 3, 4, 5 \end{aligned} \quad (2.5)$$

Finalmente, definindo o termo E_i^j dado como $\sum_{j \in \sigma_r} |e_i^{(j)}|$, obtemos uma estimativa do ET nas aproximações das derivadas de primeira ordem:

$$|\bar{T}_{i,n+l}| \leq h^2 [\tilde{D}_i^2 f] E_i^2, \quad \text{para } i = 1, 2 \quad (2.6)$$

e para as aproximações das derivadas de segunda ordem:

$$|\bar{T}_{i,n+l}| \leq h [\tilde{D}_i^1 f] E_i^1 + h^2 [\tilde{D}_i^2 f] E_i^2, \quad \text{para } i = 3, 4, 5. \quad (2.7)$$

Erro em células C(I).

Para estudar a dependência do ET do MDFG com respeito à forma geométrica das células, é feita uma análise estatística através das médias dos termos E_i^1 e E_i^2 para $i = 1, \dots, 5$ (utilizando uma função peso uniforme $\omega = 1$), para uma série de 200 células irregulares com um correspondente δs (isto feito para cada $\delta s = 0.0, 0.1, \dots, 1.0$), cuja escolha é aleatória. Os valores estimados dos termos E_i^1 e E_i^2 são representados pelas curvas apresentadas na figura 2.3 para células possuindo 5, 6, 7 e 8 nós.

Quando as células são “quase” regulares (ou seja, para células com $\delta s \approx 0$), os termos E_3^1 , E_4^1 e E_5^1 são muito pequenos, e podem em alguns casos ser desprezados (ver figuras 2.3c, 2.3e e 2.3g), exceto nas células com 5 nós (ver figura 2.3a). Portanto, quando a malha é formada por células quase regulares (cada célula com 6 ou mais nós), as aproximações das derivadas de segunda ordem têm um ET aproximada de ordem quadrática $O(h^2)$.

Pode-se observar que os valores dos termos E_i^1 e E_i^2 são muito semelhantes para todas as células, exceto nas células de 5 nós, as quais, apresentam uma aumento drástico em seus valores quando tem perturbações maiores que $\delta s \approx 0.6$ são empregadas. Podemos concluir, que a utilização de células tipo C(I) devem ser aplicadas somente quando elas possuem 6 ou mais nós, apresentando a vantagem de ter os ET de ordem quadráticas em células quase regulares. Quando as células forem irregulares, os valores dos termos E_i^1 e E_i^2 tendem a ser em média menores que 6.

Nas curvas das figura 2.4 são comparadas os valores dos termos E_4^1 e E_4^2 correspondentes ao ET das aproximação c_4 (que é a aproximação que tem erro mais alto, portanto teremos uma estimativa dos erros das outras aproximações) nos diferentes tipos de célula C(I), C(II), e C(III), possuindo 5, 10 e 15 nós respectivamente. Deste resultado concluímos, que ao utilizar células C(II) ou C(III) no nós cuja célula C(I) possui só 5 nós, os termos E_i^1 e E_i^2 mantêm valores baixos ($\lesssim 5$), mesmo com grau de perturbação $\delta s > 0.6$. Portanto numa boa implemetação, inicialmente todos os nós do interior do domínio terão células tipo C(I) como foi estudado na seção 2.2, e posteriormente, todos os nós com células que possuam menos de 6 nós são trocadas por células tipo C(II), e se ainda a célula não completa o mínimo de nós, é trocada por uma célula tipo C(III).

Erro em celulas C(II) e C(III).

Como foi estudado anteriormente na seção anterior e na seção 2.2, as células C(II) e C(III) são implemetadas nos nós que estão sobre o bordo ∂D e alguns outros do interior do domínio D . Precisamos portanto analisar como o ET do MDFG se comportar sob os nós que se encontram sobre o bordo ∂D . Nas figuras 2.5a e 2.5b são apresentadas dois exemplos de células C(II) e C(III) de nós sobre o bordo $\partial(D)$.

Por exemplo estudemos o comportamento dos ET das aproximações das derivadas $D_3 = \partial^2 f / \partial x^2$ e $D_5 = \partial^2 f / \partial y^2$ utilizando as células C(II) e C(III). Suponhamos como

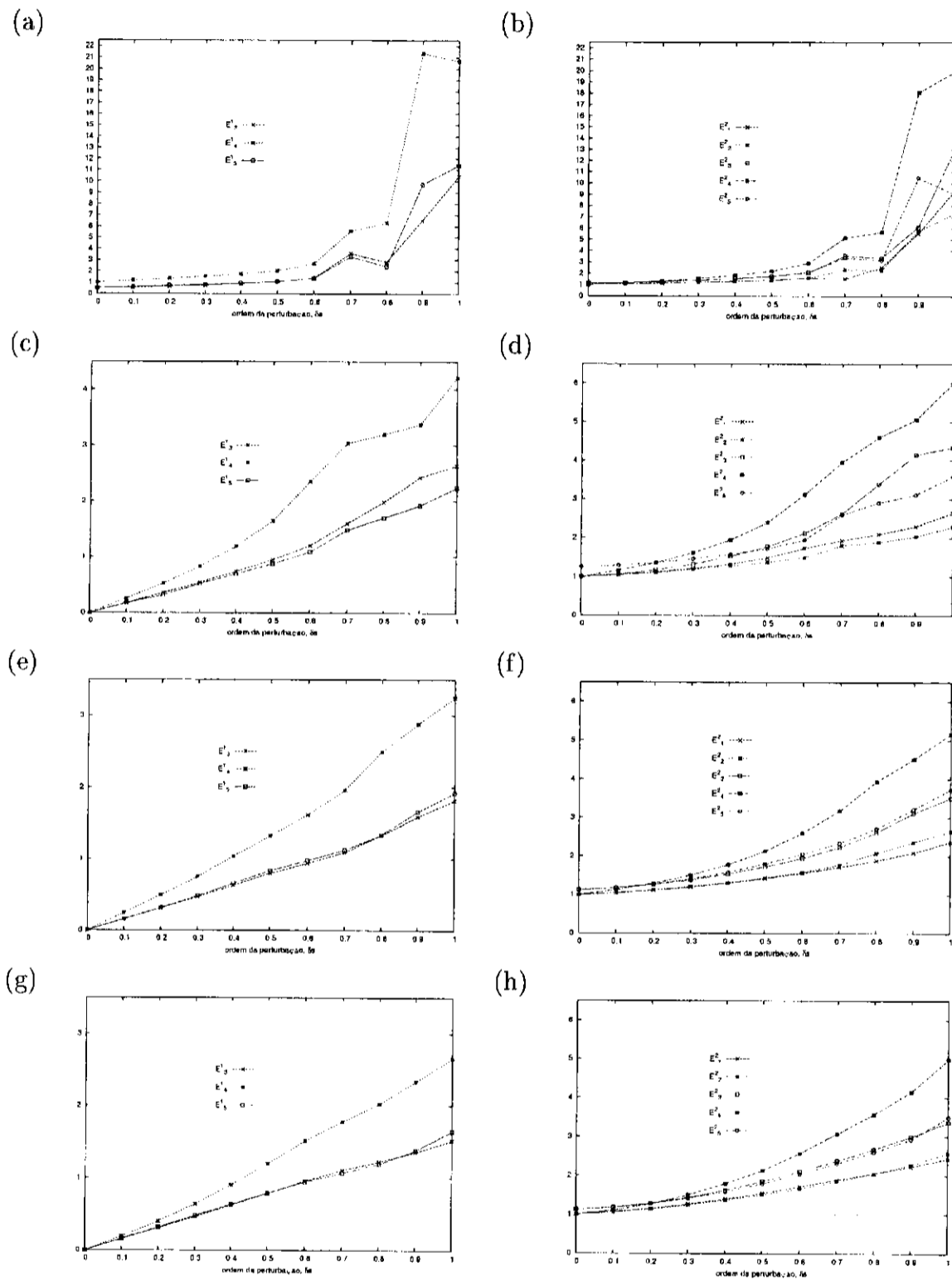


Figura 2.3: Valores das estimativas dos termos E_i^1 e E_i^2 em células de 5 nós (a,b), 6 nós (c,d), 7 nós (e,f), e 8 nós (g,h), respeito à ordem de perturbação δs .

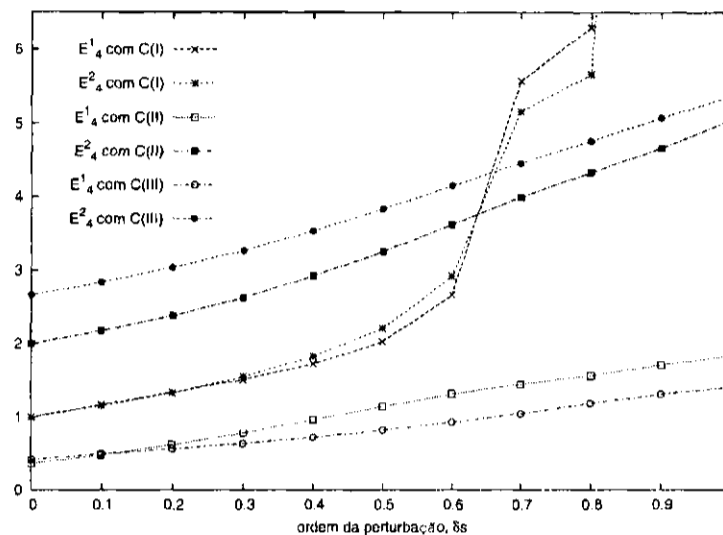


Figura 2.4: Valores das estimativas dos termos E_4^1 e E_4^2 em células tipo C(I), C(II), e C(III), possuindo 5, 10 e 15 nós respectivamente.

suporte dos nós, uma malha cujos elementos são triângulos equiláteros quando $\delta s = 0$ como nas figuras 2.5; onde a linha do bordo ∂D está alinhado com o eixo x . Nas figuras 2.6a e 2.6b são apresentadas os termos E_3^1 e E_3^2 correspondente ao ET da aproximação das derivadas D_3 , e os termos E_5^1 e E_5^2 correspondente ao ET da aproximação do D_5 respectivamente, quando os diferentes nós são perturbados.

Com estes resultados podemos ver que valores do ET para ambas células não diferem muito, sendo o ET para as aproximações de D_5 ligeiramente maior que os ET de D_3 . Portanto, para ter um menor custo computacional, são sempre implementadas células C(II) nos nós do bordo ∂D , a exceção dos nós que estão nos cantos do domínio. Para os nós que estão nos cantos, é aconselhável utilizar células C(III) na sua implementação.

2.3.2 ET relacionado ao tamanho relativo das células.

Os tamanhos das células de diferentes nós de V (tamanho medido pelo raio da célula) podem geralmente variar em diferentes regiões do domínio, especialmente quando são utilizadas malhas não estruturadas como suporte dos nós; embora, em algumas malhas estruturadas como da figura 1.6b este problema pode aparecer. Em algumas malhas não estruturadas o tamanho dos elementos (e portanto das células) pode variar apreciavelmente, como por exemplo a malha gerada pelo algoritmo de Rupert da figura 2.7.

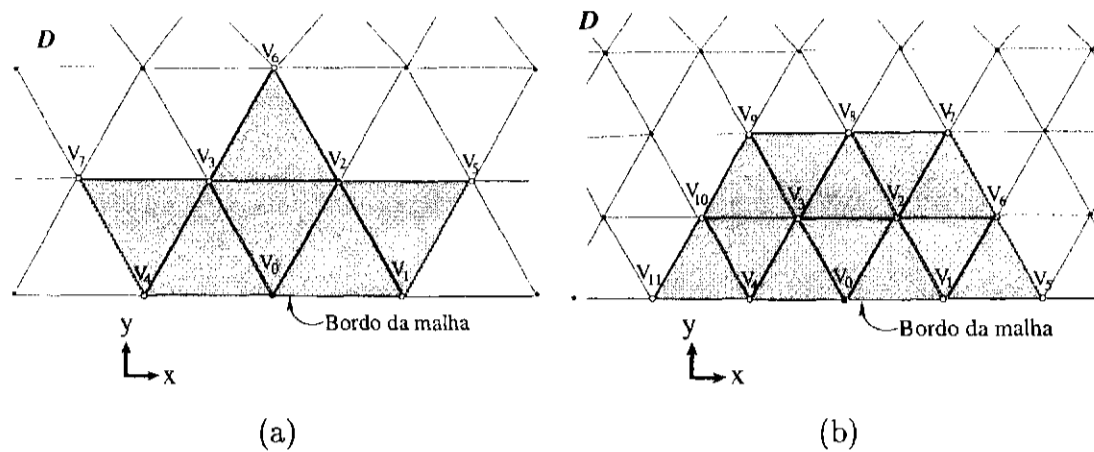


Figura 2.5: Células tipo C(I) (a) e do tipo C(III) (b), para um nó no bordo ∂D do domínio D .

Para examinar a relação existente entre o tamanho de células de diferentes nós com o ET, é introduzido o termo \bar{h}_i , chamado "tamanho relativo da célula C_i ", o qual está definido como $\bar{h}_i = \rho_i/h$, quando h é fixo. Por exemplo, as duas células C_1 e C_2 da figura 2.7 tem um tamanho relativo $\bar{h}_1 = 1.1$ e $\bar{h}_2 = 11.3$ respectivamente, ou seja, a célula C_2 tem um tamanho quase 11 vezes maior que a célula C_1 .

Chamando como $\bar{e}_i^{(n+l)}$ aos termos do ET dependentes da forma geométrica da célula, quando ela pode ter um tamanho relativo maior que 1; e deve cumprir com

$$\lim_{h \rightarrow 1} \bar{e}_i^{(n+l)} = e_i^{(n+l)}$$

Nos seguintes exemplos são calculados os termos $\bar{e}_i^{(n+l)}$ em diferentes células:

Exemplo 2.1. *Seja uma distribuição de nós que possuam somente células pentagonais e hexagonais regulares de diferentes tamanhos. Sejam as células C_1 (pentagonal, com 5 nós) e C_2 (hexagonal, com 6 nós) com seus respectivos tamanhos relativos \bar{h}_1 e \bar{h}_2 , tal como são dadas na figura 2.8. Os termos $e_i^{(n+l)}$ obtidos ao utilizar um polinômio de aproximação \mathcal{P}_2^2 para estas duas células são dados nas tabelas 2.1 e 2.2.*

Exemplo 2.2. *Suponha-se uma malha composta por células hexagonais irregulares C(I). Por exemplo, seja a célula C_3 do nó $v_3 \in V$, mostrada na figura 2.9, a qual tem uma ordem*

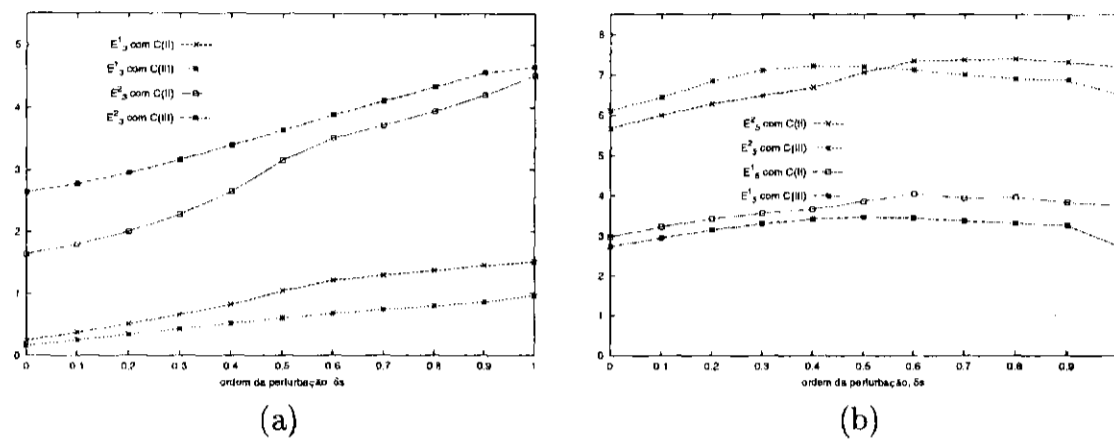


Figura 2.6: Valores das estimativas dos termos E_3^1 , E_3^2 associados à aproximação do $\partial^2 f / \partial x^2$ (a), e dos termos E_5^1 , E_5^2 associados à aproximação do $\partial^2 f / \partial y^2$ (b), para células C(II) e C(III) de um nó $v_o \in \partial D$

i	$\bar{e}_i^{(6)}$	$\bar{e}_i^{(7)}$	$\bar{e}_i^{(8)}$	$\bar{e}_i^{(9)}$	$\bar{e}_i^{(10)}$	$\bar{e}_i^{(11)}$	$\bar{e}_i^{(12)}$	$\bar{e}_i^{(13)}$	$\bar{e}_i^{(14)}$
1	$0.75\bar{h}_1^{-2}$	0	$0.25\bar{h}_1^{-2}$	0	$0.125\bar{h}_1^{-3}$	0	$-0.125\bar{h}_1^{-3}$	0	$0.125\bar{h}_1^{-3}$
2	0	$0.25\bar{h}_1^{-2}$	0	$0.75\bar{h}_1^{-2}$	0	$-0.125\bar{h}_1^{-3}$	0	$0.125\bar{h}_1^{-3}$	0
3	$0.25\bar{h}_1$	0	$-0.25\bar{h}_1$	0	$0.875\bar{h}_1^{-2}$	0	$0.125\bar{h}_1^{-2}$	0	$-0.125\bar{h}_1^{-2}$
4	0	$-0.5\bar{h}_1$	0	$0.5\bar{h}_1$	0	$0.5\bar{h}_1^{-2}$	0	$0.5\bar{h}_1^{-2}$	0
5	$-0.25\bar{h}_1$	0	$0.25\bar{h}_1$	0	$-0.125\bar{h}_1^{-2}$	0	$0.125\bar{h}_1^{-2}$	0	$0.875\bar{h}_1^{-2}$

Tabela 2.1: Elementos $e_i^{(n+l)}$ da célula pentagonal regular C_1 obtidos a partir de um polinômio de aproximação \mathcal{P}_2^2 .

i	$\bar{e}_i^{(6)}$	$\bar{e}_i^{(7)}$	$\bar{e}_i^{(8)}$	$\bar{e}_i^{(9)}$	$\bar{e}_i^{(10)}$	$\bar{e}_i^{(11)}$	$\bar{e}_i^{(12)}$	$\bar{e}_i^{(13)}$	$\bar{e}_i^{(14)}$
1	$0.75\bar{h}_2^{-2}$	0	$0.25\bar{h}_2^{-2}$	0	0	0	0	0	0
2	0	$0.25\bar{h}_2^{-2}$	0	$0.75\bar{h}_2^{-2}$	0	0	0	0	0
3	0	0	0	0	\bar{h}_2^{-2}	0	0	0	0
4	0	0	0	0	0	$0.25\bar{h}_2^{-2}$	0	$0.75\bar{h}_2^{-2}$	0
5	0	0	0	0	$-0.25\bar{h}_2^{-2}$	0	$0.25\bar{h}_2^{-2}$	0	$0.75\bar{h}_2^{-2}$

Tabela 2.2: Elementos $e_i^{(n+l)}$ da célula hexagonal regular C_2 obtidos a partir de um polinômio de aproximação \mathcal{P}_2^2 .

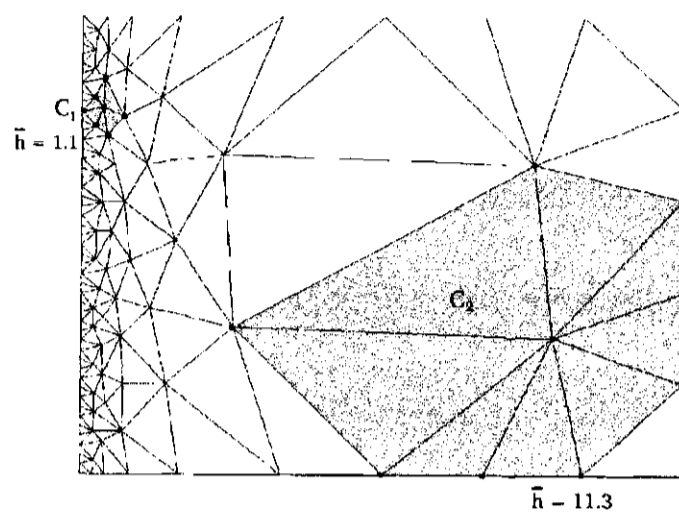


Figura 2.7: Malha com duas células C_1 e C_2 tipo $C(I)$ de diferentes forma e tamanho.

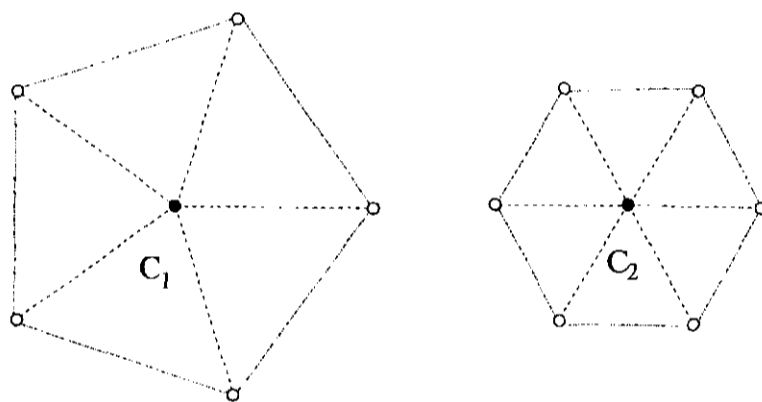
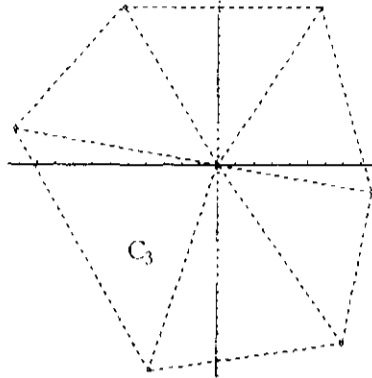


Figura 2.8: Células $C(I)$ regulares: C_1 pentagonal e C_2 hexagonal.

de perturbação $\delta s = 0.2$. Os elementos $e_i^{(n+1)}$, gerados pela célula C_3 são apresentadas na tabela 2.3.

2.4 ET devido a função peso.

Na seção 1.2 foi introduzido a função peso ω , que é uma função que depende das posições dos nós de uma célula. Nos resultados obtidos anteriormente, foi utilizada uma função peso uniforme, dada por $\omega = 1$, para todos os nós da célula. O objetivo desta seção é observar como é afectado o ET do MDFG para uma função peso que privilegie os nós de

Figura 2.9: Célula C(I) hexagonal irregular C_3 do exemplo 2.2.

i	$\bar{e}_i^{(6)}$	$\bar{e}_i^{(7)}$	$\bar{e}_i^{(8)}$	$\bar{e}_i^{(9)}$	$\bar{e}_i^{(10)}$	$\bar{e}_i^{(11)}$	$\bar{e}_i^{(12)}$	$\bar{e}_i^{(13)}$	$\bar{e}_i^{(14)}$
1	$0.756\bar{h}_3^{-2}$	$-0.11\bar{h}_3^{-2}$	$0.258\bar{h}_3^{-2}$	$0.088\bar{h}_3^{-2}$	$-0.206\bar{h}_3^{-3}$	$0.029\bar{h}_3^{-3}$	$0.047\bar{h}_3^{-3}$	$-0.049\bar{h}_3^{-3}$	$-0.042\bar{h}_3^{-3}$
2	$-0.004\bar{h}_3^{-2}$	$0.274\bar{h}_3^{-2}$	$0.828\bar{h}_3^{-2}$	$0.68\bar{h}_3^{-2}$	$-0.082\bar{h}_3^{-3}$	$0.057\bar{h}_3^{-3}$	$0.023\bar{h}_3^{-3}$	$-0.038\bar{h}_3^{-3}$	$-0.134\bar{h}_3^{-3}$
3	$-0.32\bar{h}_3$	$0.073\bar{h}_3$	$0.01\bar{h}_3$	$-0.014\bar{h}_3$	$1.064\bar{h}_3^{-2}$	$-0.112\bar{h}_3^{-2}$	$0.005\bar{h}_3^{-2}$	$0.094\bar{h}_3^{-2}$	$-0.09\bar{h}_3^{-2}$
4	$-0.224\bar{h}_3$	$0.212\bar{h}_3$	$-0.026\bar{h}_3$	$-0.114\bar{h}_3$	$0.132\bar{h}_3^{-2}$	$0.307\bar{h}_3^{-2}$	$-0.11\bar{h}_3^{-2}$	$0.671\bar{h}_3^{-2}$	$0.09\bar{h}_3^{-2}$
5	$0.2\bar{h}_3$	$-0.005\bar{h}_3$	$-0.034\bar{h}_3$	$-0.188\bar{h}_3$	$-0.307\bar{h}_3^{-2}$	$0.007\bar{h}_3^{-2}$	$0.269\bar{h}_3^{-2}$	$-0.008\bar{h}_3^{-2}$	$0.722\bar{h}_3^{-2}$

Tabela 2.3: Elementos $\bar{e}_i^{(n+l)}$ da célula hexagonal irregular C_3 obtidos a partir de um polinômio de aproximação \mathcal{P}_2^2 .

uma célula que estão mais próximos ao nó principal.

Introduzimos uma nova função peso, expressa como:

$$\omega = \frac{1}{(\rho_{o,k})^4}$$

Por exemplo, é calculado os termos E_3^2 e E_5^2 do ET das aproximações das derivadas D_3 e D_5 , utilizando uma célula C(II), em um nó do interior do domínio. Os valores dos termos E_3^2 e E_5^2 dependentes das perturbações δs , são apresentados nas curvas da figura 2.10.

Pode-se concluir, que quando é utilizada uma função peso que privilegia os nós mais próximos do nó principal, o ET das aproximações se reduzem, em $\sim 30\%$ a $\sim 40\%$ do valor do ET obtido quando é utilizado uma função peso que não privilegia nenhum dos nós, tal como a função peso uniforme.

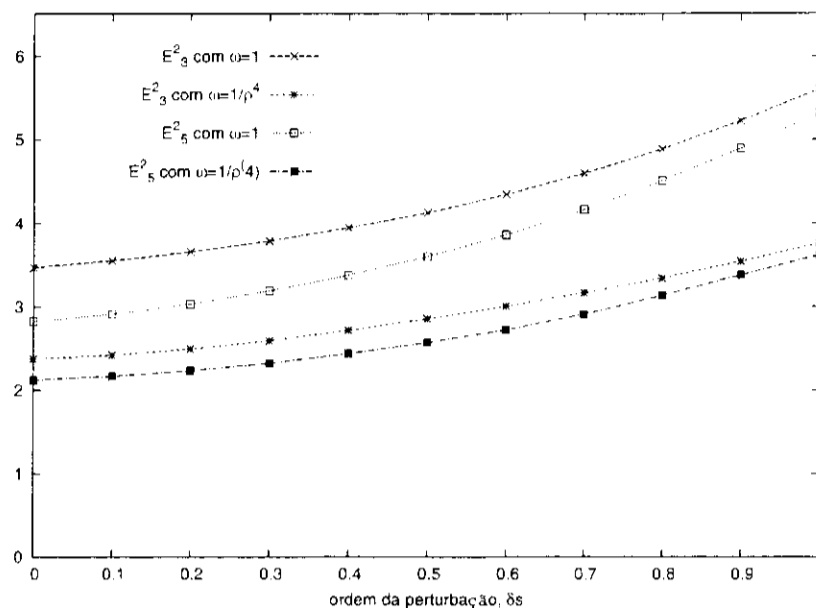


Figura 2.10: Termos de E_3^2 e E_5^2 para duas diferentes funções peso ω .

2.5 Teste Numérico

Para finalizar este capítulo é feito um experimento numérico para comprovar as propriedades do MDFG introduzidas nas seções anteriores. Para isto, é calculado utilizando o MDFG, as derivadas de uma função $g(x, y)$ cujas derivadas são conhecidas.

É utilizado como erro da solução numérica a seguinte expressão:

$$E = \sqrt{\frac{1}{n_v} \sum_{j=1}^{n_v} |F^{(real)}(\vec{r}_j) - F^{(num)}(\vec{r}_j)|^2} \quad (2.8)$$

onde n_v é a quantidade de nós utilizados na discretização do domínio, $F^{(real)}(\vec{r}_j)$ e $F^{(num)}(\vec{r}_j)$ expressam os valores da solução real e numérica do problema em \vec{r}_j . Para nosso problema, a função real e numérica correspondem à derivada D_i da função $g(x, y)$, e aos valores numéricos (calculados pelo MDFG), desta derivada. Neste caso, chamaremos $E = E_{D_i}$.

Utilizando a seguinte função teste:

$$g(x, y) = c \exp(a((x - 0.2)^2 + (y - 0.2)^2)) \quad (2.9)$$

definida no domínio quadrado $D = [0, 1] \times [0, 1]$.

São calculados os erros E_{D_i} para $i = 1, \dots, 5$, utilizando discretizações do domínio D , para diferentes valores de h . Estas discretizações são feitas a partir de malhas criadas pelo algoritmo de Chew, o qual nos garante células quase regulares. Os erros E_{D_1} , E_{D_3} , e E_{D_5} da função teste são apresentados na figura 2.11, para $c = 1$ e $a = 0.25$.

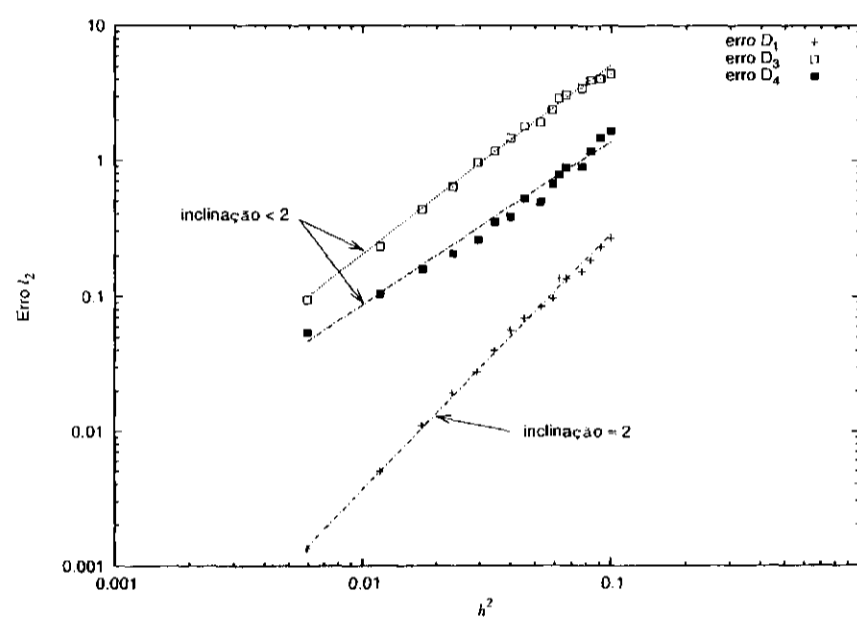


Figura 2.11: Valores do erro para as aproximações de D_1 , D_3 , e D_5 da função (2.9), para $c = 1$ e $a = 0.25$

Observando a inclinações das curvas (escala logarítmica), se comprova o carácter linear dos erros das aproximações de D_3 e D_4 , e quadrático para a aproximação de D_1 , tal como foi visto em (2.6), utilizando um polinômio de aproximação \mathcal{P}_2^2 .

Capítulo 3

Aplicações do MDFG.

Neste capítulo utilizaremos o MDFG na solução de diferentes tipos de EDP. Serão estudadas as aplicações do MDFG na solução de alguns modelos de problemas elípticos, problemas parabólicos, e finalmente na solução de problemas mistos como as equações de Navier-Stokes. Em cada um destes casos serão apresentados exemplos, onde se utilizou a estrutura de dados *singular handle edge (SHE)*, para manipular as malhas utilizadas na discretização do domínio (os detalhes do funcionamento da estrutura de dados SHE são apresentados no apêndice B).

3.1 MDFG para problemas elípticos.

Uma das mais comuns classes de equações diferenciais parciais são as equações parciais elípticas. As EDP elípticas são fundamentalmente diferentes das EDP hiperbólicas e parabólicas, e os esquemas numéricos para a aproximação de soluções de problemas elípticos são essencialmente diferentes dos esquemas de solução de problemas parabólicos e hiperbólicos.

O modelo mais comum de equação diferencial parcial linear elípticas é a equação de Poisson. Para um problema bi-dimensional ela é dada como:

$$-\nabla^2 f(\mathbf{r}) = F(\mathbf{r}), \quad \mathbf{r} \in D \quad (3.1)$$

Problemas que requerem a solução da equação de Poisson (ou equação de Laplace, quando $F(\mathbf{r}) = 0$), estão sujeitas a diferentes tipos de condições de fronteira. No problema

de Dirichlet, os valores da solução da equação de Poisson nas fronteiras estão condicionadas a valores prescritos,

$$f = g(\mathbf{r}) , \quad \mathbf{r} \in \partial D \quad (3.2)$$

No problema de Neumann, a derivada normal de f é especificada sob ∂D , ou seja,

$$\frac{\partial f}{\partial n} = h(\mathbf{r}) , \quad \mathbf{r} \in \partial D \quad (3.3)$$

3.1.1 Representação da equação de Poisson pelo MDFG.

A partir da equação (3.1), onde f e F são definidos em subconjuntos de \mathbb{R}^2 num problema bi-dimensional, e o operador ∇^2 é expresso como:

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$$

Os “métodos” de solução de uma equação de Poisson utilizando as diferenças finitas clássicas e o MDFG, requerem solucionar um sistema linear de equações algébricas.

Utilizando as aproximações c_3 e c_5 para as derivadas de segunda ordem de f em um nó v_i (obtidas pelo MDFG a partir da solução do sistema linear (1.8)), a equação (3.1) será discretizada como:

$$2 c_3 + 2 c_5 = F(\mathbf{r}_i) , \quad v_i = v_i(\mathbf{r}_i) \in V \quad (3.4)$$

onde V é o conjunto de nós que estão discretizando o domínio D .

Cada um dos termos c_3 e c_5 pode ser expresso como uma relação linear dos valores de f no nó v_i e nos nós $v_k \in C_i$,

$$\begin{aligned} c_3 &= \bar{w}_{i,i}^3 f(\mathbf{r}_i) + \sum_{v_k \in C_i} \bar{w}_{i,k}^3 f(\mathbf{r}_k), \\ c_5 &= \bar{w}_{i,i}^5 f(\mathbf{r}_i) + \sum_{v_k \in C_i} \bar{w}_{i,k}^5 f(\mathbf{r}_k). \end{aligned} \quad (3.5)$$

Então, o operador $\nabla^2 f(\mathbf{r}_i)$, aplicado no nó v_i , pode ser escrito por:

$$\begin{aligned}\nabla^2 f(\mathbf{r}_i) &= 2 (\bar{w}_{i,i}^3 + \bar{w}_{i,i}^5) f(\mathbf{r}_i) + \sum_{v_k \in C_i} 2 (\bar{w}_{i,k}^3 + \bar{w}_{i,k}^5) f(\mathbf{r}_k) \\ &= w_{i,i} f(\mathbf{r}_i) + \sum_{v_k \in C_i} w_{i,k} f(\mathbf{r}_k)\end{aligned}\quad (3.6)$$

onde $w_{i,j} = 2 (\bar{w}_{i,j}^3 + \bar{w}_{i,j}^5)$ para $j = i, k$.

Proposição 3.1. *Os termos $w_{i,i}$ e $a_{i,k}$ associados a v_i e $\forall v_k \in C_i$, cumprem a seguinte desigualdade:*

$$|w_{i,i}| \leq \sum_{v_k \in C_i} |w_{i,k}| \quad (3.7)$$

Demonstração. Para qualquer nó $v_i \in V$, os componentes c_3 e c_5 são obtidos do sistema linear (1.8), cuja matriz \mathbf{A} pode ser expressa como $\mathbf{A} = \mathbf{H}\tilde{\mathbf{A}}\mathbf{H}$, então o vetor \mathbf{c} é dado por:

$$\mathbf{c} = \mathbf{H}^{-1}\tilde{\mathbf{A}}^{-1}\mathbf{H}^{-1}\mathbf{b}$$

onde a matriz $\tilde{\mathbf{A}}^{-1}$ é composta pelos elementos matriciais $\tilde{u}_{j,k}$, e \mathbf{b} é um vetor definido em (1.10), que pode ser escrito como o produto $\mathbf{H}\mathbf{b}'$, onde o j -ésimo componente de \mathbf{b}' é

$$b'_j = \sum_{v_k \in C_i} (f(\mathbf{r}_k) - f(\mathbf{r}_i)) u_{j,k} \sqrt{\omega_k}$$

então, teremos que $\mathbf{c} = \mathbf{H}^{-1}\tilde{\mathbf{A}}^{-1}\mathbf{b}'$. Calculando os termos c_3 e c_5 se obtém:

$$c_3 = \sum_{v_k \in C_i} f(\mathbf{r}_k) \left(\frac{1}{h^2} \sum_{j=i}^n \tilde{u}_{3,j} u_{j,k} \sqrt{\omega_k} \right) - f(\mathbf{r}_i) \sum_{v_k \in C_i} \left(\frac{1}{h^2} \sum_{j=i}^n \tilde{u}_{3,j} u_{j,k} \sqrt{\omega_k} \right)$$

e

$$c_5 = \sum_{v_k \in C_i} f(\mathbf{r}_k) \left(\frac{1}{h^2} \sum_{j=i}^n \tilde{u}_{5,j} u_{j,k} \sqrt{\omega_k} \right) - f(\mathbf{r}_i) \sum_{v_k \in C_i} \left(\frac{1}{h^2} \sum_{j=i}^n \tilde{u}_{5,j} u_{j,k} \sqrt{\omega_k} \right)$$

assim, o operador ∇^2 será escrito como:

$$\nabla^2 = 2 c_3 + 2 c_5 = \sum_{v_k \in C_i} f(\mathbf{r}_k) \left(\frac{2}{h^2} \sum_{j=i}^n (\tilde{u}_{3,j} + \tilde{u}_{5,j}) u_{j,k} \sqrt{\omega_k} \right) - f(\mathbf{r}_i) \sum_{v_k \in C_i} \left(\frac{2}{h^2} \sum_{j=i}^n (\tilde{u}_{3,j} + \tilde{u}_{5,j}) u_{j,k} \sqrt{\omega_k} \right)$$

Portanto, de (3.6) obtemos:

$$w_{i,i} = - \sum_{v_k \in C_i} \left(\frac{2}{h^2} \sum_{j=i}^n (\tilde{u}_{3,j} + \tilde{u}_{5,j}) u_{j,k} \sqrt{\omega_k} \right) \quad (3.8)$$

$$w_{i,k} = \frac{2}{h^2} \sum_{j=i}^n (\tilde{u}_{3,j} + \tilde{u}_{5,j}) u_{j,k} \sqrt{\omega_k} \quad (3.9)$$

ou seja:

$$w_{i,i} = - \sum_{v_k \in C_i} w_{i,k}$$

aplicando o valor absoluto se obtém imediatamente a relação (3.7). \square

Exemplo 3.1. Os termos $w_{i,j}$ obtidos por uma célula regular $C(I)$ de 6 nós, utilizando um polinômio de aproximação P_2^2 , são dados por: $w_{i,k} = 2/3$, para $\forall v_k \in C_i$, e $w_{i,i} = -4$, para o nó v_i .

Sucessivamente, se calcula para cada um dos nós de V , obtendo-se um problema simultâneo de n_v equações algébricas com n_v desconhecidos valores de $f(v_i)$. Matematicamente, nosso problema (3.1) pode ser expresso como:

$$\begin{aligned} m_{1,1} f(\mathbf{r}_1) + m_{1,2} f(\mathbf{r}_2) + \cdots + m_{1,n_v} f(\mathbf{r}_{n_v}) &= F(\mathbf{r}_1) \\ m_{2,1} f(\mathbf{r}_1) + m_{2,2} f(\mathbf{r}_2) + \cdots + m_{2,n_v} f(\mathbf{r}_{n_v}) &= F(\mathbf{r}_2) \\ &\vdots \\ m_{n,1} f(\mathbf{r}_1) + m_{n,2} f(\mathbf{r}_2) + \cdots + m_{n,n_v} f(\mathbf{r}_{n_v}) &= F(\mathbf{r}_n) \end{aligned} \quad (3.10)$$

onde $m_{i,i} = -w_{i,i}$, $m_{i,k} = -w_{i,k}$ para $v_k \in C_i$, e os demais elementos matriciais serão nulos. Este sistema linear pode ser escrito compactamente como $\mathbf{M}\mathbf{f} = \mathbf{F}$.

No caso de um problema com condições de fronteira de Dirichlet, temos que:

$$f(\mathbf{r}_j) = g(\mathbf{r}_j), \quad v_j \in \partial D \quad (3.11)$$

portanto, podem ser eliminados de (3.10) as equações algébricas relacionadas com $v_j \in \partial D$, ficando um novo sistema linear dado como

$$\mathbf{M}'\mathbf{f}' = \mathbf{y} \quad (3.12)$$

onde \mathbf{f}' é um vetor com os valores desconhecidos da função f sobre os nós do interior do domínio D , ou seja $\mathbf{f}' = \{f(\mathbf{r}_l) \mid \forall v_l \in (D - \partial D)\}$; e \mathbf{y} é um vetor dado por:

$$y_l = \begin{cases} F(\mathbf{r}_l), & \text{se } v_l \text{ não é adjacente a } \partial D, \\ F(\mathbf{r}_l) + \sum_{v_k \in (C_l \cap \partial D)} w_{l,k} g(\mathbf{r}_k), & \text{caso contrário.} \end{cases} \quad (3.13)$$

Definição 3.1. A matriz $\mathbf{G} \in \mathbb{R}^n \times \mathbb{R}^n$ é monotónica se e só se

$$\mathbf{G}u \leq \mathbf{G}v \quad \Rightarrow \quad u \leq v \quad \forall u, v \in \mathbb{R}^n$$

denotando “ \leq ” como:

$$u \leq v \quad \Leftrightarrow \quad u^i \leq v^i, \quad i = 1, 2, \dots, n.$$

Definição 3.2. A matriz $\mathbf{G} = (g_{i,j})$ de um sistema linear

$$\sum_{j=1}^n g_{i,j} x_j = f_i, \quad i = 1, 2, \dots, n,$$

é irredutível se e só se não existe nenhuma transformação linear

$$x'_i = \sum_{j=1}^n \beta_{i,j} x_j,$$

tal que o sistema sistema de equações equivalente

$$\sum_{j=1}^n g'_{i,j} x'_j = f'_i, \quad i = 1, 2, \dots, n,$$

possa ser substituído por dois sistemas independentes

$$\begin{aligned} \sum_{j=1}^k g'_{i,j} x'_j &= f'_i, \quad i = 1, 2, \dots, k, \\ \sum_{j=k+1}^n g'_{i,j} x'_j &= f'_i, \quad i = k+1, \dots, n, \end{aligned}$$

O seguinte teorema é dado para nossa exposição:

Teorema 3.1. Se cada um dos termos $w_{i,k}$ são não negativos e a matriz \mathbf{M}' é irredutível, então \mathbf{M}' é monotónica.

Demonstração. As seguintes condições tem que ser validadas [27] :

$$\begin{aligned}
 \text{(i)} \quad & m'_{i,j} \leq 0 \quad \forall i, j = 1, \dots, N, \quad i \neq j, \\
 \text{(ii)} \quad & m'_{i,i} > 0 \quad \forall i = 1, \dots, N, \\
 \text{(iii)} \quad & \sum_j m'_{i,j} \geq 0 \quad \forall i = 1, \dots, N, \\
 \text{(iv)} \quad & \exists i \quad \sum_i m'_{i,j} > 0.
 \end{aligned} \tag{3.14}$$

□

No seguinte teorema é formulada uma estimacão da convergência do MDFG aplicada a problemas elípticos:

Teorema 3.2. *Se a matriz \mathbf{M}' é monotónica, então o MDFG converge linearmente, isto é:*

$$\max_{v_i \in V} |f(\mathbf{r}_i) - \bar{f}(\mathbf{r}_i)| \leq Ch \tag{3.15}$$

onde C é uma constante positiva independente do parâmetro de comprimento da malha h e \bar{f} denota o valor calculado da solução f .

Demonstração. A prova deste teorema é feita no trabalho de Demkowicz [10]. □

Portanto, para solucionar o problema (3.1) pelo MDFG, todas as células tem que cumprir as condições (3.14) para garantir a convergência.

3.1.2 Exemplo numérico.

Para testar o que foi visto anteriormente, utilizamos uma equação de Poisson cuja solução é conhecida, e posteriormente é estudada a convergência do método. Utilizamos para isto a seguinte equação:

$$\nabla^2 f(x, y) = 6xy(x^2 + y^2 - 2), \quad (x, y) \in D = [-1, 1] \times [-1, 1]$$

com condição de Dirichlet na fronteira:

$$f(x, y) = 0, \quad (x, y) \in \partial D$$

cuja solução analítica é:

$$f(x, y) = xy(x^2 - 1)(y^2 - 1)$$

Para solucionar o sistema linear (3.12) é utilizado o método iterativo Gauss-Seidel. Com uma condição de suficiência para garantimos que o método Gauss-Seidel é convergente, exigimos que o sistema seja irredutível e que os elementos matriciais $m'_{i,j}$ da matriz \mathbf{M}' cumpram com:

$$|m'_{i,i}| \geq \sum_{j=1, j \neq i}^n |m'_{i,j}| \quad (3.16)$$

para todos os i , e que para pelo menos um i ,

$$|m'_{i,i}| > \sum_{j=1, j \neq i}^n |m'_{i,j}| \quad (3.17)$$

Então, se pudermos garantir que para todos os nós $v_i \in V$, os termos $a_{i,k}$ dados em (3.9), são não negativos, ou seja, ocorre a igualdade na equação (3.7), a relação (3.16) é cumprida, e para qualquer i -ésima linha associada à solução num nó v_i adjacente à fronteira ∂D , cumprirá imediatamente (3.17).

São utilizadas discretizações do domínio criadas a partir de malhas geradas pelo algoritmo de Chew, com diferentes valores de h ; garantindo que as células cumpram com a não negatividade dos termos $a_{i,k}$. Na figura 3.1 é apresentado o erro, o qual é calculado utilizando (2.8). A convergência neste exemplo concorda com o resultado teóricos dado no teorema 3.1. Na figura 3.2 é apresentada a solução analítica do problema e a solução calculada pelo MDFG correspondente a $h = 0.117648$.

3.2 MDFG para problemas parabólicos.

Na física alguns problemas tais como de difusão, a função f que representa, dependendo do caso, a temperatura, concentração de uma substância, etc; é expressa pelas variáveis espaciais \mathbf{r} (por exemplo num problema tri-dimensional podem ser utilizadas as variáveis cartesianas x , y e z) e a variável temporal t . Esta função f é submetida a uma equação diferencial da forma:

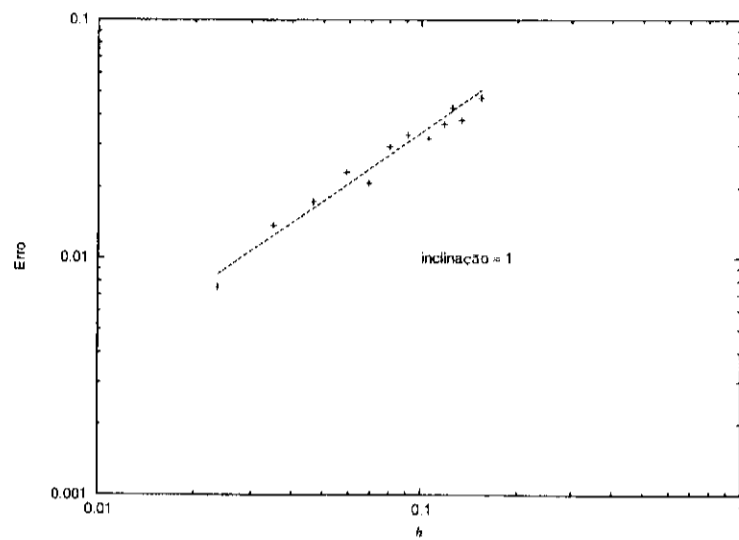
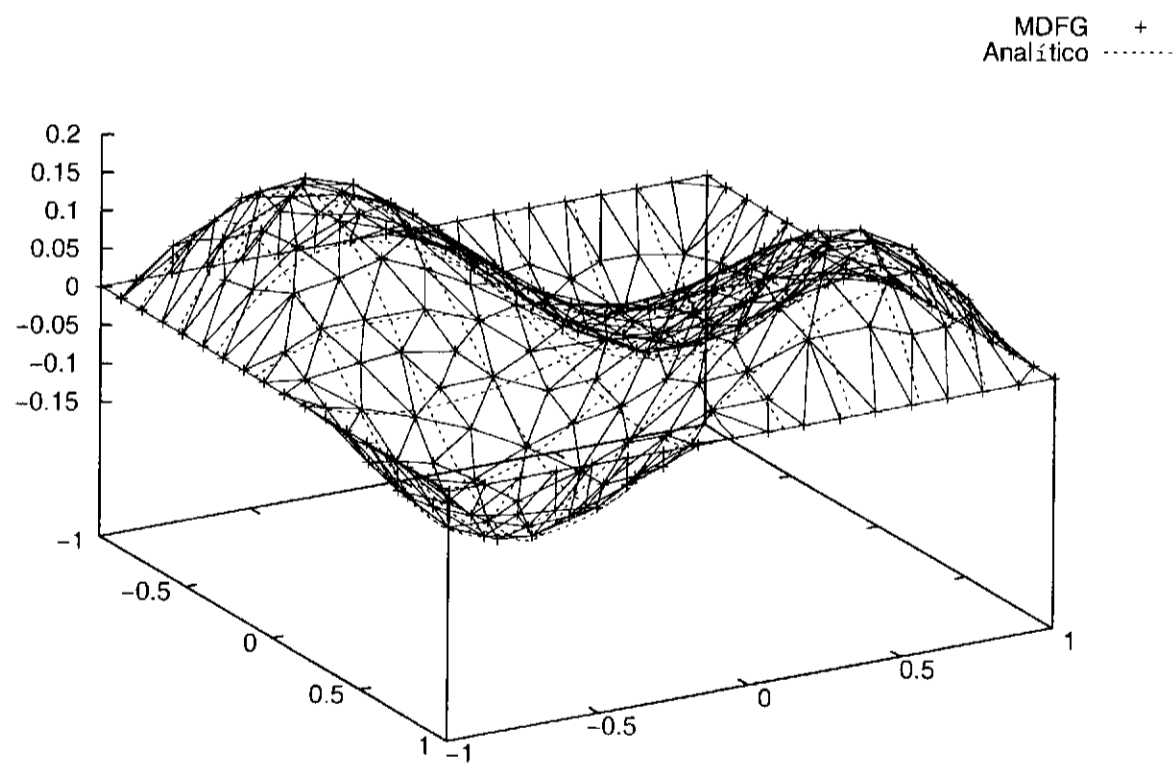


Figura 3.1: Resultado do teste de convergência do MDFG num problema elíptico.

Figura 3.2: Solução analítica e calculada pelo MDFG ($h = 0.117648$)

$$\frac{\partial f(\mathbf{r})}{\partial t} = \nu \nabla^2 f(\mathbf{r}) \quad (3.18)$$

cuja condição inicial é dada como

$$f(\mathbf{r}, t)|_{t=0} = f(\mathbf{r}, 0) = f_0(\mathbf{r}) \quad (3.19)$$

e suas condições de fronteira

$$f(\mathbf{r}, t) = g(\mathbf{r}), \quad \mathbf{r} \in \partial D \quad (3.20)$$

para um problema de Dirichlet; e num problema com condições de fronteira de Neumann:

$$\frac{\partial f(\mathbf{r}, t)}{\partial n} = h(\mathbf{r}), \quad \mathbf{r} \in \partial D \quad (3.21)$$

onde \mathbf{n} é o vetor normal unitário à fronteira ∂D . No decorrer desta seção será estudada a equação do calor por ser este um exemplo característico das equações parabólicas.

3.2.1 Solução da equação de calor pelo MDFG.

Num problema bi-dimensional, a equação do calor expressa a partir de (3.18) é dada por:

$$\frac{\partial T}{\partial t} = \kappa \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right) \quad (3.22)$$

onde $T(x, y, t)$ representa um campo escalar de temperaturas num determinado tempo t , e κ é a difusividade térmica (neste exemplo a difusividade térmica é considerada constante em qualquer lugar do domínio D).

A partir de um esquema explícito, utilizamos as diferenças progressivas para representar a derivada temporal e as aproximações calculadas pelo MDFG para as diferenças espaciais em um nó v_i , calculadas a partir de um polinômio de aproximação do espaço \mathcal{P}_2^2 . Representamos discretamente a equação (3.22) como:

$$\frac{T_i^{n+1} - T_i^n}{\Delta t} = \kappa(2 c_3 + 2 c_5) \quad (3.23)$$

onde as aproximações c_3 e c_5 são calculadas no tempo n .

Como:

$$\begin{aligned} \frac{T_i^{n+1} - T_i^n}{\Delta t} &= \left. \frac{\partial T_i}{\partial t} \right|_{t=n} + \frac{\Delta t}{2!} \left. \frac{\partial^2 T_i}{\partial t^2} \right|_{t=n} + \frac{(\Delta t)^2}{3!} \left. \frac{\partial^3 T_i}{\partial t^3} \right|_{t=n} + \dots \\ &= \left. \frac{\partial T_i}{\partial t} \right|_{t=n} + O(\Delta t) \end{aligned} \quad (3.24)$$

a partir das expressões (1.25), (1.26) e (1.37) calculadas na seção 1.3, temos que c_3 pode ser expresso como:

$$\begin{aligned} c_3 &= \frac{\partial^2 T_i}{\partial x^2} + \frac{h}{6} e_3^{(6)} \frac{\partial^3 T_i}{\partial x^3} + \frac{h}{2} e_3^{(7)} \frac{\partial^3 T_i}{\partial x^2 \partial y} + \frac{h}{2} e_3^{(8)} \frac{\partial^3 T_i}{\partial x \partial y^2} + \frac{h}{6} e_3^{(9)} \frac{\partial^3 T_i}{\partial y^3} + \dots \\ &= \frac{\partial^2 T_i}{\partial x^2} + O(h) \end{aligned} \quad (3.25)$$

e igualmente para c_5 :

$$\begin{aligned} c_5 &= \frac{\partial^2 T_i}{\partial y^2} + \frac{h}{6} e_5^{(6)} \frac{\partial^3 T_i}{\partial x^3} + \frac{h}{2} e_5^{(7)} \frac{\partial^3 T_i}{\partial x^2 \partial y} + \frac{h}{2} e_5^{(8)} \frac{\partial^3 T_i}{\partial x \partial y^2} + \frac{h}{6} e_5^{(9)} \frac{\partial^3 T_i}{\partial y^3} + \dots \\ &= \frac{\partial^2 T_i}{\partial y^2} + O(h) \end{aligned} \quad (3.26)$$

Substituindo (3.24), (3.25) e (3.26) em (3.23) se obtém:

$$\frac{\partial T}{\partial t} - \kappa \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right) = O(h, \Delta t) \quad (3.27)$$

portanto o esquema de aproximação é consistente, já que quando $h \rightarrow 0$ e $\Delta t \rightarrow 0$, a solução calculada tende à solução real do problema.

Por exemplo, se o domínio é discretizado utilizando uma malha regular cujas células $C(I)$ são hexagonais, a partir dos resultados da seção 1.4, sabemos que o MDFG aplicado na solução de um problema difusivo é condicionalmente estável (ver exemplo 1.10), quando se cumpre,

$$\kappa \frac{\Delta t}{h^2} \lesssim 0.32 \quad (3.28)$$

A partir destes resultados de consistência e estabilidade, demonstramos que o MDFG é convergente, utilizando o teorema de equivalência de Lax:

Teorema 3.3. *Dado um problema com valores iniciais, uma aproximação por diferenças finitas satisfaz as condições de consistência e estabilidade, então ela é convergente.*

3.2.2 Exemplo numérico.

Como exemplo numérico é solucionado o sistema (3.22), para $\kappa = 0.1$, num domínio quadrado $D = [-1, 1] \times [-1, 1]$, com condições de fronteira de Dirichlet, $T(\mathbf{r}, t) = 0$ para qualquer $t > 0$ e $\mathbf{r} \in \partial D$; utilizando como condição inicial a:

$$T(\mathbf{r}, 0) = \cos\left(x \frac{\pi}{2}\right) \cos\left(x \frac{\pi}{2}\right), \quad \mathbf{r} \in D$$

cuja solução analítica é

$$T(\mathbf{r}, t) = \exp\left(-\frac{\pi^2}{20} t\right) \cos\left(x \frac{\pi}{2}\right) \cos\left(x \frac{\pi}{2}\right), \quad \mathbf{r} \in D$$

Utilizando um passo temporal $\Delta t = 0.001$, e diferentes malhas não estruturadas (variando h em cada uma delas), tais que cada uma de suas células sejam estáveis, é calculado a solução para $t = 2.0$, calculando posteriormente o erro da solução numérica utilizando (2.8). Estes erros são apresentados na figura 3.3, mostrando que o MDFG utilizando como polinômios de aproximação do espaço \mathcal{P}_2^2 , possui convergência linear com respeito a h .

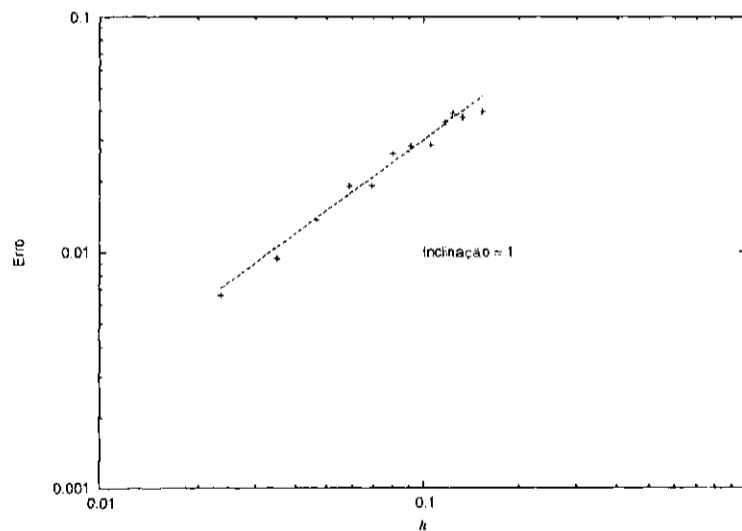


Figura 3.3: Resultado do teste de convergência do MDFG num problema parabólico.

3.3 MDFG para problemas de Navier-Stokes.

A partir dos esquemas apresentados nas seções anteriores, vamos construir um método numérico para a solução das equações de Navier-Stokes. Suponhamos um fluido Newto-

niano e incompressível, sem presença de forças externas. As equações de Navier-Stokes em duas dimensões na forma adimensional podem ser escritas como:

$$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{1}{Re} \frac{\partial^2 u_i}{\partial x_j \partial x_j} \quad (3.29)$$

$$\frac{\partial u_i}{\partial x_i} = 0 \quad (3.30)$$

onde p , Re e u_i (com $i = 1, 2$) definem a pressão, número de Reynolds e o campo de velocidades do fluido.

Implementa-se uma disposição co-localizada para as variáveis sobre a malha não estruturada, na qual os valores da velocidade e a pressão são armazenadas nos vértices da malha. O cálculo das variáveis para tempos posteriores são obtidos utilizando um esquema explícito para os termos não lineares e viscoso, seguido de uma correção implícita da pressão ao impôr a condição de incompressibilidade.

Fazendo a discretização com o método explícito de Euler para um tempo posterior $n + 1$, tem-se,

$$u_i^{n+1} = \Delta t \left(\frac{u_i^n}{\Delta t} + H_i^n - \frac{\delta p^n}{\delta x_i} \right) \quad (3.31)$$

onde H_i inclui os termos advectivo e viscoso,

$$H_i = \frac{1}{Re} \frac{\delta^2 u_i}{\delta x_j \delta x_j} - u_j \frac{\delta u_i}{\delta x_j} \quad (3.32)$$

O campo de velocidade no tempo $n + 1$ tem que satisfazer a equação de continuidade, assim:

$$\frac{\delta u_i^{n+1}}{\delta x_i} = 0 \quad (3.33)$$

Para o cumprimento de (3.33), toma-se a divergência numérica na equação (3.31)

$$\frac{\delta u_i^{n+1}}{\delta x_i} = \Delta t \left[\frac{\delta u_i^n}{\delta x_i} + \frac{\delta}{\delta x_i} \left(H_i^n - \frac{\delta p^n}{\delta x_i} \right) \right] \quad (3.34)$$

O primeiro termo é a divergência para o novo campo de velocidade, o qual desejamos que seja zero. O termo $\delta u_i^n / \delta x_i$ é zero se a continuidade for garantida no tempo n . Portanto, o termo da direita tem que ser zero, resultando:

$$\frac{\delta}{\delta x_i} \left(\frac{\delta p^n}{\delta x_i} \right) = \frac{\delta H_i^n}{\delta x_i} \quad (3.35)$$

o operador $\delta/\delta x_i$ fora do parêntese é o operador divergência inerente na equação de continuidade, quando $\delta p/\delta x_i$ é o gradiente da pressão da equação de momento. Com a pressão satisfazendo a equação (3.35), a velocidade no tempo $n + 1$ terá divergência nula.

Condições para a velocidade e pressão são aplicadas aos contornos do domínio. Assim, apresentam-se quatro casos particulares:

- **Parede fixa.** As componentes da velocidade são nulas:

$$u_{\hat{n}} = u_{\hat{t}} = 0 \quad , \quad (3.36)$$

onde \hat{n} e \hat{t} são os vetores normais e transversais ao bordo do domínio (à parede) respectivamente. Para a pressão, sua derivada na direção normal \hat{n} é igual a zero:

$$\frac{\partial p}{\partial \hat{n}} = 0 \quad . \quad (3.37)$$

- **Parede deslizando.** Para uma parede deslizando com uma velocidade U_0 , as componentes da velocidade são:

$$u_{\hat{n}} = 0 \quad , \quad u_{\hat{t}} = U_0 \quad (3.38)$$

A pressão no contorno apresenta o mesmo tipo de condição que no caso anterior, equação (3.37).

- **Entrada de fluxo.** No caso de uma região do bordo do domínio pela qual existe uma entrada de fluido com uma velocidade U_1 , as componentes da velocidades serão:

$$u_{\hat{n}} = U_1 \quad , \quad u_{\hat{t}} = 0 \quad (3.39)$$

A pressão cumprirá com a equação (3.37).

- **Saida de fluxo.** Para as regiões do bordo do domínio pela qual sai fluido livremente, as componentes da velocidade são dadas por:

$$\frac{\partial u_{\hat{n}}}{\partial \hat{n}} = 0 \quad , \quad u_i = 0 \quad , \quad (3.40)$$

e a pressão será nula,

$$p = 0 \quad . \quad (3.41)$$

Com a implementação do MDFG obtém-se de forma imediata o termo H a partir da equação (3.32), utilizando as aproximações geradas a partir dos campos escalares das componentes da velocidade $u_1(x, y)$ e $u_2(x, y)$. Igualmente procede-se para calcular a pressão p , utilizando as aproximações obtidas pelo MDFG do campo da pressão $p(x, y)$ e o campo de $H(x, y)$, ao resolver o sistema (3.35).

Assim, tem-se o seguinte algoritmo de resolução para as equações de Navier-Stokes:

1. Inicialize com um campo de velocidade u_i^n no tempo t_n o qual é assumido ter divergência nula, e cumpram as condições de contorno para a velocidade.
2. Calcula-se a componente H_i^n , dos termos advectivos e viscosos e suas divergências.
3. Resolve-se o sistema (3.35), obtendo a pressão p^n , garantindo as condições de contorno para a pressão.
4. Calcula-se o campo de velocidade para um novo tempo t_{n+1} . Ele deve ter divergência zero.
5. retorna-se ao passo 2, até alcançar um tempo final.

3.3.1 Problema da cavidade.

Inicialmente trabalhou-se com a simulação do escoamento dentro de uma cavidade quadrada com fluido, onde sua parede superior desliza com velocidade constante. Para a geração de malha se utilizou o algoritmo de Chew.

A figura (3.4) mostra a malha utilizada na simulação e o campo de velocidades para um número de Reynolds $Re = 10$. É importante notar que o resultado é muito similar às simulações descritas na literatura [15]

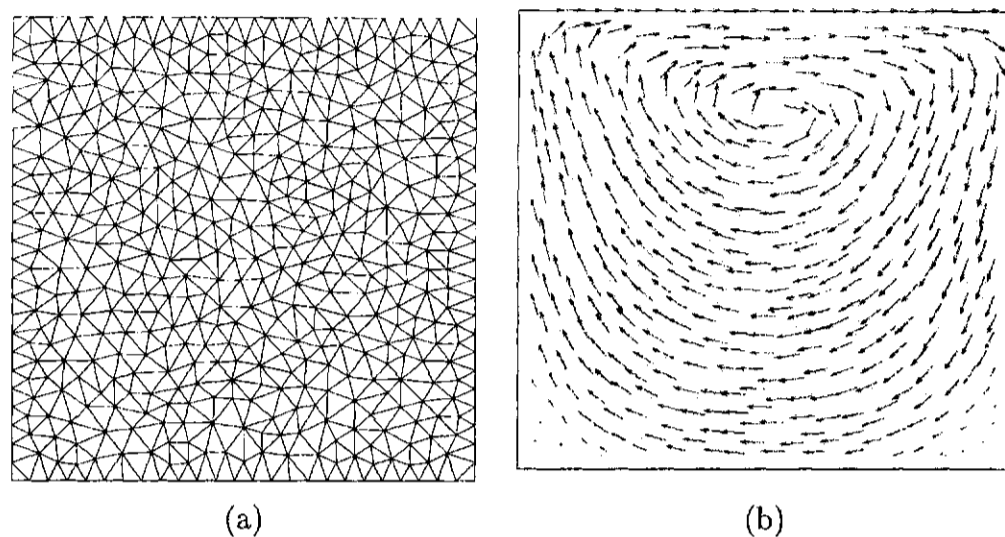


Figura 3.4: (a) malha não estruturada da cavidade (malha 2), (b) campo de velocidade no regime permanente.

Para este problema, a validade quantitativa verificar-se comparando os valores extremos das componentes da velocidade no estado estacionário. Como parâmetro de referência calculamos os valores extremos utilizando o método MDFG proposto na seção 1.2, comparando-os com valores de referência obtidos por o método de Diferenças Finitas com malhas estruturadas, utilizando duas malhas cartesianas finas ($\Delta x = 0.01$ e $\Delta x = 0.05$). Na tabela 3.3.1 são mostrados os valores obtidos usando-se três diferentes malhas não estruturadas: malha 1, com o comprimento da máxima aresta $h_{max} = 0.077$, malha 2 com $h_{max} = 0.040$, e a malha 3 com $h_{max} = 0.034$; com os valores de referência.

	Malha 1 $h_{max} = 0.077$	Malha 2 $h_{max} = 0.040$	Malha 3 $h_{max} = 0.034$	Malha Ref.1 $\Delta x = 0.01$	Malha Ref.2 $\Delta x = 0.05$
u_x max	1.000000	1.000000	1.000000	1.000000	1.000000
u_x min	-0.193458	-0.203028	-0.204136	-0.208160	-0.203127
u_y max	0.357710	0.355756	0.348211	0.363490	0.323435
u_y min	-0.312526	-0.345987	-0.356363	-0.395921	-0.358723

Tabela 3.1: Comparação das velocidades extremas da cavidade.

Os valores de u_x max são sempre iguais à velocidade da parede superior $u_x = 1$, assim não são revelantes para a comparação. As outras velocidades obtidas com as malhas 1, 2 e 3 são muito próximas aos valores das soluções de referência. Pode-se observar que os erros do MDFG são da mesma ordem que os erros no método de Diferenças Finitas sobre

uma malha uniforme cartesiana (o qual é um método de precisão de segunda ordem) com malhas de tamanho comparáveis.

3.3.2 Escoamento numa artéria.

O segundo exemplo apresenta a flexibilidade do método MDFG para geometrias complexas. Uma das aplicações na qual estamos interessados é o escoamento de fluidos em estruturas orgânicas. A modelagem de estruturas orgânicas são feitas através de técnicas de processamento de imagens em combinação com métodos de geração de malhas [25].



Figura 3.5: Imagem microscópica de uma artéria.

A figura 3.5 mostra a imagem microscópica de uma artéria, a partir da qual a malha da figura 3.6 foi gerada. O número de Reynolds do escoamento é $Re = 10$, baseado no diâmetro da artéria e na velocidade do escoamento entrando nela. A figura 3.7 mostra o campo vetorial de velocidade gerado pelo MDFG. Note que o método numérico cria uma simulação bastante séria do escoamento na geometria complexa da artéria.

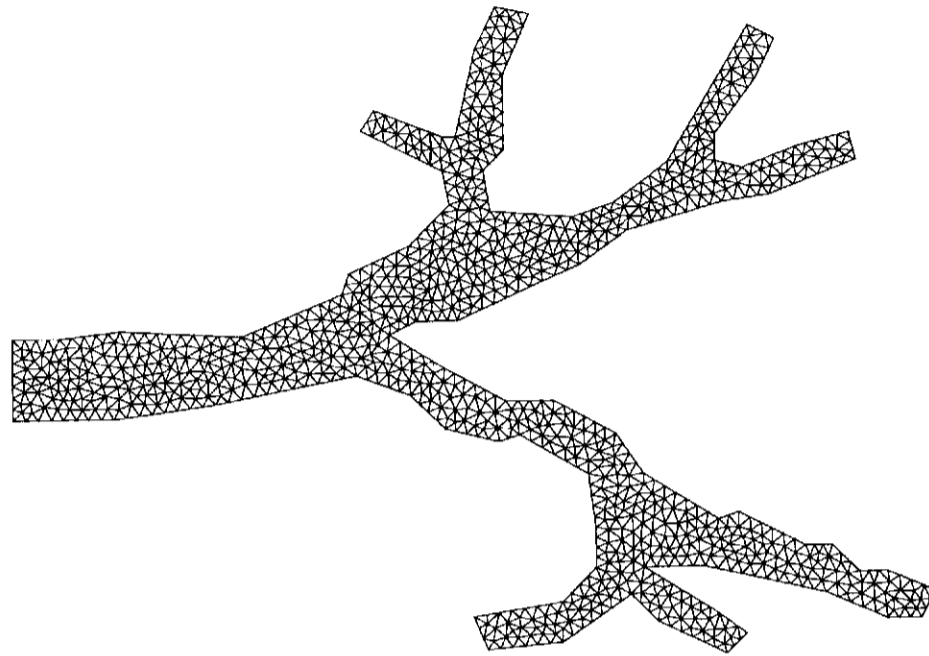


Figura 3.6: Malha da artéria

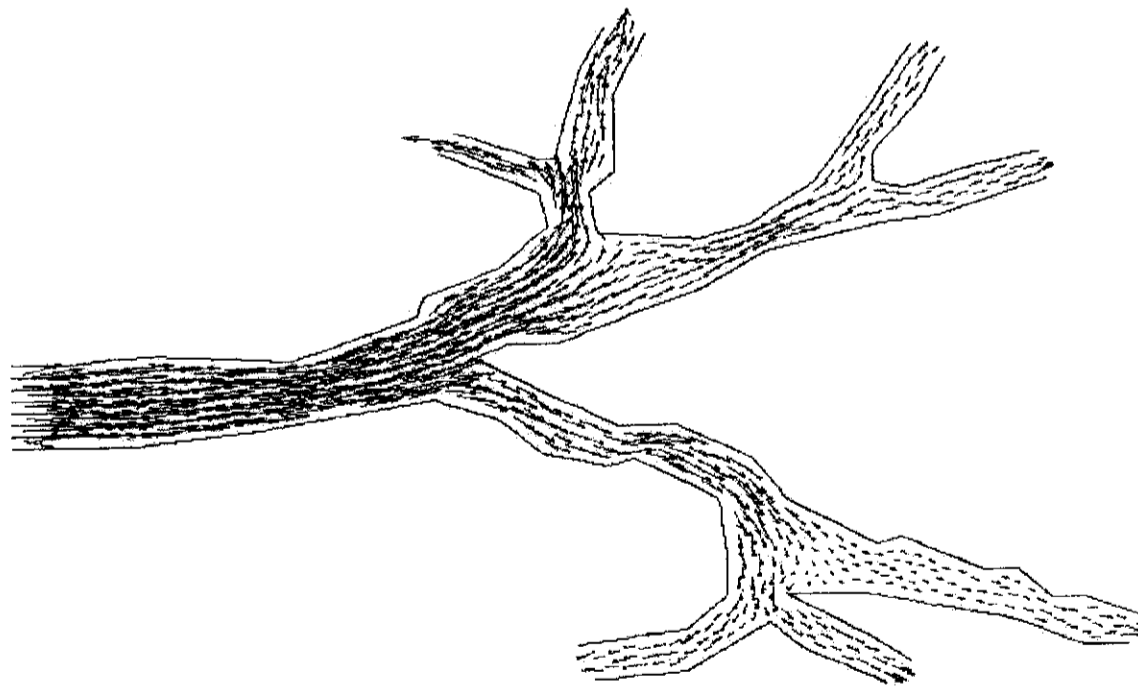


Figura 3.7: Campo de velocidade de um escoamento dentro da artéria.

Conclusões.

Neste trabalho foi estudado um Método de Diferenças Finitas Generalizadas (MDFG) o qual é clasificado dentre o grupo de métodos meshless, já que permite calcular as derivadas de uma função f numa coleção de nós espalhados sob um domínio. Os principais resultados obtidos com este trabalho são:

- O MDFG é rápido e fácil de implementar, mostrando ser muito robusto e flexível.
- É demonstrado que as aproximações das derivadas calculadas pelo MDFG são consistentes quando é utilizado um polinômio de aproximação completo (ou seja, é formado por todos os monômios base de grau igual ou menor ao grau do polinômio). Assim, quando é utilizado um polinômio não completo, as aproximações das derivadas podem ser inconsistentes.
- Um critério eficiente de seleção dos nós de uma células de um nó é proposto, baseado nas relações implícitas de conectividade que possui uma malha utilizada como suporte dos nós no domínio. Aplicando algoritmos eficientes de geração de malhas, pode-se garantir uma boa distribuição dos nós, permitindo reduzir os erros da aproximação.
- A geometria de células bi-dimensionais não afetam significativamente o ET, quando a célula possui mais de n nós (onde n é a quantidade de monômios base que compõem o polinômio de aproximação) e a distribuição desses nós têm baixa irregularidade.
- O MDFG aplicado na solução de problemas elípticos com condições de fronteira de Dirichlet são convergentes. Além disso, o erro do método é linear para problemas bi-dimensionais utilizando um polinômio de aproximação de segundo grau.

- A convergência do MDFG aplicado na solução de equações diferenciais parciais com dependência temporal é dada quando o esquema utilizado é estável e consistente (Teorema de Lax). É demonstrado que problemas advectivos (hiperbólicos) são instáveis (quando $\omega = 1$). Em problemas difusivos, o MDFG converge para células C(I) hexagonais regulares, uma vez são condicionalmente estáveis.

Para futuros trabalhos se propõe fazer a implementação do MDFG para ambientes tridimensionais; estudar mais a convergência do método com respeito às características geométricas das células, e para diferentes classes de função peso ω (isotrópicas e anisotrópicas), e utilizar esquemas deslocados para solucionar diferentes tipos problemas (por exemplo: escoamento de fluidos).

Apêndice A

Geração de Malhas Não-Estruturadas.

A.1 Triangulação de Delaunay.

Uma construção bem conhecida para geração de malhas triangulares é a Triangulação de Delaunay (*TD*). A triangulação de Delaunay é uma estrutura geométrica de grande importância na geração de malhas, já que ela simultaneamente otimiza parâmetros de qualidade dos elementos. Por exemplo, em R^2 , de todas as triangulações geradas a partir de um conjunto de pontos no plano, a *TD* é a única triangulação que maximiza o mínimo ângulo dos elementos da malha.

Seja $P = p_1, \dots, p_n$ um conjunto de pontos em posição geral em R^d . O simplexo definido por $(d + 1)$ pontos de P ¹ é um simplexo de Delaunay se a circunferência do simplexo não contém pontos de P em seu interior (figura A.1a). A união de todos os simplexos de Delaunay formam a Triangulação de Delaunay $DT(P)$. Na figura A.1b se mostra o diagrama de Voronoi, que é o dual geométrico da Triangulação de Delaunay. O diagrama de Voronoi consiste de um conjunto de políedros V_1, \dots, V_n associados a cada um dos pontos de P . V_i é chamada a célula de Voronoi de p_i , e p_i é chamado o centro de V_i . Geometricamente, V_i é formado pelo conjunto de pontos em R^d cuja distância a p_i é menor ou igual a distância a qualquer outro ponto de P (figura A.1b).

¹Um conjunto P de n pontos em R^d é dito estar em posição geral se nenhum subespaço afim de R^d contém P , e se não há uma esfera S^{d-1} passando através de $d + k$ pontos de P , com $k > 1$.

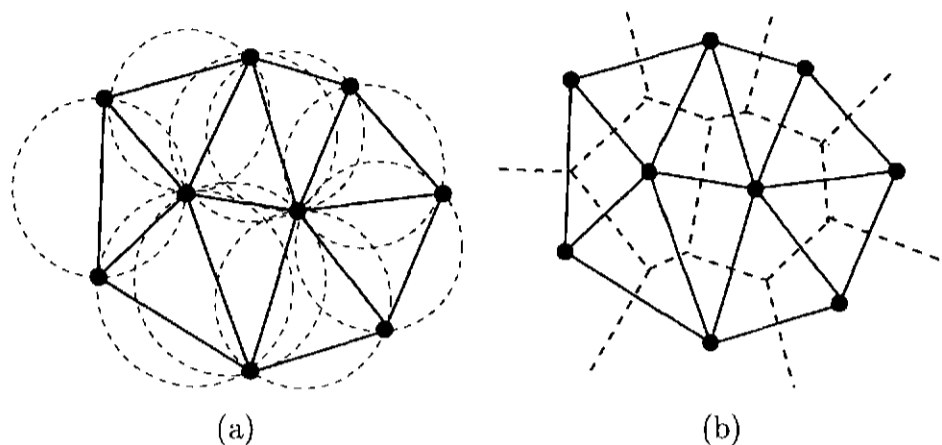


Figura A.1: (a) Nenhuma das circunferências dos triângulos da TD contém pontos no seu interior. (b) TD e seu diagrama de Voronoi (linhas pontilhadas.)

Vários algoritmos são conhecidos para a criação da triangulação de Delaunay a partir de um conjunto de pontos, como o algoritmo de fecho convexo, o dividir-e-conquistar, o algoritmo sweep-line, e o algoritmo incremental. Este último é um dos mais populares devido a sua simplicidade. Uma das vantagens do algoritmo incremental é que ele permite a inserção progressiva de pontos com ajustes locais.

A.2 Algoritmos de Refinamento.

Um dos principais temas de investigação no ramo da geometria computacional é referente a implementação de algoritmos que desenvolvam refinamento de malhas. Estes refinamentos utilizam a introdução de novos vértices ou da modificam a malha mediante a translação de vértices já existentes nela.

Os algoritmos do tipo iterativo como os propostos por Chew [7] e Rupert [29], utilizam inserção de pontos na triangulação de Delaunay.

A.2.1 Algoritmo de Chew.

Paul Chew [7] criou um algoritmo de refinamento de Delaunay de grande interesse, o qual gera uma triangulação de densidade uniforme de elementos.

Dada uma triangulação de Delaunay cuja menor aresta tenha comprimento h_{min} , o algoritmo de Chew quebra todos os triângulos cujos circunraios são maiores que h_{min} , in-

introduzindo um novo vértice no circuncentro de cada um destes triângulos. A inclusão do circuncentro causa uma re-triangulação local que refina a malha original. Pode ser mostrado que o algoritmo nunca introduz arestas com comprimento menor que h_{min} , assim, dois vértices quaisquer são separados por uma distância maior ou igual a h_{min} . O algoritmo terminará inevitavelmente quando os vértices da triangulação estiverem suficientemente próximos.

O algoritmo inicialmente deve subdividir a fronteira inicial (que é composta de segmentos de reta), antes de começar o refinamento. Ao fazer esta subdivisão evita-se a presença de triângulos de baixa qualidade na fronteira da triangulação, os quais poderiam ter seu circuncentro fora do domínio da triangulação. Assim, garante-se que todos os novos triângulos gerados pelo algoritmo ficarão dentro do domínio original.

Em geral, o usuário escolhe um parâmetro h a partir do qual todos os segmentos da fronteira são subdivididos, de modo que cada sub-segmento possua comprimento no intervalo $[h, \sqrt{h}]$. O parâmetro h tem que ser escolhido de tal forma que não existam arestas na fronteira com menor comprimento que h . Devido a não existência de sub-segmentos com comprimento maior que \sqrt{h} , pode ser mostrado que o circuncentro de qualquer triângulo cujo circunraio exceda h ficará na triangulação a uma distância mínima $h/2$ de qualquer sub-segmento, isto garante que os circuncentros dos triângulos sempre estão no interior do domínio.

Outro fato mostrado por Chew é que no final do processo, nenhum triângulo possui o circunraio maior que h e nenhuma aresta tem comprimento menor que h (ou maior que $2h$), assim, a triangulação não contém triângulos com ângulos menores que 30° .

Na figura A.2 é apresentado um exemplo de refinamento de Chew, sob um domínio. A figura A.2a mostra a entrada original. A triangulação de Delaunay do contorno original mais os novos vértices adicionados na fronteira são mostrados em A.2b. Na figura A.2c, os elementos externos da triangulação são eliminados, além disso, é mostrado um dos triângulos de baixa qualidade (o triângulo com linhas pontilhadas) com seu respectivo circuncentro (indicado por um “x”). Depois de inserir um novo ponto steiner no circuncentro e refazendo a triangulação localmente se obtém uma malha como a apresentada na figura A.2d. A figura A.2e mostra a malha depois de inserir 15 novos pontos e finalmente a figura A.2f mostra a malha refinada, depois de inserir 95 novos vértices.

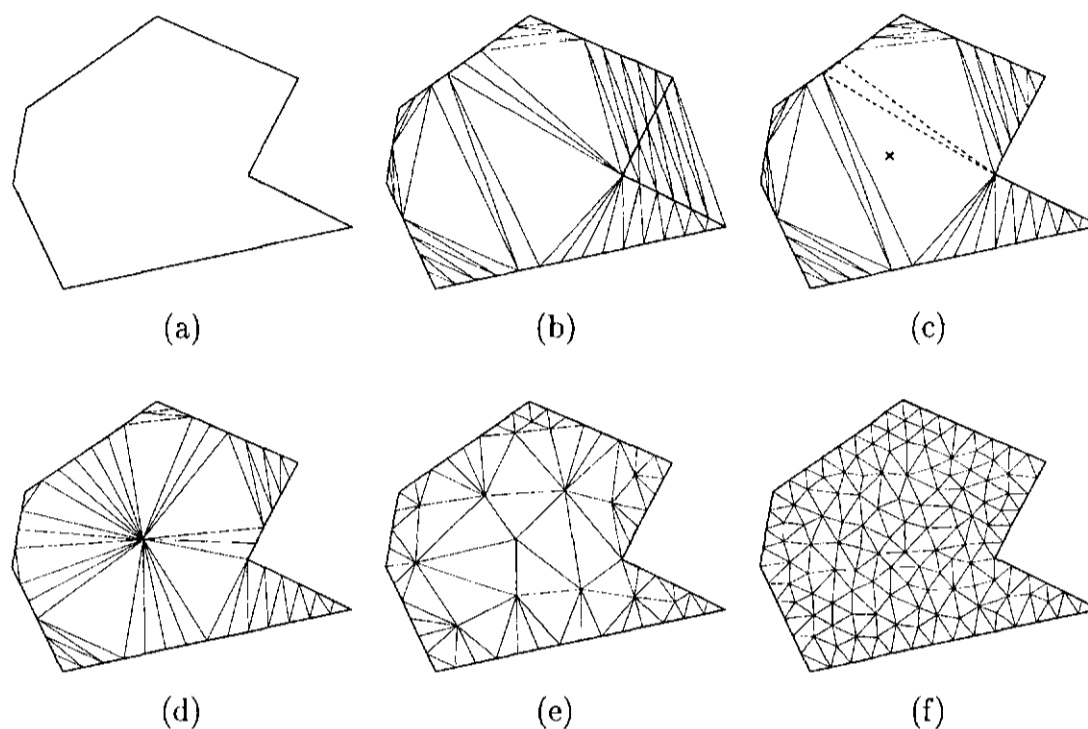


Figura A.2: Algoritmo de Chew. (a) Entrada original. (b) Triangulação de Delaunay do contorno original mais novos vértices na fronteira. (c) Eliminação dos elementos externos da triangulação. O “x” indica o circuncentro do triângulo de baixa qualidade, (d) inserção do ponto steiner no circuncentro e correspondente triangulação. (e) Malha depois de inserir 15 novos pontos e (f) triangulação final.

A.2.2 Algoritmo de Rupert.

O algoritmo de refinamento criado por Rupert [29], apresenta, em geral, melhores resultados que o algoritmo de Chew, pois as malhas geradas possuem elementos de boa qualidade, que podem variar rapidamente de tamanho. Estas vantagens fazem este método muito apropriado para geração de malhas voltadas a problemas de simulação numérica, já que ele gera maior quantidade de triângulos em regiões onde a precisão dos cálculos deve ser maior, aumentando assim a velocidade dos cálculos.

O algoritmo de Rupert pode refinar qualquer triangulação de Delaunay com ou sem restrições. O algoritmo trata os segmentos iniciais de forma única, utilizando apenas os vértices para o cálculo da triangulação de Delaunay. Os segmentos da entrada original são garantidos como uma consequência natural do algoritmo.

O algoritmo de Rupert refina a malha mediante a inserção de novos vértices (os quais

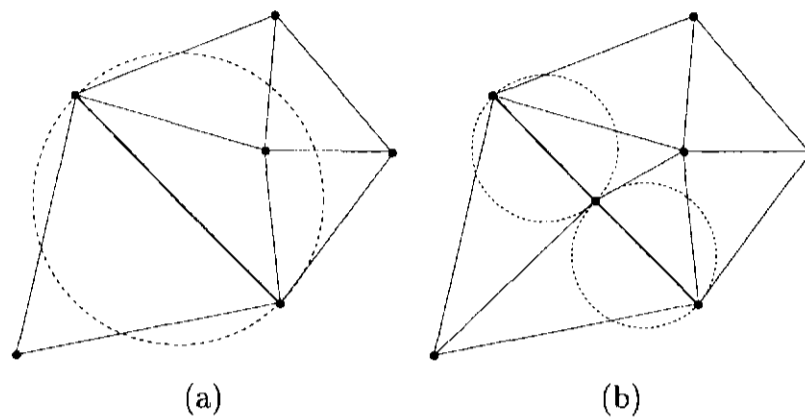


Figura A.3: (a) Aresta violada, (b) duas novas arestas geradas pela inserção do novo vértice no meio da aresta anterior.

cumprem com as propriedades de Delaunay), até que os triângulos cumpram um critério de qualidade. Como no algoritmo de Chew, o algoritmo de Rupert pode dividir cada segmento original em sub-segmentos mesmo que não seja no primeiro passo do algoritmo, isto é, a subdivisão é permitida no decorrer do programa. A inserção de vértices é governada pelas seguintes regras:

- O círculo diametral de um segmento é o (único) menor círculo que contém o segmento. Um segmento é dito violado se um vértice está dentro do seu círculo diametral ou se o segmento não aparece como uma aresta da triangulação de Delaunay. Qualquer segmento violado é imediatamente dividido em dois segmentos inserindo um novo vértice no seu ponto médio (como exemplo a figura A.3a mostra uma semi-aresta violada, e a divisão da aresta em duas novas arestas pela inserção de um novo vértice, figura A.3b). Estes sub-segmentos têm círculos diametraes menores, e podem ou não ser sub-segmentos violados. Quando os sub-segmentos gerados são violados, estes são divididos novamente até que não existam sub-segmentos violados.
- Qualquer triângulo cuja relação entre seu circunraio dividido pelo comprimento da menor aresta é maior que um valor limite B (ex: $B \approx \sqrt{2}$), é eliminado pela triangulação ao inserir um novo vértice em seu circuncentro. Se um novo vértice viola algum segmento da triangulação, então ele não é inserido, em vez disso, todos os segmentos que seriam violados são subdivididos.

Pode ser visto que no algoritmo de Rupert, segmentos violados tem prioridade sobre

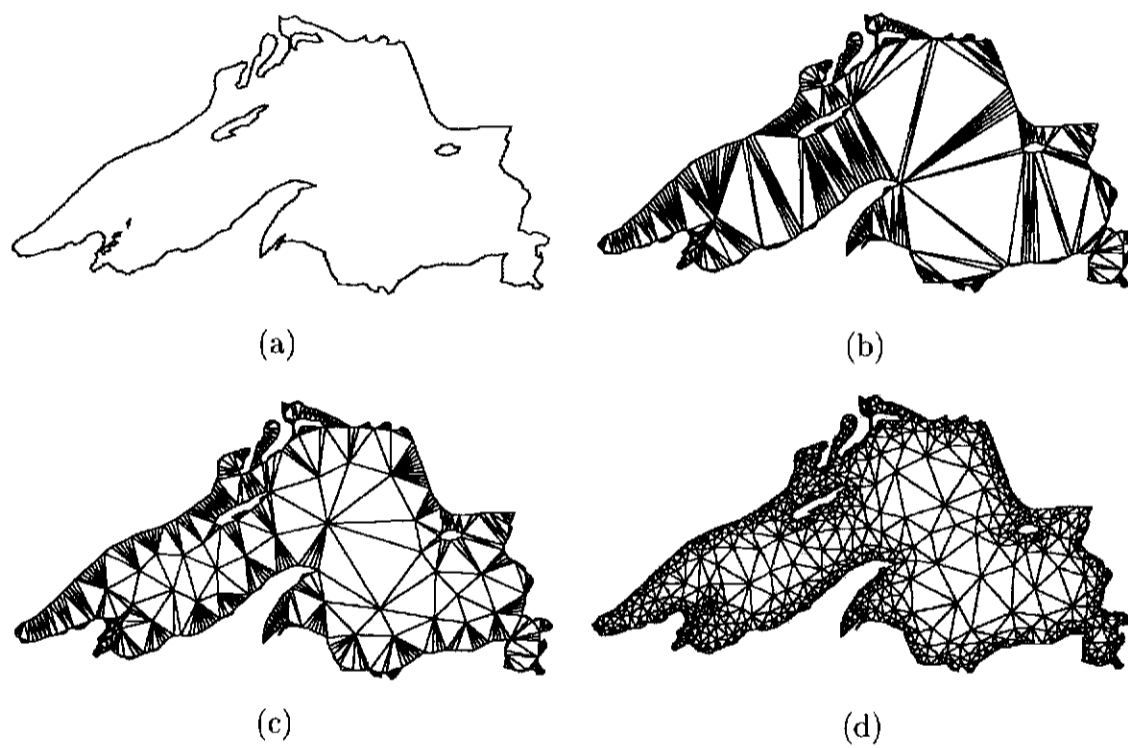


Figura A.4: (a) Entrada inicial, (b) triangulação de Delaunay, (c) triangulação após o refinamento de Rupert, com um ângulo mínimo de 5° , (d) triangulação com ângulo mínimo de 25° . [30]

triângulos de baixa qualidade. Rupert demonstra em seu trabalho [29] que este algoritmo sempre finaliza em um número finito de passos. Além disso, os triângulos gerados apresentam boa qualidade em sua forma, pode não se garantir que os triângulos não tem ângulos menores que 20.7° (este ângulo depende do parâmetro B). Na figura A.4 é apresentado um exemplo de malha refinada com o algoritmo de Rupert, para diferentes valores de B .

Apêndice B

Estrutura de Dados.

No desenvolvimento deste trabalho foi utilizada a estrutura de dados *Singular Handle-Edge* (SHE) [26], para a representação de malhas não estruturadas em duas dimensões. Ela tem a habilidade de representar vértices singulares e curvas do contorno. A introdução de vértices singulares é comum nos processos de inserção e remoção de triângulos nas malhas não estruturadas. Por exemplo, é inevitável a presença de vértices singulares durante a geração de malhas triangulares com buracos. Em aplicações numéricas as condições dos contornos associados à uma equação diferencial podem ser manipuladas mais facilmente por uma representação explícita dos elementos de fronteira.

B.1 A estrutura Singular Handle-Edge.

A Singular Handle-Edge é organizada em termos de sete entidades representadas explicitamente, os quais são:

- **Malha:** Representa cada componente conexa da malha.
- **Vértice:** Representa os vértices da malha.
- **Semi-aresta:** Representa uma aresta contida em um triângulo.
- **Estrela do Vértice:** Representa cada componente singular incidente a um vértice
- **Face:** Representa as faces ou células da malha.
- **Curva de Contorno:** Representa a curva do contorno da malha.

- **Aresta de Contorno:** Representa as arestas contidas na curva do contorno.

As entidades Semi-aresta, Estrela de Vértice e Arestas de Contorno são armazenadas em listas circulares dinâmicas, enquanto o resto dos elementos são armazenados em listas não circulares.

B.1.1 O elemento Malha.

Cada elemento malha aponta para a lista de seus vértices, faces e curvas de contorno (fig B.1)¹.

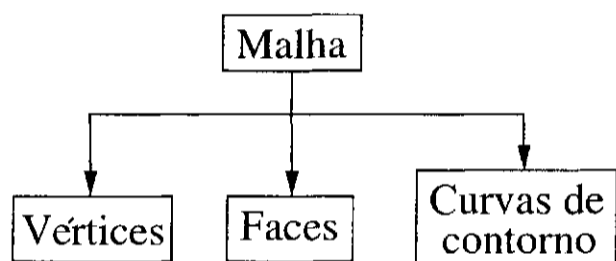


Figura B.1: Relações do elemento Malha.

B.1.2 O elemento Vértice.

Cada elemento vértice aponta para sua lista Estrela de Vértices e volta ao elemento Malha o qual o contém (fig B.2).

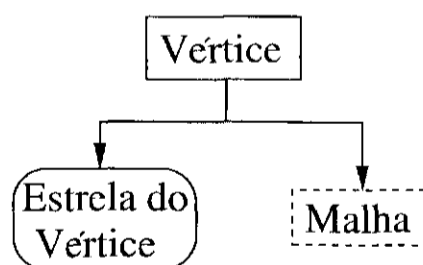


Figura B.2: Relações do elemento Vértice.

¹Neste e nos próximos diagramas, as entidades que estão numa caixa quadrada contínua representam listas não circulares, em caixas quadradas pontilhadas representam ponteiros e em caixas circulares representam listas circulares.

O elemento Semi-aresta.

O elemento Semi-aresta deve ser sempre orientado, e aponta o vértice que está em seu extremo inicial (fig B.3). O elemento também aponta o elemento Semi-aresta da face adjacente se não é uma aresta de contorno. A aresta apontada recebe o nome de “Semi-aresta Mate”. Se a Semi-aresta pertence ao contorno, então ela aponta para o elemento Aresta de Contorno (fig B.4).

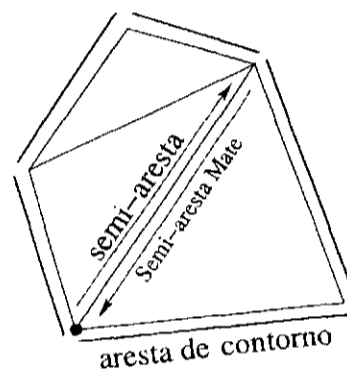


Figura B.3: Exemplo de Semi-aresta, Semi-aresta mate e Aresta de Contorno. O ponto preto representa o vértice ao qual a Semi-aresta aponta.

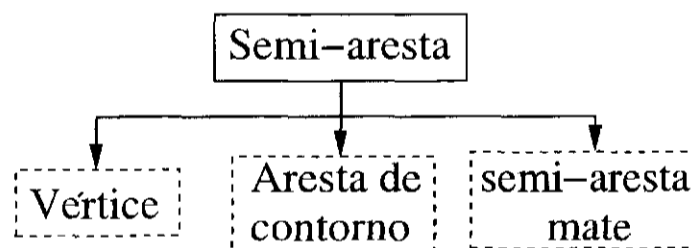


Figura B.4: Relações do elemento Semi-aresta.

B.1.3 O elemento Estrela do Vértice.

Cada elemento Estrela do Vértice aponta para um elemento Semi-aresta que está associada ao vértice. Se o vértice é singular, ele tem uma lista circular de Estrela de Vértice, e cada um dos elementos desta lista apontará para um elemento Semi-aresta no contorno de cada componente que forma a singularidade (fig B.5).

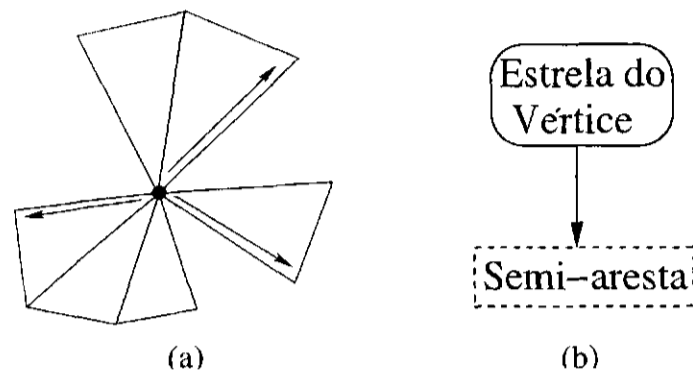


Figura B.5: (a) Exemplo da Estrela do Vértice com uma singularidade, (b) relações do elemento Estrela do Vértice.

B.1.4 O elemento Face.

Cada elemento Face aponta para uma lista circular de Semi-arestas que define sua orientação (fig B.6).

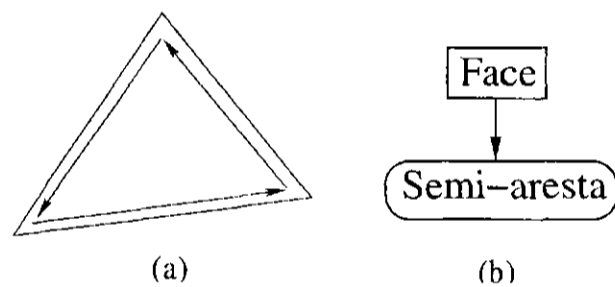


Figura B.6: (a) Exemplo de Face orientada, (b) relações do elemento Face.

B.1.5 Os elementos Curva de Contorno e Arestas de Contorno.

O elemento Curva de Contorno aponta para uma lista circular de suas Arestas de Contorno. Cada elemento Aresta de contorno aponta o elemento Curva de Contorno na qual ele está contido e o elemento Semi-aresta que o representa (fig B.7)

B.2 Implementação.

A estrutura de dados SHE e o completo conjunto de métodos para manipular a armazenagem das informações estão implementados em C++.

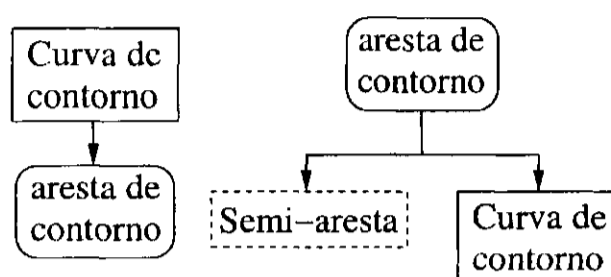


Figura B.7: Relações dos elementos Curva de Contorno e Arestas de Contorno.

Um mecanismo chamado *iterator* foi implementado para percorrer os elementos do SHE, o qual permite a exploração de listas com um simples FOR. Métodos `Begin()` e `End()` definidos para cada classe são responsáveis por iniciar e finalizar o *iterator*. Por exemplo, o seguinte código pode ser empregado para imprimir as coordenadas de todos os vértices na malha ao percorrer a lista de Vértice:

```
Iterator<SHE.Vertice> iv;

for (iv = m -> Begin_vertex(); iv != m->End_vertex(); ++iv)
    cout << iv->Get_x() << iv->Get_y();
```

onde `m` é um ponteiro para o elemento Malha. A principal vantagem do *iterator* é que ele encapsula as listas e padroniza o acesso a elas.

·
·
·

Bibliografia

- [1] D.A. Anderson, J.C. Tannehill, e R.H. Pletcher. Computational Fluid Mechanics and Heat Transfer. *Ed McGraw-Hill*. 1984.
- [2] S.N. Atluri, T. Zhu. A new meshless local Petrov-Galerkin (MLPG) approach. *Comput. Mech.* 22, pp 117-127, 1998.
- [3] I. Babuška, J.M. Melenk. The partition of unity method. *Int. J. Numer. Meth. Eng.* 40, pp 727-758, 1997.
- [4] T. Belytschko, Y. Krongauz, D. Fleming, P. Krysl. Meshless methods: An Overview and Recent Developments. *Comput. Meth. Appl. Mech. Engrg.* 139, pp 2-47, 1996.
- [5] T. Belytschko, Y.Y. Lu, L. Gu. Element-free Galerkin methods. *Int. J. Meth. Eng.* 37, pp 229-256, 1994.
- [6] J.J. Benito. F. Ureña e L. Gavete. Influence of Several Factors in the Generalized Finite Difference Method. *Applied Mathematical Modelling*, 25, pp 1039-1053, 2001.
- [7] L.P. Chew. Constrained Delaunay Triangulations. *Algorithmica* 4:(1), pp 97-108, 1989.
- [8] C.A.M. Duarte. A review of some meshless methods to solve partial differential equations. Technical Report 95-06, TICAM, The University of Texas at Austin, 1995.
- [9] C.A.M. Duarte, J. T. Oden. h - p Cloud - An h - p meshless method. *Numer. Meth. Partial Diff. Eqs.* 12, pp 673-705, 1996.
- [10] L. Demkowicz, A. Karafiat e T. Liszka. On some convergence results for FDM with irregular mesh. *Computer Methods in Applied Mechanics and Engineering* 42, pp 343-355, 1984.

- [11] J.H. Ferziger, P Milovan. Computational Methods for Fluid Dynamics. *Ed. Springer-Verlag*, 1999.
- [12] R. Franke e G. Nielson. Smooth interpolation of large sets of scattered data. *International Journal of Numerical Methods in Engineering*, 15, pp 1691-1704, 1980.
- [13] R.A. Gingold, J.J. Monaghan, Smoothed particle hydrodynamics: theory and application to non-spherical stars, *Mon. Not. Roy. Astron. Soc.* 181, pp 375-389, 1977.
- [14] A. Gossler. Moving Least-Squares: A Numerical Differentiation Method for Irregularly Spaced Calculation Points. *Sandia technical Report, SAND2001-1669*, 2001.
- [15] M. Griebel, T. Dornseifer e T. Neunhoeffler. Numerical Simulation in Fluid Dynamics: A Practical Introduction. *SIAM*, 1998.
- [16] P.S. Jensen. Finite difference techniques for variable grids. *Comp. Structures* 2, pp 17-29, 1972.
- [17] H.S. Kou, e C.L. Lee. Implementation of Generalized Finite Difference Formulas and Truncation Errors for Each Derivative. *International Communications in Heat and Mass Transfer*, 21(3), pp 359-370, 1994.
- [18] P. Krysl e T. Belytschko, An efficient linear-precision partition of unity basis for unstructured meshless methods. *Commun. Num. Meth. Eng.*, 16:(4), pp 239-255, 1999.
- [19] T. Liszka e J. Orkisz. The finite difference method at arbitrary irregular grids and its application in applied mechanics. *Computers and Structures*. 11, pp 83-95, 1980.
- [20] W.K. Liu, S. Jun, Y.F. Zhang. Reproducing kernel particle methods. *Int. J. Numer. Meth. Fluids*. 20, pp 1081-1106, 1995.
- [21] Y. Luo, U. Häussler. A generalized finite-difference method based on minimizing global residual. *Comput. Methods Appl. Mech. Engrn.* 191, pp 1421-1438, 2002.

- [22] J.S. Marshall e J.R. Grand. A Lagrangian Vorticity Collocation Method for Viscous, Axisymmetric Flows With and Without Swirl. *J. Comput Phys.*, Vol. 138, pp 302-330, 1997.
- [23] J.J. Monaghan. An introduction to SPH. *Comput. Phys. Comm.* 48, pp 89-96, 1988.
- [24] B. Nayroles, G. Touzot, P. Villon. Generalizing the finite element method: diffuse approximation and diffuse elements. *Comput. Mech.* 10, pp 307-318, 1992.
- [25] L.G. Nonato, A. Castelo, R. Minghim e J.E.S. Batista. Morse Operators for Digital Planar Surfaces and their Application to Image Segmentation. *submitted*, 2001.
- [26] L.G. Nonato, A. Castelo, M.C. Ferreira. A topological Approach for Handling Triangle Insertion and Removal into Two-Dimensional Unstructured Meshes. *Cuadernos de Computação, ICMC-USP.* 3(02), pp 221-243, 2002.
- [27] H. Nguyen e J. Reynen. A space-time finite element method for hyperbolic equation. *Proc. 3rd Internat. Symp. on Numerical Methods in Engineering*, Paris, 1983.
- [28] N. Perrone e R. Kao. A general finite difference method for arbitrary mesh. *Computers and Structures.* 5, pp 647-664, 1975.
- [29] J. Rupert. A Delaunay Refinement Algorithm for Quality 2-Dimensional Mesh Generation. *Journal of Algorithms*, 18(3), pp 548-585, 1995.
- [30] J.R. Sheiwchuk. Delaunay Refinement Mesh Generation. *Thesis of Doctor of Philosophy, Carnegie Mellon University, Pittsburgh*, 1997.
- [31] D. Shepard. A two dimensional function for irregularly spaced data. *ACM National Conference*. 1968.
- [32] F. Ureña , J.J. Benito L. e R. Alvarez, Método adaptativo para la resolución de ecuaciones diferenciales en derivadas parciales de segundo orden utilizando Diferencias Finitas Generalizadas. *Aplicadon a Mathematical Modelling*. ref: MC/EM/AMM.2884 (Decembro 2001).