

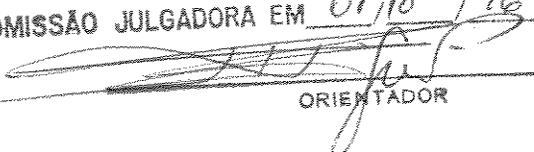
UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA

GERADOR AUTOMÁTICO DE MALHAS ESTRUTURADAS
PARA EQUAÇÕES DE NAVIER-STOKES
E VISUALIZAÇÃO DE ESCOAMENTOS

Autor: Carlos Alberto de Almeida Vilela
Orientador: José Ricardo Figueiredo

09/96

ESTE EXEMPLAR CORRESPONDE A REDAÇÃO FINAL DA
TESE DEFENDIDA POR Carlos Alberto de
Almeida Vilela E APROVADA PELA
COMISSÃO JULGADORA EM 01/10/96.


ORIENTADOR

V711g

30013/BC

**UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
DEPARTAMENTO DE ENERGIA**

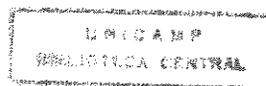
**GERADOR AUTOMÁTICO DE MALHAS ESTRUTURADAS
PARA EQUAÇÕES DE NAVIER-STOKES
E VISUALIZAÇÃO DE ESCOAMENTOS**

Autor: Carlos Alberto de Almeida Vilela
Orientador: José Ricardo Figueiredo

Curso: Engenharia Mecânica
Área de Concentração: Térmica e Fluidos

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

Campinas, 1996
S.P. - Brasil



UNIDADE	-BC
N.º DE REGISTRO	JUNI CAMP
V.º	V711g
TÍTULO	30013
PREÇO	281,97
DATA	06/05/97
N.º CPD	

CM-00099235-4

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

V711g

Vilela, Carlos Alberto de Almeida

Gerador automático de malhas estruturadas para equações de Navier-Stokes e visualização de escoamentos / Carlos Alberto de Almeida Vilela. --Campinas, SP: [s.n.], 1996.

Orientador: José Ricardo Figueiredo.

Dissertação (mestrado) - Universidade Estadual de Campinas, Faculdade de Engenharia Mecânica.

1. Visualização de fluxo. 2. Método dos elementos finitos. 3. Geração numérica de malhas (Análise numérica).
I. Figueiredo, José Ricardo. II. Universidade Estadual de Campinas. Faculdade de Engenharia Mecânica. III. Título.

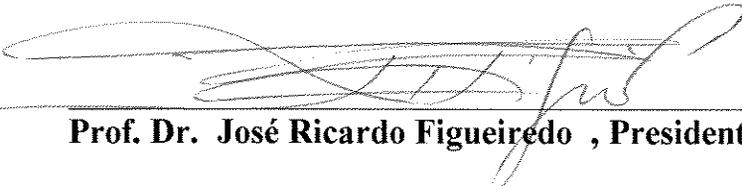
**UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
DEPARTAMENTO DE ENERGIA**

DISSERTAÇÃO DE MESTRADO

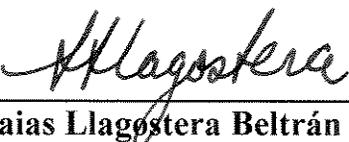
**GERADOR AUTOMÁTICO DE MALHAS ESTRUTURADAS
PARA EQUAÇÕES DE NAVIER-STOKES
E VISUALIZAÇÃO DE ESCOAMENTOS**

Autor: Carlos Alberto de Almeida Vilela

Orientador: José Ricardo Figueiredo



Prof. Dr. José Ricardo Figueiredo , Presidente



Prof. Dr. Jorge Isaias Llagostera Beltrán



Prof. Dr. Philippe Remy Bernard Devloo

Campinas, 30 de setembro de 1996

Tende cuidado, para que ninguém vos faça presa sua, por meio de filosofias e vãs sutilezas, segundo a tradição dos homens, segundo os rudimentos do mundo, e não segundo Cristo.

Colossenses 2:8

Porque não temos que lutar contra a carne e o sangue, mas sim contra os principados, contra as potestades, contra os príncipes das trevas deste século, contra as hostes espirituais da maldade, nos lugares celestiais.

Efésios 6:12

Consagrado ao Senhor Deus, Todo Poderoso.

Agradecimentos

Presto esta homenagem a quem teve muita importância, em todos os sentidos, durante todo o tempo de desenvolvimento deste trabalho:

Ao meu pai, minha mãe, minhas tias e tios, por todas as repreensões e ajuda.

A José Ricardo Figueiredo por toda a confiança.

Aos amigos fieis Fábio, Francis, Paulo, Edivaldo, Rodrigues, Mirkos, Chan.

Aos professores, com os quais tive a oportunidade de adquirir mais conhecimentos.

Resumo

Almeida Vilela, Carlos Alberto, Gerador automático de malhas estruturadas para equações de Navier-Stokes e visualização de escoamentos, Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, 1996, 144p, Dissertação (Mestrado).

Este trabalho apresenta o esquema e desenvolvimento de um pacote interativo para geração de malha estruturada em domínio qualquer pelo método de solução de equações elípticas transformadas. Fazem parte do pacote: um programa gráfico de geração geométrica para o domínio físico e visualização da malha gerada, um solvedor para gerar a malha e um para aproximar as equações de Navier-Stokes e a equação da Energia. Com uma malha gerada numericamente, são aproximadas as equações de Navier-Stokes e da Energia para escoamentos de fluidos newtonianos incompressíveis, e os resultados são comparados com soluções padrões conhecidas para o estudo da adequação de malha para cada problema.

Para a solução das equações geradoras de malha foram utilizados os métodos dos elementos finitos e diferenças finitas, e para as equações de Navier-Stokes e da Energia o método dos elementos finitos.

Palavras chave

Visualização de escoamentos, Elementos finitos, Geração de malhas.

Conteúdo

Nomenclatura	iii
Lista de tabelas	v
Listas de figuras	vi
1 Introdução	1
1-1 Apresentação geral	1
1-2 Equações de Navier-Stokes	4
1-3 Equação da Continuidade	6
1-4 Equação da Energia	6
2 Geração da malha	8
2-1 Sistemas de coordenadas	8
2-2 Equações geradoras de malha	10
3 Métodos numéricos	19
3-1 Apresentação	19
3-2 Métodos de aproximação por resíduos ponderados	21
3-3 Método dos elementos finitos	24
4 Tratamento numérico das equações	32
4-1 Equações geradoras de malha	32
4-2 Equações de Navier-Stokes e da Continuidade	47
4-3 Equação da Energia	54
4-4 Técnica de solução das matrizes	57
4-5 Critério de convergência	59

5	Tópicos computacionais	60
5-1	Programas GridGen, Gridg e Therm	60
6	Apresentação e comparação de resultados	82
6-1	Geração da malha	82
6-2	Escoamento em cavidade quadrada com fronteira deslizante	88
6-3	Escoamento em torno de um cilindro	101
6-4	Escoamento em uma expansão abrupta	116
7	Conclusões	129
	Referências bibliográficas	131
	Anexos	133
A	Procedimento para geração de malhas	133

Nomenclatura

Símbolos latinos

A - lado esquerdo da equação de Navier-Stokes

B - termos que representam força de campo na equação de Navier-Stokes

C - condições de contorno de derivada na equação de Navier-Stokes

C_p - calor específico

k - condutividade térmica

t - temperatura

u - componente em x da velocidade

v - componente em y da velocidade

p - pressão

J - jacobiano da transformação entre plano físico e transformado

J - jacobiano da transformação entre plano físico e normalizado, ou entre plano transformado e normalizado

P - função de controle para as linhas da malha

Q - função de controle para as linhas da malha

W - função ponderadora

N - função de forma para elemento de oito nós

M - função de forma para elemento de quatro nós

R - resíduo

Símbolos gregos

α - coeficiente da equação geradora de malha

β - coeficiente da equação geradora de malha

γ - coeficiente da equação geradora de malha

ξ - coordenada do plano transformado

η - coordenada do plano transformado

λ - coordenada do elemento padrão

κ - coordenada do elemento padrão

μ - viscosidade do fluido

ρ - densidade do fluido

θ_i - coeficientes para aproximação por resíduos ponderados

ϕ_i - funções de aproximação estabelecidas (resíduos ponderados)

φ - coeficientes da função de aproximação de variáveis no elemento

Ω - área elementar

ϖ - valores ponderadores relativos à posição dos pontos de Gauss

Φ - dissipação viscosa

Φ - lado direito da equação, para x , da geração de malha

Θ - lado direito da equação, para y , da geração de malha

$\Delta\lambda$ - metade da dimensão em λ do elemento padrão

$\Delta\kappa$ - metade da dimensão em κ do elemento padrão

ω - matriz de incógnitas para na equação de Navier-Stokes

Lista de tabelas

Capítulo 4

4.1	Fatores ponderadores e posições dos pontos de Gauss	38
4.2	Posições da célula para pontos de Gauss	43

Capítulo 5

5.1	Principais arquivos cabeçalhos utilizados	62
-----	---	----

Capítulo 6

6.1	Iterações para cálculo das malhas para cavidade	86
6.2	Iterações para cálculo das malhas para cilindro	86
6.3	Iterações para cálculo das malhas para expansão	86
6.4	Características das malhas - cavidade	89
6.5	Propriedades do ar	89
6.6	Velocidades U da fronteira superior	90
6.7	Características das malhas - cilindro	102
6.8	Velocidades U na fronteira esquerda	103
6.9	Características das malhas - expansão	117
6.10	Velocidades U médias	118

Lista de figuras

Capítulo 1

- | | | |
|-----|---|---|
| 1.1 | Fluxograma da sequência de uso dos programas | 3 |
| 1.2 | Condições de contorno para equações de Navier-Stokes e da Energia | 7 |

Capítulo 2

- | | | |
|------|---|----|
| 2.1 | Representação do plano físico | 9 |
| 2.2 | Representação do plano transformado | 9 |
| 2.3 | Representação do plano normalizado | 10 |
| 2.4 | Domínios simples conectados e multiconectados | 11 |
| 2.5 | Plano físico e transformado | 12 |
| 2.6 | Malha gerada pelo método de Amsden | 12 |
| 2.7 | Malha gerada pelo método de Truong | 13 |
| 2.8 | Caminho das linhas η | 15 |
| 2.9 | Região transformada retangular | 15 |
| 2.10 | Região transformada mantendo a forma da região física | 16 |
| 2.11 | Região transformada configuração slab | 16 |
| 2.12 | Região transformada configuração slit | 17 |
| 2.13 | Região transformada configuração tipo O | 17 |
| 2.14 | Região transformada configuração tipo C | 18 |

Capítulo 3

- | | | |
|-----|-------------------------------------|----|
| 3.1 | Representação gráfica de uma função | 19 |
| 3.2 | Célula de nove pontos | 20 |
| 3.3 | Elementos típicos bidimensionais | 25 |
| 3.4 | Triângulo de Pascal | 27 |

3.5	Elemento retangular de quatro e oito nós	28
3.6	Comportamento das funções de forma	31
Capítulo 4		
4.1	Elemento característico da malha transformada	32
4.2	Plano transformado, elemento característico	36
4.3	Pontos de integração numérica, pontos de Gauss	37
4.4	Sub-regiões do elemento padrão	42
4.5	Célula em torno do ponto de Gauss	42
4.6	Numeração local e global dos nós	44
Capítulo 5		
5.1	Janela do arquivo de projeto <i>gridgen.prj</i>	61
5.2	Janela principal com exemplo para cavidade - GridGen	63
5.3	Relação dos comandos disponíveis no programa	64
5.4	Quadro de diálogo para construção de linhas	65
Capítulo 6		
6.1	Malhas utilizadas na cavidade	83
6.2	Malhas utilizadas no cilindro	84
6.3	Malhas utilizadas na expansão	85
6.4	Escoamento em cavidade, coordenadas e condições de contorno	88
6.5	Fronteira de referência para cálculo do número de Re	90
6.6	Componente u ao longo do centro geométrico, $x=0.5$, $Re=100$	91
6.7	Componente v ao longo do centro geométrico, $y=0.5$, $Re=100$	92
6.8	Componente u ao longo do centro geométrico, $x=0.5$, $Re=400$	93
6.9	Componente v ao longo do centro geométrico, $y=0.5$, $Re=400$	94
6.10	Componente u ao longo do centro geométrico, $x=0.5$, $Re=1000$	95
6.11	Componente v ao longo do centro geométrico, $y=0.5$, $Re=1000$	96
6.12	Função corrente - cavidade	97
6.13	Zonas de velocidades absolutas - cavidade	98

6.14	Vetores de velocidade - cavidade	98
6.15	Zonas de pressão constante - cavidade	99
6.16	Zonas de temperatura constante - cavidade	99
6.17	Escoamento em torno de cilindro, coordenadas e condições de contorno	101
6.18	Comprimento da zona de recirculação	103
6.19	Superfície do cilindro - cálculo da pressão	104
6.20	Superfície do cilindro - cálculo do número de Nu	104
6.21	Posição dos pontos para cálculo do número de Nu	105
6.22	Comprimento da zona de recirculação - resultados	106
6.23	Coefficiente de pressão na superfície do cilindro - $Re=5$	107
6.24	Coefficiente de pressão na superfície do cilindro - $Re=10$	108
6.25	Coefficiente de pressão na superfície do cilindro - $Re=40$	109
6.26	Nusselt local na superfície do cilindro - $Re=10$	110
6.27	Função corrente - cilindro	111
6.28	Zonas de velocidades absolutas - cilindro	112
6.29	Vetores de velocidade - cilindro	112
6.30	Zonas de pressão constante - cilindro	113
6.31	Zonas de temperatura constante - cilindro	113
6.32	Posição de nós na zona de recirculação - cilindro	114
6.33	Escoamento em uma expansão, coordenadas e condições de contorno	116
6.34	Comprimento da zona de recirculação	117
6.35	Posição da linha de referência	118
6.36	Comprimento da zona de recirculação - resultados	120
6.37	Perfis de velocidade u ao longo do canal - $Re=50$, $Re=100$, $Re=150$	121
6.38	Coefficiente de pressão na linha de referência - $Re=50$	122
6.39	Coefficiente de pressão na linha de referência - $Re=100$	123
6.40	Coefficiente de pressão na linha de referência - $Re=150$	124
6.41	Função corrente - expansão	125
6.42	Zonas de velocidades absolutas - expansão	126
6.43	Vetores de velocidade - expansão	126

6.44	Zonas de pressão constante - expansão	127
6.45	Zonas de temperatura constante- expansão	127

Anexo A

a.1	Exemplo de malha simples conectada	134
a.2	Vértices da malha física - simples conectada	135
a.3	Numeração na geração da geometria - simples conectada	136
a.4	Comando para tipo de malha	136
a.5	Comando para identificar o contorno gerado	137
a.6	Comando para identificar os vértices	137
a.7	Ferramentas de desenho	137
a.8	Sequência de construção da geometria - simples conectada	138
a.9	Exemplo de malha multi conectada	139
a.10	Vértices da malha física - multi conectada	140
a.11	Numeração na geração da geometria - multi conectada	141
a.12	Sequência de construção da geometria - multi conectada	142
a.13	Malha original para cavidade	143
a.14	Atração para linha $\xi=11$	143
a.15	Atração para linha $\eta=11$	143
a.16	Atração para ponto $\xi=11, \eta=11$	144
a.17	Atração para ponto $\xi=11, \eta=11$	144
a.18	Atração para ponto $\xi=11, \eta=11$	144

Capítulo 1

Introdução

1-1 Apresentação geral

As equações de Navier-Stokes são equações gerais que descrevem o escoamento de um fluido newtoniano qualquer relacionando as forças nele atuantes com a variação da quantidade de movimento.

São conhecidas, segundo White (1991), aproximadamente 80 soluções analíticas, considerando geometrias específicas e utilizando simplificações tais que irão reduzir as equações completas para admitirem soluções analíticas.

Aproximações assim podem ser feitas por exemplo, considerando fluido incompressível e escoamento sem viscosidade, resultando nas equações de Euler e Bernoulli, ou ainda considerando um escoamento em camada limite onde se obtêm soluções por similaridade das equações simplificadas, representando casos como escoamento em placa plana e escoamento em cunha.

Ainda não existe um método analítico geral para tratar estas equações para um problema de escoamento arbitrário, por isso técnicas numéricas têm surgido para contornar as limitações das soluções analíticas, que se restringem a geometrias e condições específicas. Alguns dos métodos numéricos mais conhecidos e utilizados são: método das diferenças finitas, o método dos volumes finitos e método dos elementos finitos. Estes métodos podem ser

aplicados a vários tipos de equações diferenciais, e cada método tem uma característica própria que o torna mais vantajoso para o uso, dependendo das equações a serem discretizadas, geometria do problema, etc.

Tanto no método das diferenças finitas como no dos volumes finitos e no dos elementos finitos é necessário que o domínio seja dividido em partes, células ou elementos formadores da malha que dará suporte para aproximações das equações diferenciais.

Como a aproximação das equações se dará nos nós da malha, fica claro que a distribuição dos nós é de fundamental importância para uma boa aproximação dos resultados. Toda esta preocupação em conseguir uma malha adequada pode ser confirmada pelos muitos trabalhos publicados relacionados a este assunto, por exemplo Thompson (1985), que apresenta estudos sobre diversas maneiras de construção de malhas que se diferenciam para cada caso de equação diferencial, desde malhas de simples construção por interpolação algébrica, até malhas adaptativas, de construção mais trabalhada.

Em seu livro, ele afirma que para algoritmos baseados nas equações de Navier-Stokes, é preferível linhas de coordenadas que se aproximam das linhas de corrente do escoamento.

Outro aspecto importante é também mencionado por Maliska (1995), que apresenta argumentos sobre o uso de malhas estruturadas para casos onde há isosuperfícies como solução, e ele também enfatiza o fato de que não há perigo de cruzamento de linhas de coordenadas quando geradas como solução numérica.

Então o uso de malhas, como sendo soluções de equações diferenciais, é fundamentado nestes pontos favoráveis, e também pela maior facilidade que se tem para controle da distribuição dos nós da malha.

Como ferramenta básica este trabalho emprega o programa de livre uso *Fluid*, desenvolvido por Taylor (1981). Este programa utiliza malhas

quaisquer formadas por elementos quadrangulares de oito nós, que devem ser fornecidos previamente, e aproxima as equações de Navier-Stokes por elementos finitos, como será descrito.

O objetivo deste trabalho, foi primeiramente o desenvolvimento de um programa para gerar automaticamente a malha. Empregou-se para isto, o método dos elementos finitos para resolver equações elípticas que dariam como resultado as posições dos nós formadores da malha.

Em segundo lugar, foi ampliado o trabalho com a inclusão da equação da energia para fenômenos de convecção forçada e o emprego de técnicas de visualização do escoamento por computador.

Todo o trabalho que será apresentado a seguir, foi de um certo modo compilado e utilizado para o desenvolvimento de três programas computacionais que serão mencionados ao longo do texto.

Foram desenvolvidos os programas *GridGen*, *Gridg*, *Therm*, e foi adaptado o programa *Fluid*. Cada programa apresenta uma característica própria, e que é utilizado independentemente dos outros, mas que formam uma sequência lógica de cálculo, mostrada no fluxograma da figura 1.1:

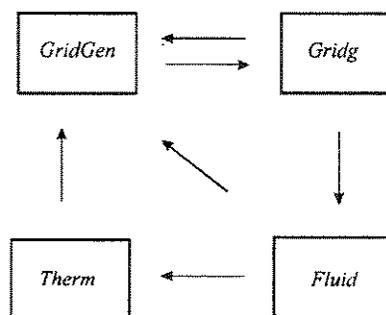


Figura 1.1: fluxograma da sequência de uso dos programas

O programa *GridGen* é responsável por:

- Informação gráfica da geometria física do problema.
- Geração da malha transformada para o problema.
- Visualização da malha gerada numericamente.
- Informação das condições de contorno para *Fluid* e *Therm*.
- Visualização dos resultados obtidos por *Fluid* e *Therm* por meio de gráficos coloridos.

Um procedimento para informar geometrias físicas, pode ser encontrado no Anexo A.

O programa *Gridg* é responsável por:

- Gerar a malha física a partir da malha transformada.

O programa *Fluid* é responsável por:

- Aproximar o campo de escoamento.

O programa *Therm* é responsável por:

- Aproximar o campo de temperaturas.

1-2 Equações de Navier-Stokes

O campo de velocidades no problema será calculado pelas equações de Navier-Stokes tratadas no plano físico, descritas abaixo com as seguintes condições, Fox (1988) :

- Caso bidimensional
- Escoamento em regime permanente
- Fluido newtoniano
- Escoamento incompressível
- Propriedades constantes

$$\begin{aligned} \rho \left(u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} \right) &= -\frac{\partial p}{\partial x} + \mu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) + \rho g_x \\ \rho \left(u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} \right) &= -\frac{\partial p}{\partial y} + \mu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) + \rho g_y \end{aligned} \quad (1.1)$$

onde :

- ρ - densidade do fluido
- μ - viscosidade do fluido
- u - velocidade paralela ao eixo x
- v - velocidade paralela ao eixo y
- p - pressão
- g_x - força de campo atuante em x
- g_y - força de campo atuante em y

As condições de contorno, representadas graficamente na figura 1.2, em um elemento que pode fazer parte da fronteira do problema aplicadas ao programa *Fluid*, são :

- Essenciais $\begin{cases} u \\ p \\ v \end{cases}$, (definidos nos nós).

- Naturais $\begin{cases} \frac{\partial u}{\partial x} \\ \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} \\ \frac{\partial v}{\partial y} \end{cases}$, (definidos na superfície do elemento).

1-3 Equação da continuidade

Como serão considerados casos em regime permanente, propriedades constantes e para casos bidimensionais, a equação da continuidade reduz-se, no plano físico, a:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (1.2)$$

1-4 Equação da Energia

A distribuição de temperatura será calculada pela equação da Energia tratada no plano físico, com as seguintes considerações, Burmeister (1983):

- Caso bidimensional
- Propriedades constantes
- Sem fonte de calor imposta
- Regime permanente
- Convecção forçada

A equação da energia simplificada pode ser escrita da forma:

$$\rho C_p \left(u \frac{\partial}{\partial x} + v \frac{\partial}{\partial y} \right) = k \left(\frac{\partial^2 t}{\partial x^2} + \frac{\partial^2 t}{\partial y^2} \right) + \mu \Phi \quad (1.3)$$

onde

- t - temperatura
- k - condutividade térmica do fluido
- C_p - calor específico do fluido

$$\Phi = 2 \left[\left(\frac{\partial u}{\partial x} \right)^2 + \left(\frac{\partial v}{\partial y} \right)^2 \right] + \left[\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right]^2 \quad (1.4)$$

As condições de contorno, representadas na figura 1.2 que mostra um elemento na fronteira, serão de valor especificado, Dirichlet, em cada nó do elemento, e de derivada, Neumann, aplicada na superfície do elemento.

•Essencial $\{t\}$, (definida nos nós).

•Naturais $\left\{ \begin{array}{l} \partial/\partial x \\ \partial/\partial y \end{array} \right\}$, (definidos na superfície do elemento).

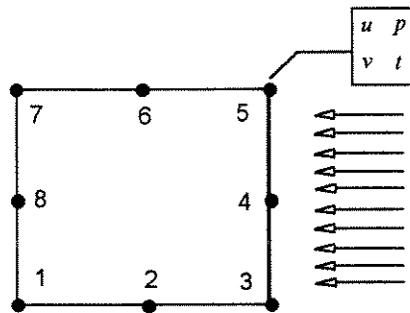


Figura 1.2 : condições de contorno para equações de Navier-Stokes e da Energia

u, p, v, t definidos no nó 5; $\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}, \frac{\partial v}{\partial x}, \frac{\partial v}{\partial y}, \frac{\partial t}{\partial x}, \frac{\partial t}{\partial y}$, definidos na superfície $\bar{35}$.

Capítulo 2

Geração da malha

2-1 Sistemas de coordenadas

O presente trabalho emprega três sistemas de coordenadas para tratamento das equações, e serão mostradas as suas aplicações e relações mútuas.

- Plano xy , também chamado de plano físico.
- Plano $\xi\eta$, também chamado de plano transformado.
- Plano $\lambda\kappa$, também chamado de plano normalizado.

Começando pelo plano físico, este é o plano onde se representa toda a geometria física, dimensões físicas, do problema. É o plano de aplicação das equações de Navier-Stokes e da Energia. Neste plano, xy são eixos de coordenadas.

Considerando uma geometria de um semi-anel circular. A representação deste semi-anel no plano físico é mostrada a seguir, figura 2.1. Este será então dividido em elementos formados pelo cruzamento das linhas $\xi\eta$, formando assim a malha que representará nas equações de Navier-Stokes e da Energia através do método dos elementos finitos aplicado.

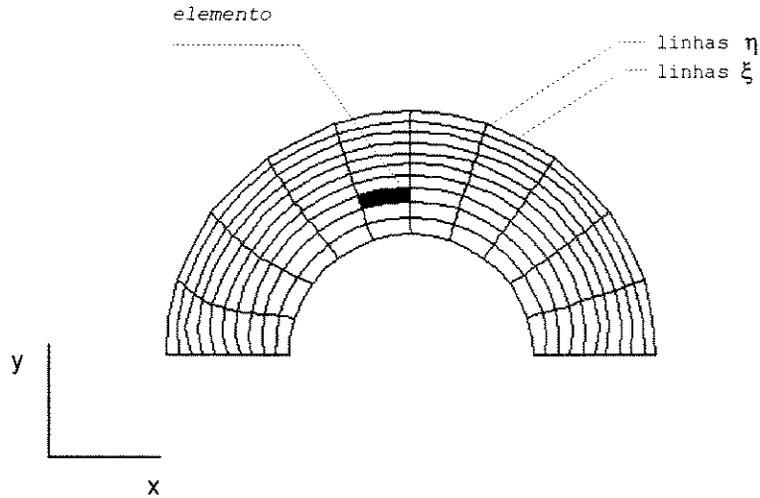


Figura 2.1 : representação do plano físico.

O plano transformado é onde serão aplicadas as equações da geração da malha. O semi-anel do plano físico é representado como um retângulo no plano transformado, figura 2.2. Aqui xy são variáveis, e $\xi\eta$ são eixos de coordenadas, pois o objetivo da geração da malha é encontrar quais os pontos com coordenadas (x,y) pertencem a uma linha constante ξ ou η . Tem-se com isso a representação de geometrias complexas em uma geometria mais simples, onde é conhecida a distribuição das linhas ξ e η .

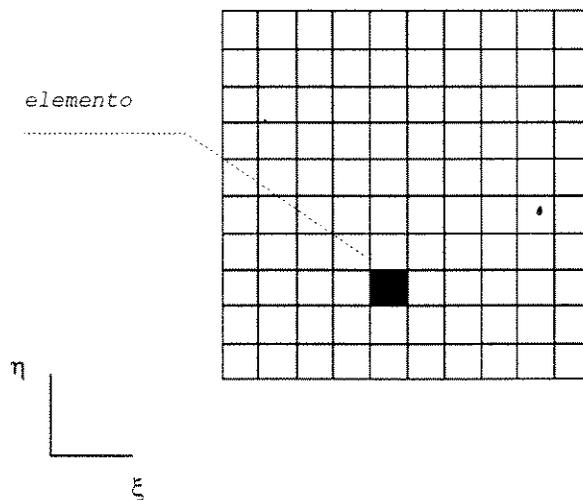


Figura 2.2 : representação do plano transformado

Já o plano normalizado, figura 2.3c, é o plano onde são encontradas as funções de forma para um elemento padrão. Estas funções são relacionadas com xy , quando em Navier-Stokes e Energia, e com $\xi\eta$ quando na geração da malha. Aqui os eixos de coordenadas são $\lambda\kappa$, e suas relações com os eixos xy , de modo semelhante à relação com $\xi\eta$, estão apresentadas na equação 3.15.

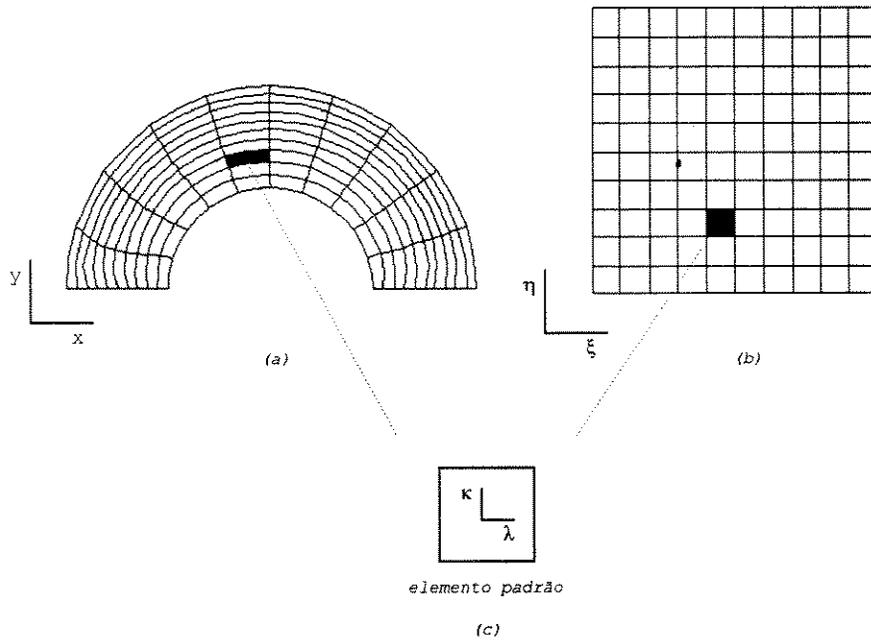


Figura 2.3 : representação do plano normalizado

2-2 Equações geradoras de malha

Como já foi mencionado anteriormente, a disposição da malha utilizada em um problema pode ter grande influência nos resultados finais, portanto a procura de uma malha bem adequada deve ser o objetivo inicial para qualquer solução numérica das equações de Navier-Stokes e da Energia. É importante o fato de que se deve ter malhas refinadas em regiões de altos gradientes: no caso de escoamentos, nas regiões próximas às fronteiras físicas onde u e v

apresentam altos gradientes, ou no encontro de fluxos com direções diferentes; e para estudo térmico, onde há diferenças de temperaturas e fontes geradoras de calor.

Com relação à geração da malha, o domínio em questão pode ser classificado em dois grupos, simples conectados e multiconectados, exemplificados na figura 2.4.

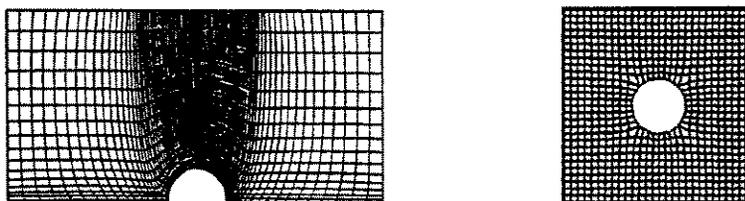


Figura 2.4: domínios simples conectados e multiconectados

Os domínios simples conectados representam regiões fechadas que não contém nenhuma outra sub-região interna, já os domínios multiconectados, representam uma região externa contendo sub-regiões em seu interior e a malha será gerada em um espaço compreendido entre elas, caracterizando geometrias como corpos imersos.

O tratamento numérico destes domínios se dará em um plano transformado, onde qualquer geometria poderá ser representada por uma região geometricamente mais simples que o domínio físico, figura 2.5, seguindo algumas relações de transformação de coordenadas.

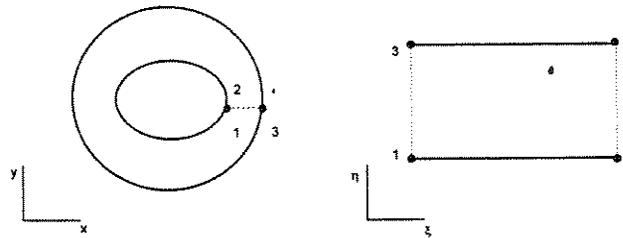


Figura 2.5: plano físico e transformado

Uma malha pode ser gerada a partir de dois métodos básicos mais utilizados: por interpolação algébrica ou por solução de um sistema de equações diferenciais. Existem ainda outros métodos de construção de malha que podem ser adaptados dependendo do caso em particular. Amsden (1973) apresenta um método simples de geração de malha onde uma malha base de fácil construção é transformada em uma outra mais complexa representando o domínio desejado, utilizando relações matemáticas que deformam a malha base, figura 2.6. O método de Amsden (1973) torna-se mais complicado quando a complexidade da geometria cresce apresentando a limitação de ser usado somente em regiões simples conectadas e ainda tem-se o risco de cruzamento de linhas da malha. Truong (1993) utiliza um método de geração de malha não estruturada onde o domínio é coberto por uma malha regular e depois é ajustada às regiões de interesse, figura 2.7. Neste método os elementos formados que estão totalmente dentro da região permanecem intactos, então o ajuste será nos elementos que estarão formando as fronteiras das regiões. Com este método torna-se difícil o controle das linhas da malha caso se desejar ter um ajuste às fronteiras.

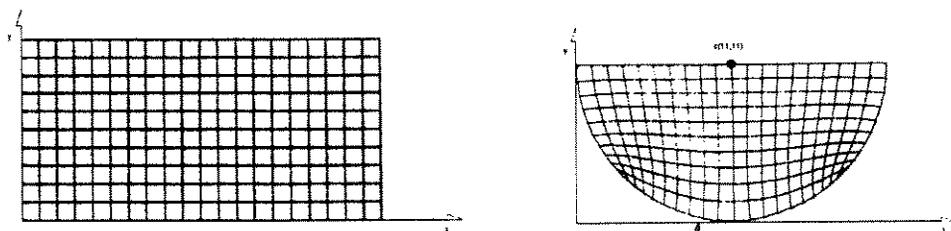


Figura 2.6: malha gerada pelo método de Amsden

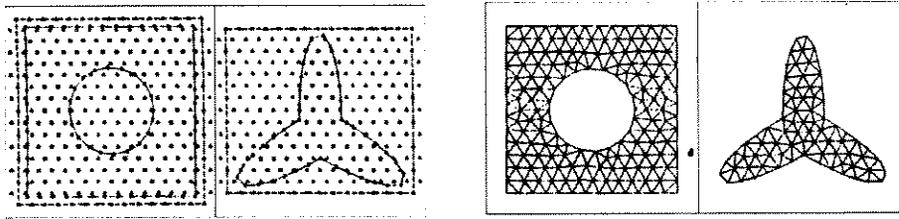


Figura 2.7: malha gerada pelo método de Truong

No método por interpolação algébrica, como o próprio nome diz, faz-se interpolações entre pontos conhecidos do domínio utilizando polinômios interpolantes. Thompson (1985) apresenta diversos polinômios para serem usados na geração de malha por este método.

Finalmente, usando um sistema de equações diferenciais, a malha pode ser gerada como a solução numérica deste sistema. Os tipos de equações usadas podem ser hiperbólicas, parabólicas ou elípticas. A escolha do tipo de equação para geração da malha será dependente somente da geometria do problema: nos sistemas hiperbólicos e parabólicos as fronteiras das regiões não são especificadas por completo caracterizando uma geometria onde a distribuição dos pontos longe da fronteira especificada não é importante, enquanto no sistema elíptico todas as fronteiras são conhecidas. Neste trabalho será usado um sistema elíptico, pois a geometria do problema por mais complexa que seja será conhecida. A escolha por um sistema formado por equações, no caso elípticas, pode ser justificada pelo fato de que com estas equações é conseguido um mapeamento um-para-um entre os planos físico e transformado, implicando o não cruzamento das linhas de coordenadas da malha, e outra importante propriedade é a suavidade inerente a este tipo de equação. Maiores detalhes a respeito das propriedades pode ser encontrados em Thompson (1985) e Maliska (1995).

O sistema elíptico é formado por equações de Poisson que, para o caso bidimensional são, no plano físico :

$$\nabla^2 \xi = P \qquad \nabla^2 \eta = Q \qquad (2.1)$$

Mas o uso deste sistema no plano físico não teria sentido, pois se conhecemos os pontos xy do domínio, conhecemos então a malha, então este sistema é escrito para um plano transformado, onde conhecemos as linhas ξ e η

A forma transformada do sistema foi apresentada primeiramente por Thompson (1974):

$$\begin{aligned} \alpha \frac{\partial^2 x}{\partial \xi^2} - 2\beta \frac{\partial^2 x}{\partial \xi \partial \eta} + \gamma \frac{\partial^2 x}{\partial \eta^2} + J^2 \left(P \frac{\partial x}{\partial \xi} + Q \frac{\partial x}{\partial \eta} \right) &= 0 \\ \alpha \frac{\partial^2 y}{\partial \xi^2} - 2\beta \frac{\partial^2 y}{\partial \xi \partial \eta} + \gamma \frac{\partial^2 y}{\partial \eta^2} + J^2 \left(P \frac{\partial y}{\partial \xi} + Q \frac{\partial y}{\partial \eta} \right) &= 0 \end{aligned} \qquad (2.2)$$

onde

x, y são variáveis dependentes
 P e Q são funções de controle.

$$\begin{aligned} \alpha &= \frac{\partial x^2}{\partial \eta} + \frac{\partial y^2}{\partial \eta} & \beta &= \frac{\partial x}{\partial \xi} \frac{\partial x}{\partial \eta} + \frac{\partial y}{\partial \xi} \frac{\partial y}{\partial \eta} \\ \gamma &= \frac{\partial x^2}{\partial \xi} + \frac{\partial y^2}{\partial \xi} & J &= \frac{\partial x}{\partial \xi} \frac{\partial y}{\partial \eta} - \frac{\partial x}{\partial \eta} \frac{\partial y}{\partial \xi} \end{aligned}$$

Os valores de α , β , γ e J aparecem pelas relações de transformação aplicadas. Um acompanhamento mais detalhado das transformações das equações (2.1) para (2.2) pode ser feito segundo Maliska (1995).

O plano transformado é onde se pode fazer de maneira simplificada a representação de uma geometria qualquer, sendo que as regiões transformadas

podem ter diversas configurações para construção. Configurações diferentes resultaram em malhas diferentes mesmo considerando domínios físicos idênticos, e isto ocorre tanto para regiões simples conectadas como para multi conectadas. É importante frisar que as linhas da malha que saem de um segmento a para um segmento b na região física, também devem seguir o mesmo caminho na região transformada, figura 2.8.

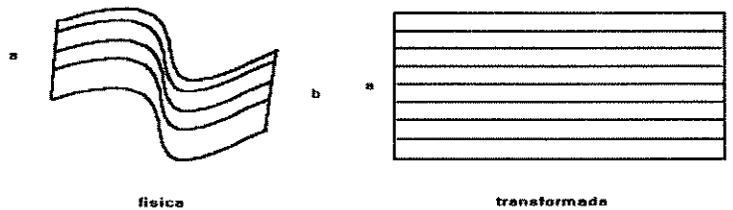


Figura 2.8 : caminho das linhas η

As principais configurações para a região transformada, considerando regiões simples conectadas, estão descritas abaixo :

- Transformação de uma região física qualquer em um retângulo vazio, figura 2.9.

Nesta configuração qualquer região física pode ser transformada em uma região retangular, desde que sejam identificados os quatro vértices (nós) na região física que serão tratados como os quatro vértices do retângulo transformado.

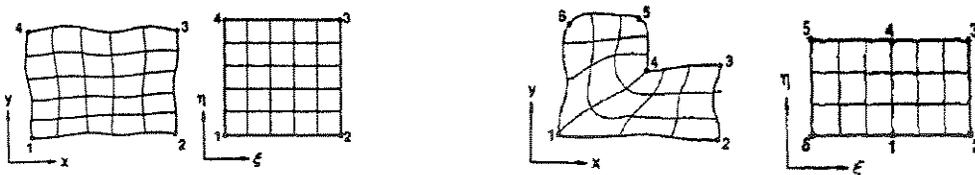


Figura 2.9: região transformada retangular

•Mantendo uma forma parecida com a da região física no plano transformado, figura 2.10.

Nesta configuração a forma original da região física é mantida parcialmente no plano transformado. Deste modo devem ser identificados mais que quatro vértices para formar a região transformada que será definida com segmentos de linha que ligam estes vértices.

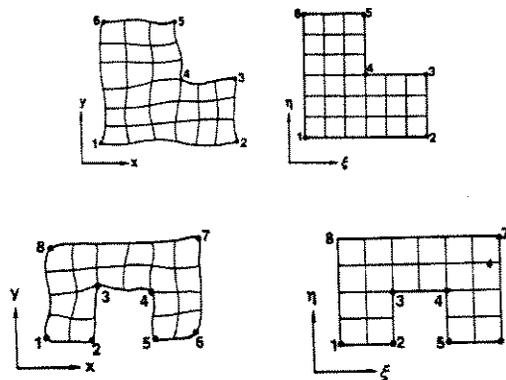


Figura 2.10: região transformada mantendo a forma da região física

Considerando agora regiões multi-conectadas, as principais configurações são :

•Slab

Na configuração slab cada sub-região interna do plano físico é transformada para o plano transformado como regiões retangulares mantendo as suas posições relativas, figura 2.11.

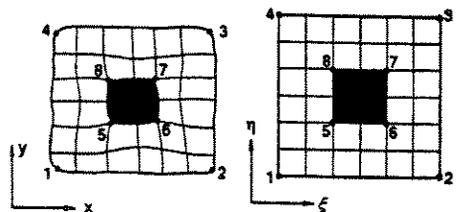


Figura 2.11: região transformada configuração slab

- Slit

Na configuração slit as sub-regiões internas são transformadas em um segmento de linha ξ ou η constantes no plano transformado, caracterizando slit horizontal ou vertical, figura 2.12.

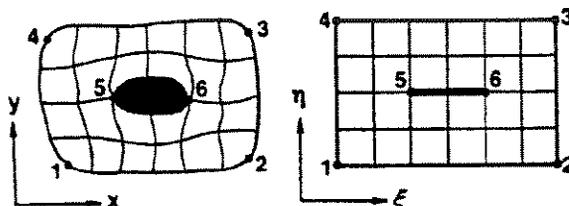


Figura 2.12: região transformada configuração slit

- Tipo O

Na configuração tipo O, todas as regiões físicas juntas são transformadas em uma região retangular no plano transformado. Isto pela introdução de linhas de corte, segmentos 1-2 e 3-4 no plano físico, e para isto devem ser identificados os nós que formarão o corte, figura 2.13.

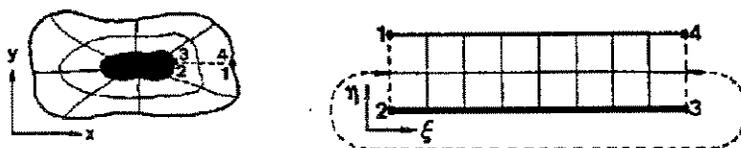


Figura 2.13: região transformada configuração tipo O

- Tipo C

A configuração tipo C é parecida com a tipo O exceto por apresentar alguns nós a mais para a formação da região transformada, e conseqüentemente isto mudará a forma da malha gerada, figura 2.14.

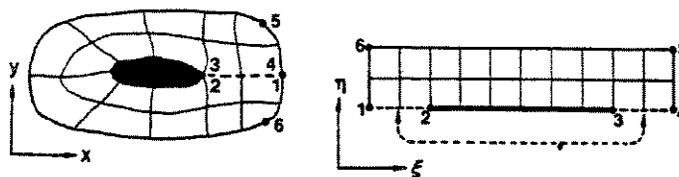


Figura 2.14: região transformada configuração tipo C

Em todas as configurações apresentadas acima, as condições de contorno no plano transformado serão do tipo de valor especificado, Dirichlet, sendo estas os valores das posições (x, y) dos nós na região física.

Capítulo 3

Métodos numéricos

3-1 Apresentação

Foram utilizados neste trabalho dois métodos de aproximação numérica: diferenças finitas e elementos finitos. Uma apresentação geral destes métodos será descrita aqui.

Considerando o método das diferenças finitas, a expressão discretizada da equação diferencial é satisfeita em cada ponto, o domínio é dividido em uma malha ortogonal ou não, regular ou não, e as derivadas em cada ponto são aproximadas por expressões algébricas que podem ser obtidas a partir de uma análise por série de Taylor da função da variável em questão, dentro de uma célula computacional. Considerando o comportamento da função $f(x)$ na figura 3.1,

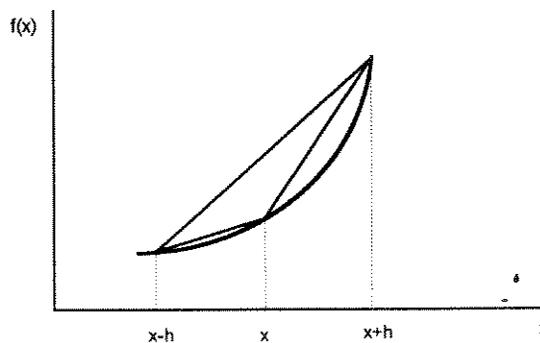


Figura 3.1 : representação gráfica de uma função

por série de Taylor, uma relação do valor desta função suficientemente diferenciável e suas derivadas pode ser da forma, a partir de sua expansão:

$$f(x+h) = f(x) + hf'(x) + \frac{1}{2}h^2 f''(x) + \frac{1}{6}h^3 f'''(x) + \dots$$

$$f(x-h) = f(x) - hf'(x) + \frac{1}{2}h^2 f''(x) - \frac{1}{6}h^3 f'''(x) + \dots$$

Subtraindo ou adicionando estas expansões encontram-se as aproximações para as derivadas das funções. Considerando desprezíveis os termos de ordem h^2 e superiores tem-se as aproximações para as derivadas primeira e segunda da função:

$$f'(x) = \frac{1}{2h} \{f(x+h) - f(x-h)\}$$

$$f''(x) = \frac{1}{h^2} \{f(x+h) - 2f(x) + f(x-h)\}$$

A figura 3.2 apresenta uma célula computacional de nove pontos,

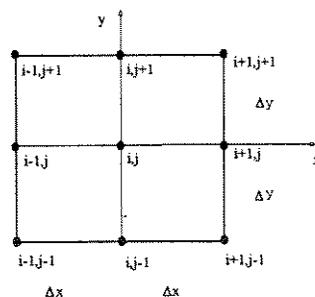


Figura 3.2: célula de nove pontos e derivadas no ponto ij

e aplicando as definições para derivadas de uma variável no ponto ij , tem-se para o caso de malha regular ortogonal, bidimensional:

$$\begin{aligned}\frac{\partial f}{\partial x} &= \frac{1}{2}(f_{i+1,j} - f_{i-1,j}) & \frac{\partial^2 f}{\partial x^2} &= (f_{i+1,j} - 2f_{i,j} + f_{i-1,j}) \\ \frac{\partial f}{\partial y} &= \frac{1}{2}(f_{i,j+1} - f_{i,j-1}) & \frac{\partial^2 f}{\partial y^2} &= (f_{i,j+1} - 2f_{i,j} + f_{i,j-1})\end{aligned}$$

$$\frac{\partial^2 f}{\partial x \partial y} = \frac{1}{4}(f_{i+1,j+1} - f_{i+1,j-1} - f_{i-1,j+1} + f_{i-1,j-1})$$

No método dos elementos finitos a expressão discretizada da equação diferencial é satisfeita de um modo integral em cada elemento que é a célula computacional, parte do domínio, onde os valores das variáveis e suas derivadas são obtidos utilizando funções de interpolação entre os pontos (nós) do elemento.

Para utilização do método dos elementos finitos são necessários alguns passos fundamentais:

- Representar todo o domínio em partes (elementos).
- Em cada elemento, procurar uma aproximação para a solução como uma combinação linear dos valores nodais e funções de forma.
- Derivar as relações algébricas entre os valores nodais da solução em cada parte, e juntar as partes para uma solução do domínio.

3-2 Métodos de aproximação por resíduos ponderados

Antes de ser apresentado o método dos elementos finitos (FEM) que foi utilizado neste trabalho, é de interesse revisar alguns métodos de aproximação dos quais o FEM é derivado.

Formulação forte

Considerando uma equação diferencial ou um sistema de equações diferenciais do tipo $f(u) = p$, onde u representa a solução exata. Uma solução aproximada U da forma escrita em (3.1), pode ser encontrada se satisfizer a todas as condições de contorno essenciais, Dirichlet, e naturais, Neumann, impostas ao problema.

$$U = \phi_0 + \sum_{i=1}^n \theta_i \phi_i \quad (3.1)$$

onde θ_i são coeficientes a serem determinados e ϕ_i são funções de aproximação pré estabelecidas adequadamente tais que as condições de contorno sejam satisfeitas. Se $u = U$, então a solução aproximada é igual à exata, mas se não for, tem-se um erro de aproximação chamado resíduo que é encontrado por $R = f(U) - f(u) \approx 0$, e a integral do produto deste resíduo por uma função ponderadora W é forçada a zero. O objetivo deste procedimento é determinar os coeficientes θ_i da solução aproximada.

$$\int_{\Omega} W \cdot R d\Omega = 0 \quad (3.2 \text{ a})$$

$$R \approx f(U) \approx f(u) \quad (3.2 \text{ b})$$

Vários são os métodos para determinação da função ponderadora W e dos coeficientes θ_i . Alguns dos métodos mais usados são o método dos mínimos quadrados e o método de Galerkin.

Pelo método dos mínimos quadrados as funções ponderadoras são definidas como as derivadas da função resíduo em relação aos coeficientes θ_i :

$$W_1 = \frac{\partial R}{\partial \theta_1} \quad W_2 = \frac{\partial R}{\partial \theta_2} \quad W_n = \frac{\partial R}{\partial \theta_n} \quad (3.3)$$

e os coeficientes θ_i são determinados pelo sistema de n equações e n incógnitas formado pelas integrais:

$$\int_{\Omega} W_1 \cdot R d\Omega = 0 \quad \int_{\Omega} W_2 \cdot R d\Omega = 0 \quad \int_{\Omega} W_n \cdot R d\Omega = 0 \quad (3.4)$$

No método de Galerkin a função ponderadora é feita igual à função pré estabelecida ϕ_i .

$$W_1 = \phi_1 \quad W_2 = \phi_2 \quad W_n = \phi_n \quad (3.5)$$

O sistema será semelhante ao formado por mínimos quadrados para determinação dos coeficientes θ_i , equação (3.4).

Na formulação forte, todas as condições de contorno são aplicadas junto com as aproximações consideradas.

Formulação fraca

Considerando a equação (3.1), a formulação fraca difere da forte na forma como é estabelecida a solução aproximada U . Nesta formulação a solução aproximada para U deve satisfazer pelo menos às condições de contorno essenciais e as funções ponderadoras devem satisfazer às condições de contorno naturais. Os métodos para encontrar as funções ponderadoras são os mesmos já apresentados.

No método dos resíduos ponderados aplica-se as mesmas definições que em (3.3), e no método de Galerkin as mesmas definições que em (3.5), em ambos os métodos o sistema de integrais será montado de forma semelhante ao usado na formulação forte, equação (3.4).

Como nesta formulação a aproximação para U satisfaz somente as condições essenciais, as condições naturais são impostas pela integração por partes da equação resultante semelhante à equação (3.4).

Atentando às condições impostas à solução aproximada, e às funções ponderadoras, nota-se logo que há uma grande dificuldade em construir funções que atendam a todas as condições de contorno. Para casos unidimensionais, ainda tem-se uma facilidade de encontrar funções e soluções que atendam aos requisitos, mas à medida que o problema se torna mais complexo, bidimensional, e o domínio também ganha uma complexidade maior, fica cada vez mais difícil e as vezes impossível encontrar funções para este método.

3-3 Método dos elementos finitos

O método dos elementos finitos, FEM, apresenta uma nova maneira de construção da solução aproximada U , e das funções ponderadoras W , apresentando o conceito de divisão do domínio Ω em sub domínios, elementos, e tratar estes sub domínios separadamente onde seja possível a construção das funções necessárias para uma aproximação por resíduos ponderados utilizando a formulação fraca. O que antes era escrito para todo o domínio, agora é escrito para um elemento, seguindo as mesmas definições:

$$\int_{\Omega^e} W \cdot R d\Omega^e = 0 \quad (3.6)$$

Como o domínio todo foi dividido em sub domínios, a solução para o problema será a somatória das soluções particulares de todos os elementos, respeitando a conectividade de cada um.

Considerando um elemento em separado, e a aproximação proposta em (3.1), mas aplicando uma formulação fraca, tem-se que a aproximação deve satisfazer somente as condições de contorno essenciais.

Então por exemplo uma aproximação de $U(x_0)=U_0$ em $x=x_0$, e $U(x_1)=U_1$ em $x=x_1$, assim por diante. Aplicando a (3.1) :

$$U(x_0) = U_0 = \phi_0(x_0) + \sum_{i=1}^n \theta_i \phi_i(x_0) \quad (3.7)$$

$$U(x_1) = U_1 = \phi_0(x_1) + \sum_{i=1}^n \theta_i \phi_i(x_1)$$

Se todas as condições de contorno essenciais forem homogêneas, segundo definição apresentada por Reddy (1980), valores específicos de ϕ_0 serão zero, então tem-se que:

$$U = \sum_{i=1}^n \theta_i \phi_i \quad (3.8)$$

Alguns elementos típicos são mostrados na figura 3.3, para o caso bidimensional.

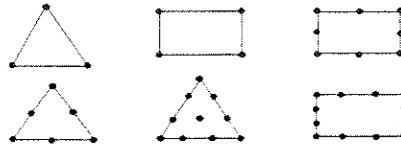


Figura 3.3: elementos típicos bidimensionais

Procurando uma aproximação no elemento que sejam respeitadas as condições de (3.7), é proposta uma nova forma para U , que é escrita como uma aproximação por elementos finitos:

$$U^e = \sum_{i=1}^n U_i^e N_i^e \quad (3.9)$$

onde U^e é o valor de U em um ponto qualquer do elemento, N_i^e é o valor da função de forma, para o elemento e neste ponto, U_i^e é o valor de U em cada nó do elemento e n é o número de nós do elemento.

Seguindo o procedimento de Rayleigh-Ritz, apresentado por Reddy (1993), substitui-se a nova aproximação para U em R , e a função de forma N para W na equação (3.6).

$$\int_{\Omega^e} N \cdot R d\Omega^e = 0 \quad (3.10)$$

lembrando que R é o resíduo da equação diferencial, já definido na equação (3.2b).

As funções de forma que descrevem o comportamento da variável no elemento, são construídas segundo uma interpolação geométrica e devem respeitar algumas condições necessárias para que a solução aproximada seja convergente à solução exata.

- U^e deve ser diferenciável, como requerido na formulação fraca, e todos os termos na forma fraca da equação são representados por valores não nulos.
- Os polinômios usados para representar U^e devem ser completos, ou seja, todos os termos começando do termo constante até o termo de maior ordem devem estar inclusos.
- Todos os termos no polinômio devem ser linearmente independentes.

Os polinômios usados e a forma de construção das funções de forma variam de acordo com a equação discretizada e as condições de contorno. Reddy (1993) apresenta várias funções de forma que são utilizadas para diferentes equações. Para as equações que serão discretizadas aqui, pode-se utilizar um elemento isoparamétrico retangular de quatro ou oito nós, onde a geometria e as variáveis em todo elemento são descritas pela mesmas funções de forma, que podem ser encontradas por polinômios lagrangeanos, obtidos a partir do triângulo de Pascal, figura 3.4.

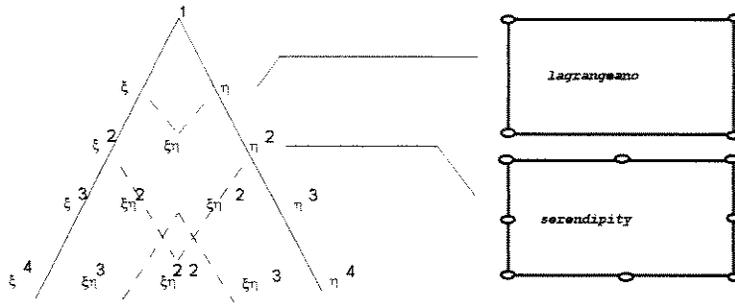


Figura 3.4 : Triângulo de Pascal

Considerando o comportamento no elemento da variável U , onde φ são valores apropriados que serão encontrados posteriormente.

Com referência ao plano transformado, figura 3.5,

$$U = \varphi_1 + \varphi_2 \xi + \varphi_3 \eta + \varphi_4 \xi \eta \tag{3.11}$$

quatro nós

$$U = \varphi_1 + \varphi_2 \xi + \varphi_3 \eta + \varphi_4 \xi \eta + \varphi_5 \xi^2 + \varphi_6 \eta^2 + \varphi_7 \xi \eta^2 + \varphi_8 \xi^2 \eta \tag{3.12}$$

oito nós

Para o elemento de quatro nós, utilizam-se todos os termos do polinômio para encontrar as funções de forma, elementos lagrangeanos; já para o elemento de oito nós, não é utilizado o termo de maior ordem do polinômio lagrangeano considerando que tais termos não contribuem para a conectividade entre elementos, elementos serendipity.

Considerando um elemento retangular de quatro nós, representado na figura 3.5, a expressão (3.11) pode ser escrita na forma matricial, para cada nó do elemento,

$$\begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} 1 & \xi_1 & \eta_1 & \xi_1 \eta_1 \\ 1 & \xi_2 & \eta_2 & \xi_2 \eta_2 \\ 1 & \xi_3 & \eta_3 & \xi_3 \eta_3 \\ 1 & \xi_4 & \eta_4 & \xi_4 \eta_4 \end{bmatrix} \begin{bmatrix} \varphi_1 \\ \varphi_2 \\ \varphi_3 \\ \varphi_4 \end{bmatrix} \quad \text{ou} \quad [U]^e = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = [C][\varphi] = C\varphi \tag{3.13}$$

Reescrevendo a equação (3.13) para avaliar φ tem-se $\varphi = C^{-1}U^e$, e a forma matricial da expressão (3.11) é:

$$U = \begin{bmatrix} 1 & \xi & \eta & \xi\eta \end{bmatrix} \begin{bmatrix} \varphi_1 \\ \varphi_2 \\ \varphi_3 \\ \varphi_4 \end{bmatrix} \quad \text{ou} \quad U = H\varphi \quad (3.14)$$

substituindo φ na expressão (3.14) tem-se $U = HC^{-1}U^e$, onde o termo HC^{-1} é denominado função de forma N , e a expressão para U fica $U = NU^e$

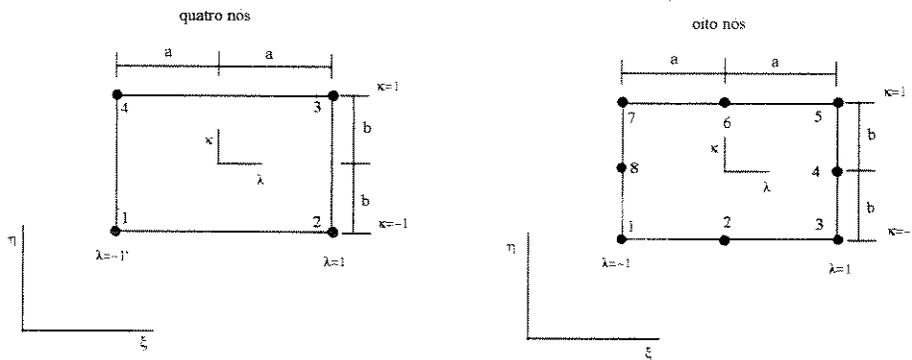


Figura 3.5 : elemento retangular de quatro e oito nós

Todas as operações realizadas podem ser normalizadas a um sistema de coordenadas locais, plano normalizado, cujas relações com o sistema global são dadas por:

$$\begin{aligned} \lambda &= \frac{(\xi - \xi_c)}{a} & d\lambda &= \frac{d\xi}{a} \\ \kappa &= \frac{(\eta - \eta_c)}{b} & d\kappa &= \frac{d\eta}{b} \end{aligned} \quad (3.15)$$

onde (ξ_c, η_c) são as coordenadas do centro do retângulo com dimensões padrões para $a=1$, e $b=1$. Neste elemento, em termos de coordenadas locais, os polinômios são escritos de forma análoga às expressões (3.11) e (3.12), resultando em:

$$U = \varphi_1 + \varphi_2 \lambda + \varphi_3 \kappa + \varphi_4 \lambda \kappa \quad (3.16)$$

quatro nós

$$U = \varphi_1 + \varphi_2 \lambda + \varphi_3 \kappa + \varphi_4 \lambda \kappa + \varphi_5 \lambda^2 + \varphi_6 \kappa^2 + \varphi_7 \lambda \kappa^2 + \varphi_8 \lambda^2 \kappa \quad (3.17)$$

oito nós

Ainda em relação à figura 3.5, tomando as posições de cada nó do elemento no sistema local, explicitam-se as expressões para N para cada nó, substituindo os valores das posições λ e κ a uma expressão análoga a (3.13), só que agora no sistema local, e obtém-se:

$$\begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \begin{bmatrix} \varphi_1 \\ \varphi_2 \\ \varphi_3 \\ \varphi_4 \end{bmatrix} \quad \text{onde } C^{-1} = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 \\ -1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \text{ e}$$

$$H = [1 \quad \lambda \quad \kappa \quad \lambda \kappa]$$

então as expressões para φ são:

$$\begin{aligned} \varphi_1 &= \frac{1}{4}(U_1 + U_2 + U_3 + U_4) & \varphi_3 &= \frac{1}{4}(-U_1 - U_2 + U_3 + U_4) \\ \varphi_2 &= \frac{1}{4}(-U_1 + U_2 + U_3 - U_4) & \varphi_4 &= \frac{1}{4}(U_1 - U_2 + U_3 - U_4) \end{aligned}$$

e as funções de forma são, HC^{-1} :

$$\begin{aligned} N_1 &= \frac{1}{4}(1-\lambda)(1-\kappa) & N_3 &= \frac{1}{4}(1+\lambda)(1+\kappa) \\ N_2 &= \frac{1}{4}(1+\lambda)(1-\kappa) & N_4 &= \frac{1}{4}(1-\lambda)(1+\kappa) \end{aligned}$$

De forma análoga para um elemento de oito nós tem-se :

$$\begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \\ U_6 \\ U_7 \\ U_8 \end{bmatrix} = \begin{bmatrix} 1 & \xi_1 & \eta_1 & \xi_1 \eta_1 & \xi_1^2 & \eta_1^2 & \xi_1 \eta_1^2 & \xi_1^2 \eta_1^2 \\ 1 & \xi_2 & \eta_2 & \xi_2 \eta_2 & \xi_2^2 & \eta_2^2 & \xi_2 \eta_2^2 & \xi_2^2 \eta_2^2 \\ 1 & \xi_3 & \eta_3 & \xi_3 \eta_3 & \xi_3^2 & \eta_3^2 & \xi_3 \eta_3^2 & \xi_3^2 \eta_3^2 \\ 1 & \xi_4 & \eta_4 & \xi_4 \eta_4 & \xi_4^2 & \eta_4^2 & \xi_4 \eta_4^2 & \xi_4^2 \eta_4^2 \\ 1 & \xi_5 & \eta_5 & \xi_5 \eta_5 & \xi_5^2 & \eta_5^2 & \xi_5 \eta_5^2 & \xi_5^2 \eta_5^2 \\ 1 & \xi_6 & \eta_6 & \xi_6 \eta_6 & \xi_6^2 & \eta_6^2 & \xi_6 \eta_6^2 & \xi_6^2 \eta_6^2 \\ 1 & \xi_7 & \eta_7 & \xi_7 \eta_7 & \xi_7^2 & \eta_7^2 & \xi_7 \eta_7^2 & \xi_7^2 \eta_7^2 \\ 1 & \xi_8 & \eta_8 & \xi_8 \eta_8 & \xi_8^2 & \eta_8^2 & \xi_8 \eta_8^2 & \xi_8^2 \eta_8^2 \end{bmatrix} \begin{bmatrix} \varphi_1 \\ \varphi_2 \\ \varphi_3 \\ \varphi_4 \\ \varphi_5 \\ \varphi_6 \\ \varphi_7 \\ \varphi_8 \end{bmatrix} = [U]^r = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \\ U_6 \\ U_7 \\ U_8 \end{bmatrix} = [C][\varphi] = C\varphi$$

Seguindo-se o mesmo procedimento apresentado para elementos de quatro nós, encontra-se para elementos de oito nós as funções de forma:

$$\begin{aligned} N_1 &= \frac{1}{4}(-1 + \lambda\kappa + \lambda\lambda + \kappa\kappa - \lambda\lambda\kappa - \lambda\kappa\kappa) & N_5 &= \frac{1}{4}(-1 + \lambda\kappa + \lambda\lambda + \kappa\kappa + \lambda\lambda\kappa + \lambda\kappa\kappa) \\ N_2 &= \frac{1}{2}(1 - \kappa - \lambda\lambda + \lambda\lambda\kappa) & N_6 &= \frac{1}{2}(1 + \kappa - \lambda\lambda - \lambda\lambda\kappa) \\ N_3 &= \frac{1}{4}(-1 - \lambda\kappa + \lambda\lambda + \kappa\kappa - \lambda\lambda\kappa + \lambda\kappa\kappa) & N_7 &= \frac{1}{4}(-1 - \lambda\kappa + \lambda\lambda + \kappa\kappa + \lambda\lambda\kappa - \lambda\kappa\kappa) \\ N_4 &= \frac{1}{2}(1 + \lambda - \kappa\kappa - \lambda\kappa\kappa) & N_8 &= \frac{1}{2}(1 - \lambda - \kappa\kappa + \lambda\kappa\kappa) \end{aligned} \quad (3.18)$$

Para que seja válida a aproximação para U , as funções de forma devem ter valor 1 no ponto em consideração e zero nos outros pontos. A figura 3.6 representa o comportamento das funções de forma para um elemento de oito nós.

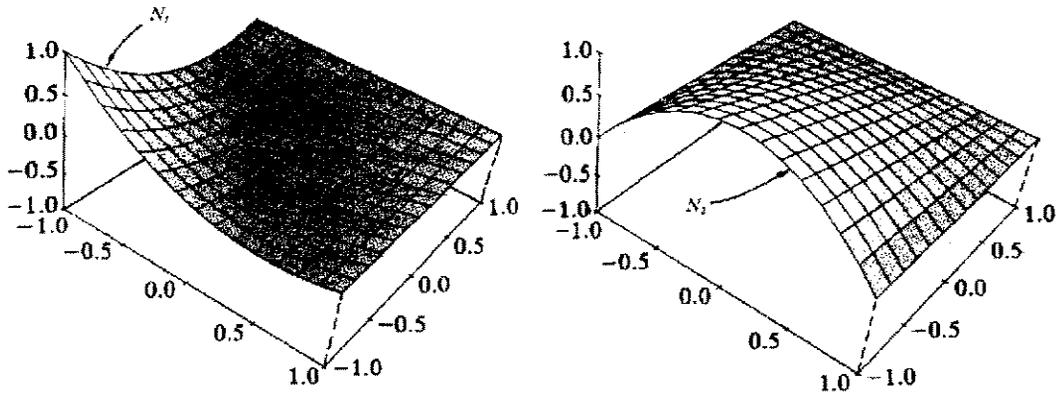


Figura 3.6: comportamento das funções de forma

Tomando a figura 3.5, para o elemento de oito nós e considerando o nó 1 onde $\lambda=-1$ e $\kappa=-1$, a aproximação deve ser:

$$U_1 = \sum_{i=1}^8 U_i^e N_i^e = 1 \times U_1 \quad (3.19)$$

ou seja, $N_1(-1,-1)=1$ e $N_i(\lambda,\kappa)=0 \quad i=2 \rightarrow n$

As derivadas de U são dadas por:

$$\frac{\partial U_1}{\partial \lambda} = \sum_{i=1}^n \frac{\partial N_1^e}{\partial \lambda} U_i^e \quad \text{e} \quad \frac{\partial U_1}{\partial \kappa} = \sum_{i=1}^n \frac{\partial N_1^e}{\partial \kappa} U_i^e \quad (3.20)$$

segundo o procedimento de Rayleigh-Ritz, Reddy (1993).

Conhecidas as funções de forma, pode-se encontrar o valor de uma variável em qualquer lugar do elemento, e portanto em todo o domínio, e está apresentada a aproximação para U , equação (3.9).

Capítulo 4

Tratamento numérico das equações

4-1 Equações geradoras de malha

As equações geradoras de malha serão aproximadas em um plano transformado de configuração simplificada, por elementos quadrangulares de oito nós com dimensões características mostradas na figura 4.1.

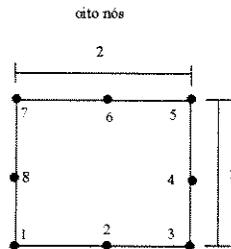


Figura 4.1: elemento característico da malha transformada

Na equação original (2.2), as funções P e Q são funções de controle, e uma das formas para elas, proposta por Thompson (1977), é:

$$\begin{aligned}
 P &= \frac{\partial^2 \xi}{\partial x^2} + \frac{\partial^2 \xi}{\partial y^2} = -\sum_{i=1}^n a_i \operatorname{sgn}(\xi - \xi_i) \exp(-c_i |\xi - \xi_i|) - \sum_{j=1}^m b_j \operatorname{sgn}(\xi - \xi_j) \exp\left(-d_j \left((\xi - \xi_j)^2 + (\eta - \eta_j)^2\right)^{\frac{1}{2}}\right) \\
 Q &= \frac{\partial^2 \eta}{\partial x^2} + \frac{\partial^2 \eta}{\partial y^2} = -\sum_{i=1}^n a_i \operatorname{sgn}(\eta - \eta_i) \exp(-c_i |\eta - \eta_i|) - \sum_{j=1}^m b_j \operatorname{sgn}(\eta - \eta_j) \exp\left(-d_j \left((\xi - \xi_j)^2 + (\eta - \eta_j)^2\right)^{\frac{1}{2}}\right)
 \end{aligned}
 \tag{4.1}$$

ξ_i, η_i identificam as linhas de atração, ξ_j, η_j os pontos de atração, n é o número de linhas que terão efeito atrativo, m é o número de pontos que terão o efeito atrativo, os fatores a, b, c, d são fatores de decaimento do efeito de atração do ponto ou da linha, e são valores escolhidos tais que proporcionarão uma atração adequada, $\operatorname{sgn}(\)$ é uma função de avaliação de sinal das diferenças entre parênteses e $\exp(\)$ é uma função exponencial.

Aplicando resíduos ponderados, considerando o resíduo R igual à equação diferencial original, e a função ponderadora W igual à função de forma N , tem-se, para um elemento:

$$\begin{aligned}
 &\iint_{\Omega} (\text{função de forma}) \times (\text{resíduo}) d\Omega = 0 \\
 &\iint_{\Omega} N \left(\alpha \frac{\partial^2 x}{\partial \xi^2} - 2\beta \frac{\partial^2 x}{\partial \xi \partial \eta} + \gamma \frac{\partial^2 x}{\partial \eta^2} + J^2 \left(P \frac{\partial x}{\partial \xi} + Q \frac{\partial x}{\partial \eta} \right) \right) d\Omega = 0 \\
 &\iint_{\Omega} N \left(\alpha \frac{\partial^2 y}{\partial \xi^2} - 2\beta \frac{\partial^2 y}{\partial \xi \partial \eta} + \gamma \frac{\partial^2 y}{\partial \eta^2} + J^2 \left(P \frac{\partial y}{\partial \xi} + Q \frac{\partial y}{\partial \eta} \right) \right) d\Omega = 0
 \end{aligned}
 \tag{4.2}$$

Considerando uma solução iterativa para o sistema, alguns termos das equações acima, α, β e γ , são tratados de forma especial como valores obtidos de uma iteração anterior à corrente. Para assegurar a característica das equações os termos de segunda ordem são mantidos do lado esquerdo da equações como incógnitas, e o restante é passado para o

lado direito, como valores já calculados. Então cada equação pode ser escrita, em primeira formulação:

$$\begin{aligned} \iint_{\Omega} N \left(\alpha \frac{\partial^2 x}{\partial \xi^2} + \gamma \frac{\partial^2 x}{\partial \eta^2} \right) d\Omega &= - \iint_{\Omega} N \left(-2\beta \frac{\partial^2 x}{\partial \xi \partial \eta} + J^2 \left(P \frac{\partial x}{\partial \xi} + Q \frac{\partial x}{\partial \eta} \right) \right) d\Omega \\ \iint_{\Omega} N \left(\alpha \frac{\partial^2 y}{\partial \xi^2} + \gamma \frac{\partial^2 y}{\partial \eta^2} \right) d\Omega &= - \iint_{\Omega} N \left(-2\beta \frac{\partial^2 y}{\partial \xi \partial \eta} + J^2 \left(P \frac{\partial y}{\partial \xi} + Q \frac{\partial y}{\partial \eta} \right) \right) d\Omega \quad (4.3) \end{aligned}$$

As equações (4.3) foram usadas como uma formulação primária desenvolvida no início do trabalho, mas que foram modificadas mais tarde, e ficaram como:

$$\begin{aligned} \iint_{\Omega} N \left(\alpha \frac{\partial^2 x}{\partial \xi^2} - 2\beta \frac{\partial^2 x}{\partial \xi \partial \eta} + \gamma \frac{\partial^2 x}{\partial \eta^2} \right) d\Omega &= - \iint_{\Omega} N \left(J^2 \left(P \frac{\partial x}{\partial \xi} + Q \frac{\partial x}{\partial \eta} \right) \right) d\Omega \\ \iint_{\Omega} N \left(\alpha \frac{\partial^2 y}{\partial \xi^2} - 2\beta \frac{\partial^2 y}{\partial \xi \partial \eta} + \gamma \frac{\partial^2 y}{\partial \eta^2} \right) d\Omega &= - \iint_{\Omega} N \left(J^2 \left(P \frac{\partial y}{\partial \xi} + Q \frac{\partial y}{\partial \eta} \right) \right) d\Omega \quad (4.4) \end{aligned}$$

Aplicando o teorema de Green às equações, para redução dos termos de segunda ordem nas equações (4.3) e (4.4), tem-se respectivamente:

$$\begin{aligned} - \iint_{\Omega} \frac{\partial N}{\partial \xi} \alpha \frac{\partial x}{\partial \xi} + \frac{\partial N}{\partial \eta} \gamma \frac{\partial x}{\partial \eta} d\Omega - \int_s \left(N \gamma \frac{\partial x}{\partial \eta} \frac{d\xi}{ds} - N \alpha \frac{\partial x}{\partial \xi} \frac{d\eta}{ds} \right) ds &= - \iint_{\Omega} N \left(-2\beta \frac{\partial^2 x}{\partial \xi \partial \eta} + J^2 \left(P \frac{\partial x}{\partial \xi} + Q \frac{\partial x}{\partial \eta} \right) \right) d\Omega \\ - \iint_{\Omega} \frac{\partial N}{\partial \xi} \alpha \frac{\partial y}{\partial \xi} + \frac{\partial N}{\partial \eta} \gamma \frac{\partial y}{\partial \eta} d\Omega - \int_s \left(N \gamma \frac{\partial y}{\partial \eta} \frac{d\xi}{ds} - N \alpha \frac{\partial y}{\partial \xi} \frac{d\eta}{ds} \right) ds &= - \iint_{\Omega} N \left(-2\beta \frac{\partial^2 y}{\partial \xi \partial \eta} + J^2 \left(P \frac{\partial y}{\partial \xi} + Q \frac{\partial y}{\partial \eta} \right) \right) d\Omega \quad (4.5) \end{aligned}$$

$$\begin{aligned}
& -\iint_{\Omega} \frac{\partial N}{\partial \xi} \alpha \frac{\partial N}{\partial \xi} \beta \frac{\partial N}{\partial \eta} \frac{\partial N}{\partial \eta} \gamma \frac{\partial N}{\partial \xi} \frac{\partial N}{\partial \eta} d\Omega - \int_s \left(N\gamma \frac{\partial \alpha}{\partial \eta} \frac{d\xi}{ds} - N\alpha \frac{\partial \alpha}{\partial \xi} \frac{d\eta}{ds} \right) ds + \int_s \left(-N\beta \frac{\partial \alpha}{\partial \xi} \frac{d\xi}{ds} + N\beta \frac{\partial \alpha}{\partial \eta} \frac{d\eta}{ds} \right) ds = -\iint_{\Omega} N \left(J^2 \left(P \frac{\partial \alpha}{\partial \xi} + Q \frac{\partial \alpha}{\partial \eta} \right) \right) d\Omega \\
& -\iint_{\Omega} \frac{\partial N}{\partial \xi} \alpha \frac{\partial N}{\partial \xi} \beta \frac{\partial N}{\partial \eta} \frac{\partial N}{\partial \eta} \gamma \frac{\partial N}{\partial \xi} \frac{\partial N}{\partial \eta} d\Omega - \int_s \left(N\gamma \frac{\partial \alpha}{\partial \eta} \frac{d\xi}{ds} - N\alpha \frac{\partial \alpha}{\partial \xi} \frac{d\eta}{ds} \right) ds + \int_s \left(-N\beta \frac{\partial \alpha}{\partial \xi} \frac{d\xi}{ds} + N\beta \frac{\partial \alpha}{\partial \eta} \frac{d\eta}{ds} \right) ds = -\iint_{\Omega} N \left(J^2 \left(P \frac{\partial \alpha}{\partial \xi} + Q \frac{\partial \alpha}{\partial \eta} \right) \right) d\Omega
\end{aligned}
\tag{4.6}$$

onde os termos de integral de superfície $\int_s () ds$ representam as condições de contorno naturais, Neumann, que para geração da malha não são utilizadas porque é conhecida toda a geometria, então estes termos podem ser retirados das equações, ficando a forma final para o tratamento numérico, respectivamente:

$$\begin{aligned}
\iint \frac{\partial N}{\partial \xi} \alpha \frac{\partial \alpha}{\partial \xi} + \frac{\partial N}{\partial \eta} \gamma \frac{\partial \alpha}{\partial \eta} d\xi d\eta &= \iint N \left(-2\beta \frac{\partial^2 \alpha}{\partial \xi \partial \eta} + J^2 \left(P \frac{\partial \alpha}{\partial \xi} + Q \frac{\partial \alpha}{\partial \eta} \right) \right) d\xi d\eta \\
\iint \frac{\partial N}{\partial \xi} \alpha \frac{\partial \alpha}{\partial \xi} + \frac{\partial N}{\partial \eta} \gamma \frac{\partial \alpha}{\partial \eta} d\xi d\eta &= \iint N \left(-2\beta \frac{\partial^2 \alpha}{\partial \xi \partial \eta} + J^2 \left(P \frac{\partial \alpha}{\partial \xi} + Q \frac{\partial \alpha}{\partial \eta} \right) \right) d\xi d\eta \quad (4.7)
\end{aligned}$$

$$\begin{aligned}
\iint \frac{\partial N}{\partial \xi} \alpha \frac{\partial \alpha}{\partial \xi} - \frac{\partial N}{\partial \xi} \beta \frac{\partial \alpha}{\partial \eta} - \frac{\partial N}{\partial \eta} \beta \frac{\partial \alpha}{\partial \xi} + \frac{\partial N}{\partial \eta} \gamma \frac{\partial \alpha}{\partial \eta} d\xi d\eta &= \iint N \left(J^2 \left(P \frac{\partial \alpha}{\partial \xi} + Q \frac{\partial \alpha}{\partial \eta} \right) \right) d\xi d\eta \\
\iint \frac{\partial N}{\partial \xi} \alpha \frac{\partial \alpha}{\partial \xi} - \frac{\partial N}{\partial \xi} \beta \frac{\partial \alpha}{\partial \eta} - \frac{\partial N}{\partial \eta} \beta \frac{\partial \alpha}{\partial \xi} + \frac{\partial N}{\partial \eta} \gamma \frac{\partial \alpha}{\partial \eta} d\xi d\eta &= \iint N \left(J^2 \left(P \frac{\partial \alpha}{\partial \xi} + Q \frac{\partial \alpha}{\partial \eta} \right) \right) d\xi d\eta \quad (4.8)
\end{aligned}$$

Os termos α , γ e β presentes no lado esquerdo das equações também são considerados como valores já calculados numa iteração anterior.

Lembrando que as equações (4.7) e (4.8) são escritas para um elemento, a aproximação para todo o domínio será a somatória das aproximações de todos os elementos do domínio. Deste modo então, para um domínio formado por n elementos é dada:

$$\text{Domínio} = \sum_{i=1}^n \iint_{\Omega} d\Omega$$

Então são encontradas as equações (4.7) e (4.8), que deverão ser resolvidas no plano transformado, figura 4.2, para obtermos a malha desejada.

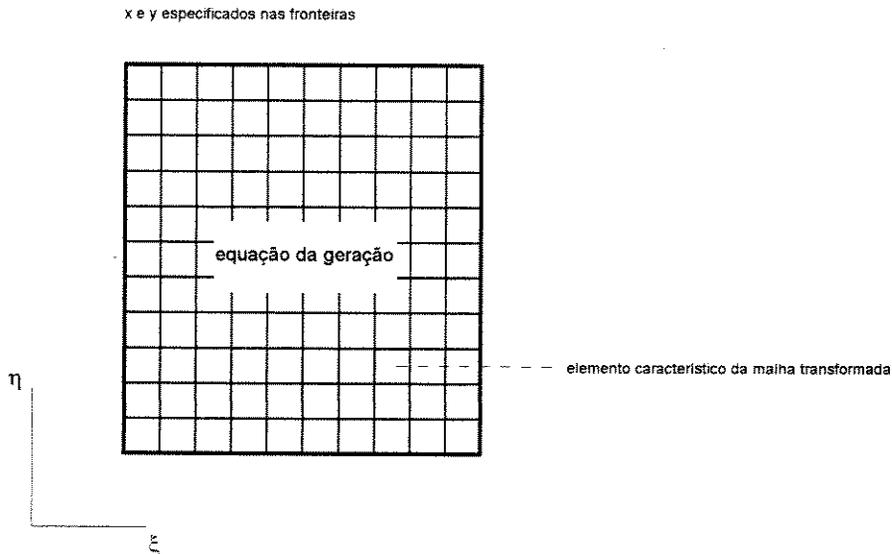


Figura 4.2: plano transformado, elemento característico

Até agora foi desenvolvida a preparação analítica das equações para geração da malha, mas para aplicação numérica, programação computacional, tem-se que ainda introduzir o conceito de integração numérica aplicada. A integração das equações (4.5) é tratada numericamente considerando-se um elemento padrão, figura 4.3, e utilizando o método da quadratura de Gauss aplicado a ele. Para que seja

aplicada a quadratura de Gauss, as equações devem ser normalizadas ao elemento padrão.

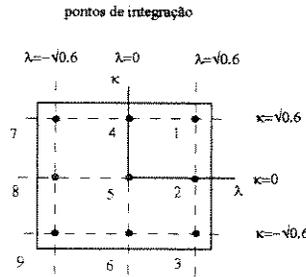


Figura 4.3: pontos de integração numérica, pontos de Gauss

Considerando a integral de uma função $F(\xi, \eta)$, como exemplo,

$$\iint_{\Omega} F(\xi, \eta) d\Omega = \int_{\xi_a}^{\xi_b} \int_{\eta_a}^{\eta_b} J * F(\xi, \eta) d\xi d\eta$$

sendo J o jacobiano da transformação, que normalizada ao elemento fica:

$$\int_{\xi_a}^{\xi_b} \int_{\eta_a}^{\eta_b} F(\xi, \eta) d\xi d\eta = \int_{-1}^1 \int_{-1}^1 F(\lambda, \kappa) d\lambda d\kappa$$

e a forma numérica desta integral referente ao elemento considerado será:

$$\int_{-1}^1 \int_{-1}^1 F(\lambda, \kappa) d\lambda d\kappa = \sum_{i=1}^3 \sum_{j=1}^3 \varpi_i \varpi_j F(\lambda_i, \kappa_j)$$

onde ϖ_i, ϖ_j são os valores ponderadores relativos à posição dos pontos de integração, que no caso são 9 pontos, como mostrado e discutido por Reddy (1993), tabela 4.1, $F(\lambda_i, \kappa_j)$ é o valor da função no ponto de integração.

Fatores ponderadores e posição dos pontos de Gauss

ponto	i	j	w_i	w_j	λ	κ
1	1	1	5/9	5/9	$\sqrt{0.6}$	$\sqrt{0.6}$
2	1	2	5/9	8/9	$\sqrt{0.6}$	0
3	1	3	5/9	5/9	$\sqrt{0.6}$	$-\sqrt{0.6}$
4	2	1	8/9	5/9	0	$\sqrt{0.6}$
5	2	2	8/9	8/9	0	0
6	2	3	8/9	5/9	0	$-\sqrt{0.6}$
7	3	1	5/9	5/9	$-\sqrt{0.6}$	$\sqrt{0.6}$
8	3	2	5/9	8/9	$-\sqrt{0.6}$	0
9	3	3	5/9	5/9	$-\sqrt{0.6}$	$-\sqrt{0.6}$

Tabela 4.1: fatores ponderadores e posições dos pontos de Gauss

Então de uma forma simplificada, as integrais para geração da malha podem ser escritas da forma, para o elemento padrão:

$$\int_{-1}^1 \int_{-1}^1 (\text{função de forma}) \times (\text{resíduo}) d\lambda d\kappa = 0 \quad (4.9)$$

Como o tratamento da integração numérica será relativo ao elemento normalizado, as derivadas devem ser expressas em relação ao sistema global de coordenadas, plano transformado, já que as funções de forma foram descritas em relação ao sistema local, plano normalizado, e as equações da geração da malha descritas para o sistema global.

Seguindo as relações entre os sistemas global e local de coordenadas, equações (3.15), tem-se o jacobiano da transformação, para a mudança de variável nas integrais de área, Kreyszig (1993).

$$[\mathbf{J}] = \begin{bmatrix} \frac{\partial \xi}{\partial \lambda} & \frac{\partial \eta}{\partial \lambda} \\ \frac{\partial \xi}{\partial \kappa} & \frac{\partial \eta}{\partial \kappa} \end{bmatrix}$$

Aplicando a regra da cadeia tem-se para as derivadas parciais:

$$\begin{aligned}\frac{\partial N}{\partial \lambda} &= \frac{\partial N}{\partial \xi} \frac{\partial \xi}{\partial \lambda} + \frac{\partial N}{\partial \eta} \frac{\partial \eta}{\partial \lambda} \\ \frac{\partial N}{\partial \kappa} &= \frac{\partial N}{\partial \xi} \frac{\partial \xi}{\partial \kappa} + \frac{\partial N}{\partial \eta} \frac{\partial \eta}{\partial \kappa}\end{aligned}$$

na forma matricial,

$$\begin{bmatrix} \frac{\partial N}{\partial \lambda} \\ \frac{\partial N}{\partial \kappa} \end{bmatrix} = \begin{bmatrix} \frac{\partial \xi}{\partial \lambda} & \frac{\partial \eta}{\partial \lambda} \\ \frac{\partial \xi}{\partial \kappa} & \frac{\partial \eta}{\partial \kappa} \end{bmatrix} \begin{bmatrix} \frac{\partial N}{\partial \xi} \\ \frac{\partial N}{\partial \eta} \end{bmatrix}$$

que resulta,

$$\begin{bmatrix} \frac{\partial N}{\partial \xi} \\ \frac{\partial N}{\partial \eta} \end{bmatrix} = [\mathbf{J}]^{-1} \begin{bmatrix} \frac{\partial N}{\partial \lambda} \\ \frac{\partial N}{\partial \kappa} \end{bmatrix} \quad \text{e} \quad \begin{aligned} \frac{\partial N}{\partial \xi} &= \hat{\mathbf{J}}_{11} \frac{\partial N}{\partial \lambda} + \hat{\mathbf{J}}_{12} \frac{\partial N}{\partial \kappa} \\ \frac{\partial N}{\partial \eta} &= \hat{\mathbf{J}}_{21} \frac{\partial N}{\partial \lambda} + \hat{\mathbf{J}}_{22} \frac{\partial N}{\partial \kappa} \end{aligned}$$

sendo $[\mathbf{J}]$ a matriz jacobiana da transformação e $\hat{\mathbf{J}}_{ij}$ são os coeficientes da matriz jacobiana inversa.

$$[\mathbf{J}]^{-1} = \frac{1}{\begin{pmatrix} \frac{\partial \xi}{\partial \lambda} \frac{\partial \eta}{\partial \kappa} - \frac{\partial \eta}{\partial \lambda} \frac{\partial \xi}{\partial \kappa} \end{pmatrix}} \times \begin{bmatrix} \frac{\partial \eta}{\partial \kappa} & -\frac{\partial \eta}{\partial \lambda} \\ -\frac{\partial \xi}{\partial \kappa} & \frac{\partial \xi}{\partial \lambda} \end{bmatrix}$$

Pela mudança de variáveis tem-se que,

$$d\xi d\eta = \mathbf{J} d\lambda d\kappa$$

Então as integrais de área ficam:

$$\iint d\xi d\eta = \int_{-1}^1 \int_{-1}^1 \mathbf{J} d\lambda d\kappa$$

Cada termo aqui encontrado deve ser substituído na equação (4.1), para a integração numérica no elemento padrão.

Considerando ainda as equações (4.7) e (4.8), e substituindo as aproximações para as derivadas das variáveis, equações (3.19) e (3.20), tem-se respectivamente:

$$\begin{aligned} \iint \frac{\partial N}{\partial \xi} \alpha \sum_1^8 \frac{\partial N}{\partial \xi} x + \frac{\partial N}{\partial \eta} \gamma \sum_1^8 \frac{\partial N}{\partial \eta} x d\xi d\eta &= \iint N \left(-2\beta \frac{\partial^2 x}{\partial \xi \partial \eta} + J^2 \left(P \sum_1^8 \frac{\partial N}{\partial \xi} x + Q \sum_1^8 \frac{\partial N}{\partial \eta} x \right) \right) d\xi d\eta \\ \iint \frac{\partial N}{\partial \xi} \alpha \sum_1^8 \frac{\partial N}{\partial \xi} y + \frac{\partial N}{\partial \eta} \gamma \sum_1^8 \frac{\partial N}{\partial \eta} y d\xi d\eta &= \iint N \left(-2\beta \frac{\partial^2 y}{\partial \xi \partial \eta} + J^2 \left(P \sum_1^8 \frac{\partial N}{\partial \xi} y + Q \sum_1^8 \frac{\partial N}{\partial \eta} y \right) \right) d\xi d\eta \end{aligned} \quad (4.10)$$

$$\begin{aligned} \iint \frac{\partial N}{\partial \xi} \alpha \sum_1^8 \frac{\partial N}{\partial \xi} x - \frac{\partial N}{\partial \xi} \beta \sum_1^8 \frac{\partial N}{\partial \eta} x - \frac{\partial N}{\partial \eta} \beta \sum_1^8 \frac{\partial N}{\partial \xi} x + \frac{\partial N}{\partial \eta} \gamma \sum_1^8 \frac{\partial N}{\partial \eta} x d\xi d\eta &= \iint N \left(J^2 \left(P \sum_1^8 \frac{\partial N}{\partial \xi} x + Q \sum_1^8 \frac{\partial N}{\partial \eta} x \right) \right) d\xi d\eta \\ \iint \frac{\partial N}{\partial \xi} \alpha \sum_1^8 \frac{\partial N}{\partial \xi} y - \frac{\partial N}{\partial \xi} \beta \sum_1^8 \frac{\partial N}{\partial \eta} y - \frac{\partial N}{\partial \eta} \beta \sum_1^8 \frac{\partial N}{\partial \xi} y + \frac{\partial N}{\partial \eta} \gamma \sum_1^8 \frac{\partial N}{\partial \eta} y d\xi d\eta &= \iint N \left(J^2 \left(P \sum_1^8 \frac{\partial N}{\partial \xi} y + Q \sum_1^8 \frac{\partial N}{\partial \eta} y \right) \right) d\xi d\eta \end{aligned} \quad (4.11)$$

Das equações (4.10) e (4.11) define-se a matriz do elemento como a junção das matrizes das duas equações,

$$\begin{bmatrix}
 \alpha_{11}^x & 0 & \alpha_{12}^x & 0 & \dots & \dots & \dots & 0 \\
 0 & \alpha_{11}^y & 0 & & & & & \vdots \\
 \alpha_{21}^x & 0 & & & & & & \vdots \\
 0 & & & & & & & \vdots \\
 \vdots & & & & & & & 0 \\
 \vdots & & & & & & 0 & \alpha_{78}^y \\
 \vdots & & & & & 0 & \alpha_{88}^x & 0 \\
 0 & \dots & \dots & \dots & 0 & \alpha_{87}^y & 0 & \alpha_{88}^y
 \end{bmatrix}
 \begin{bmatrix}
 x_1 \\
 y_1 \\
 x_2 \\
 \vdots \\
 \vdots \\
 y_7 \\
 x_8 \\
 y_8
 \end{bmatrix}
 =
 \begin{bmatrix}
 \Phi_1 \\
 \Theta_1 \\
 \Phi_2 \\
 \vdots \\
 \vdots \\
 \Theta_7 \\
 \Phi_8 \\
 \Theta_8
 \end{bmatrix}$$

(4.12)

onde $\alpha_{ij}^{x,y} = \iint \left(\frac{\partial N_i}{\partial \xi} \alpha_i \frac{\partial N_j}{\partial \xi} + \frac{\partial N_i}{\partial \eta} \gamma_i \frac{\partial N_j}{\partial \eta} \right) d\xi d\eta$ para a equação (4.10), e $\alpha_{ij}^{x,y} = \iint \left(\frac{\partial N_i}{\partial \xi} \alpha_i \frac{\partial N_j}{\partial \xi} - \frac{\partial N_i}{\partial \xi} \beta \frac{\partial N_j}{\partial \eta} - \frac{\partial N_i}{\partial \eta} \beta \frac{\partial N_j}{\partial \xi} + \frac{\partial N_i}{\partial \eta} \gamma_i \frac{\partial N_j}{\partial \eta} \right) d\xi d\eta$ para a equação (4.11), x e y superescrito indicam a qual equação, para a variável x ou y , o coeficiente está relacionado, Φ e Θ representam o lado direito das equações.

No lado direito das equações (4.7) todos os termos podem ser aproximados pelas equações (3.20), a não ser os termos de derivadas cruzadas, que são tratados de maneira especial, descrita a seguir.

Como já foi mostrado, as integrais são calculadas como somatórias dos valores ponderados das funções nos pontos de Gauss no elemento padrão, portanto as derivadas cruzadas devem ser também calculadas nos pontos de Gauss no elemento padrão.

O método das diferenças finitas é aplicado a cada ponto de Gauss para aproximar as derivadas cruzadas.

Analisando detalhadamente o elemento padrão, pode-se dividi-lo em sub-regiões tais que seja possível a aproximação pelo esquema central de diferenças finitas, figura 4.4,

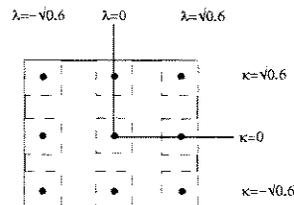


Figura 4.4: sub-regiões do elemento padrão

Cada ponto de Gauss está rodeado por quatro pontos auxiliares, formando assim uma célula computacional onde pode ser aplicado o método das diferenças finitas. A célula formada por estes pontos tem dimensões definidas e constantes, figura 4.5.

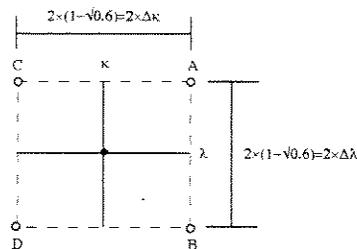


Figura 4.5: célula em torno do ponto de Gauss

Para que seja possível a aproximação por diferenças finitas, é necessário que os valores das variáveis nos pontos A,B,C,D sejam conhecidos. Estes valores podem ser calculados pelas equações (3.19), conhecendo-se as posições dos pontos, tabela 4.2.

Posição da célula para pontos de Gauss

Ponto	A(λ)	A(κ)	B(λ)	B(κ)	C(λ)	C(κ)	D(λ)	D(κ)
1	1	1	1	0.5491	0.5491	1	0.5491	0.5491
2	1	0.2254	1	-0.2254	0.5491	0.2254	0.5491	-0.2254
3	1	-0.5491	1	-1	0.5491	-0.5491	0.5491	-1
4	0.2254	1	0.2254	0.5491	-0.2254	1	-0.2254	0.5491
5	0.2254	0.2254	0.2254	-0.2254	-0.2254	0.2254	-0.2254	-0.2254
6	0.2254	-0.5491	0.2254	-1	-0.2254	-0.5491	-0.2254	-1
7	-0.5491	1	-0.5491	0.5491	-1	1	-1	0.5491
8	-0.5491	0.2254	-0.5491	-0.2254	-1	0.2254	-1	-0.2254
9	-0.5491	-0.5491	-0.5491	-1	-1	-0.5491	-1	-1

Tabela 4.2: posições da célula para pontos de Gauss

A derivada cruzada é calculada pelas equações,

$$\begin{aligned} \frac{\partial x}{\partial \lambda \partial \kappa} &= \frac{\partial x}{\partial \xi \partial \eta} = \frac{1}{4\Delta\lambda\Delta\kappa} (x_A - x_C - x_B + x_D) \\ \frac{\partial y}{\partial \lambda \partial \kappa} &= \frac{\partial y}{\partial \xi \partial \eta} = \frac{1}{4\Delta\lambda\Delta\kappa} (y_A - y_C - y_B + y_D) \end{aligned} \quad (4.13)$$

A igualdade acima é válida já que os elementos da malha transformada são semelhantes ao elemento padrão, e os sistemas de coordenadas estão orientados da mesma forma.

A numeração das variáveis na equação (4.12) está de acordo com a numeração local de cada nó, que deve ser trocada pela numeração global dos nós do elemento quando for realizada a formação da matriz global que representará todo o problema, assim para elementos que compartilham o mesmo nó, os coeficientes para este nó serão a soma das contribuições de cada elemento.

Para exemplificar de maneira simplificada a formação da matriz global, consideramos uma equação de Laplace (4.14) onde há uma variável por nó e considerando dois elementos em uma malha, elementos **a** e **b**.

$$\frac{\partial^2 \zeta}{\partial \xi^2} + \frac{\partial^2 \zeta}{\partial \eta^2} = 0 \tag{4.14}$$

Após todo o procedimento aplicado para elementos finitos, a equação (4.11) fica, sem condições naturais:

$$\iint \frac{\partial N}{\partial \xi} \frac{\partial \zeta}{\partial \xi} + \frac{\partial N}{\partial \eta} \frac{\partial \zeta}{\partial \eta} d\xi d\eta = 0 \tag{4.15}$$

A numeração local de cada nó está representada dentro do elemento, e a numeração global está representada pelos círculos, figura 4.6.

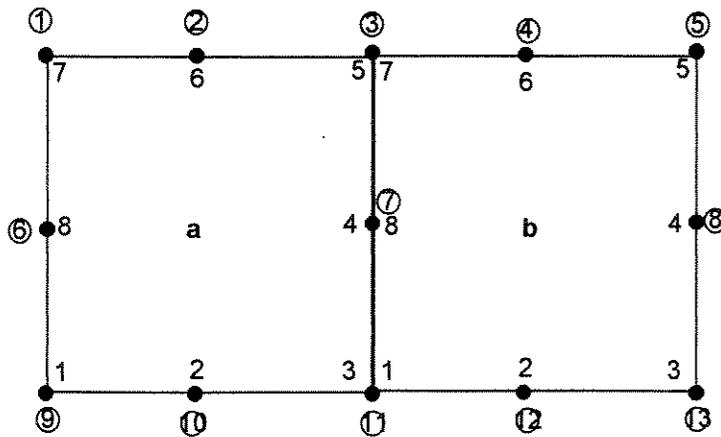


Figura 4.6: numeração local e global dos nós

Considerando cada elemento em separado, para o elemento **a**, temos a matriz representativa, como já foi demonstrado anteriormente, com as numerações locais e globais de cada nó.

$$\begin{array}{c}
 \begin{array}{c}
 \text{global} \\
 \text{local}
 \end{array}
 \begin{array}{c}
 \text{global} \\
 \text{local}
 \end{array}
 \begin{array}{c}
 9 \\
 10 \\
 11 \\
 7 \\
 3 \\
 2 \\
 1 \\
 6
 \end{array} \\
 \begin{array}{c}
 1 \\
 2 \\
 3 \\
 4 \\
 5 \\
 6 \\
 7 \\
 8
 \end{array}
 \end{array}
 \left[\begin{array}{cccccccc}
 a_{11} & a_{12} & a_{13} & a_{14} & a_{15} & a_{16} & a_{17} & a_{18} \\
 & a_{22} & a_{23} & a_{24} & a_{25} & a_{26} & a_{27} & a_{28} \\
 & & a_{33} & a_{34} & a_{35} & a_{36} & a_{37} & a_{38} \\
 & & & a_{44} & a_{45} & a_{46} & a_{47} & a_{48} \\
 & & & & a_{55} & a_{56} & a_{57} & a_{58} \\
 & & & & & a_{66} & a_{67} & a_{68} \\
 & & & & & & a_{77} & a_{78} \\
 & & & & & & & a_{88}
 \end{array} \right]$$

De forma semelhante para o elemento **b**, tem-se a matriz

$$\begin{array}{c}
 \begin{array}{c}
 \text{global} \\
 \text{local}
 \end{array}
 \begin{array}{c}
 \text{global} \\
 \text{local}
 \end{array}
 \begin{array}{c}
 11 \\
 12 \\
 13 \\
 8 \\
 5 \\
 4 \\
 3 \\
 7
 \end{array} \\
 \begin{array}{c}
 1 \\
 2 \\
 3 \\
 4 \\
 5 \\
 6 \\
 7 \\
 8
 \end{array}
 \end{array}
 \left[\begin{array}{cccccccc}
 b_{11} & b_{12} & b_{13} & b_{14} & b_{15} & b_{16} & b_{17} & b_{18} \\
 & b_{22} & b_{23} & b_{24} & b_{25} & b_{26} & b_{27} & b_{28} \\
 & & b_{33} & b_{34} & b_{35} & b_{36} & b_{37} & b_{38} \\
 & & & b_{44} & b_{45} & b_{46} & b_{47} & b_{48} \\
 & & & & b_{55} & b_{56} & b_{57} & b_{58} \\
 & & & & & b_{66} & b_{67} & b_{68} \\
 & & & & & & b_{77} & b_{78} \\
 & & & & & & & b_{88}
 \end{array} \right]$$

A matriz global será formada com a junção das matrizes do elemento **a** e **b**, seguindo a numeração global:

global	1	2	3	4	5	6	7	8	9	10	11	12	13
1	a_{77}	a_{67}	a_{57}			a_{78}	a_{47}		a_{17}	a_{27}	a_{37}		
2		a_{66}	a_{56}			a_{68}	a_{46}		a_{16}	a_{26}	a_{36}		
3			$a_{55}+b_{77}$	b_{67}	b_{57}	a_{58}	$a_{45}+b_{78}$	b_{47}	a_{15}	a_{25}	$a_{35}+b_{17}$	b_{27}	b_{37}
4				b_{66}	b_{56}		b_{68}	b_{46}			b_{16}	b_{26}	b_{36}
5					b_{55}		b_{58}	b_{45}			b_{15}	b_{25}	b_{35}
6						a_{88}	a_{48}		a_{18}	a_{28}	a_{38}		
7							$a_{44}+b_{88}$	b_{48}	a_{14}	a_{24}	$a_{34}+b_{18}$	b_{28}	b_{38}
8								b_{44}			b_{14}	b_{24}	b_{34}
9			<i>simetrica</i>						a_{11}	a_{12}	a_{13}		
10										a_{22}	a_{23}		
11											$a_{33}+b_{11}$	b_{12}	b_{13}
12												b_{22}	
13													b_{33}

Seguindo o mesmo princípio, o lado direito das equações é formado.

$$\begin{bmatrix} \Phi_7^a \\ \Phi_6^a \\ \Phi_5^a + \Phi_7^b \\ \Phi_6^b \\ \Phi_5^b \\ \Phi_8^a \\ \Phi_4^a + \Phi_8^b \\ \Phi_4^b \\ \Phi_1^a \\ \Phi_2^a \\ \Phi_3^a + \Phi_1^b \\ \Phi_2^b \\ \Phi_3^b \end{bmatrix}$$

4-2 Equações de Navier-Stokes e da Continuidade

Toda a discretização das equações de Navier-Stokes foi desenvolvida por Taylor (1981), e aqui está resumidamente apresentada.

Até aqui, x e y foram tratados como variáveis, no sistema usado para geração da malha. Agora serão tratados como valores já conhecidos, que representam as coordenadas dos pontos formadores da malha. As variáveis a serem encontradas são as componentes u e v e a pressão p em cada ponto da malha.

As equações de Navier-Stokes que serão utilizadas e a equação da Continuidade foram escritas no capítulo 1, equações (1.1) e (1.2)

Elas estão escritas em uma forma dimensional, para um sistema de coordenadas globais, plano físico (x,y) . Para o tratamento numérico por elementos finitos, é necessário que as equações tratadas sejam escritas na forma requerida pela formulação fraca, apresentada no capítulo 2.

O tipo de elemento que será usado para o tratamento das equações de Navier-Stokes será o mesmo utilizado para geração da malha, elemento quadrangular de oito nós para velocidades, com o adicional elemento de quatro nós para os pontos da pressão.

Aplicando aqui todos os conceitos já apresentados, tem-se as aproximações seguintes:

$$u = \sum_{i=1}^8 N_i u_i \quad v = \sum_{i=1}^8 N_i v_i \quad p = \sum_{i=1}^4 M_i p_i$$

$$\frac{\partial u}{\partial x} = \sum_{i=1}^8 \frac{\partial N}{\partial x} u_i \quad \frac{\partial u}{\partial y} = \sum_{i=1}^8 \frac{\partial N}{\partial y} u_i$$

$$\begin{aligned} \frac{\partial v}{\partial x} &= \sum_{i=1}^8 \frac{\partial N}{\partial x_i} v_i & \frac{\partial v}{\partial y} &= \sum_{i=1}^8 \frac{\partial N}{\partial y_i} v_i \\ \frac{\partial p}{\partial x} &= \sum_{i=1}^4 \frac{\partial M}{\partial x_i} p_i & \frac{\partial p}{\partial y} &= \sum_{i=1}^4 \frac{\partial M}{\partial y_i} p_i \end{aligned} \quad (4.16)$$

onde u , v e p são os valores das velocidades e pressão em um ponto qualquer do elemento, e os índices indicam os valores das variáveis nos nós do elemento, N e M são as funções de forma encontradas para elementos de oito e quatro nós.

Nas equações originais, tem-se termos de derivada de segunda ordem, os quais serão reduzidos aplicando o teorema de Green, para introdução das condições de contorno.

Aplicando então o método de Galerkin nas equações e o teorema de Green, nos termos de segunda ordem tem-se:

$$\begin{aligned} \int_A (\text{função de forma}) \times (\text{resíduo}) dA &= 0 \\ \int_A N_i \left(\rho N_k u_k \frac{\partial N_j}{\partial x} u_j + \rho N_k v_k \frac{\partial N_j}{\partial y} u_j + \frac{\partial M_l}{\partial y} p_l - \rho g_x - \mu \left(\frac{\partial^2 u_j}{\partial x^2} + \frac{\partial^2 u_j}{\partial y^2} \right) \right) dA &= 0 \\ \int_A N_i \left(\rho N_k u_k \frac{\partial N_j}{\partial x} v_j + \rho N_k v_k \frac{\partial N_j}{\partial y} v_j + \frac{\partial M_l}{\partial y} p_l - \rho g_y - \mu \left(\frac{\partial^2 v_j}{\partial x^2} + \frac{\partial^2 v_j}{\partial y^2} \right) \right) dA &= 0 \end{aligned} \quad (4.17)$$

nos termos de segunda ordem :

$$\int_A N_i \left(\mu \left(\frac{\partial^2 u_j}{\partial x^2} + \frac{\partial^2 u_j}{\partial y^2} \right) \right) dA = \mu \int_{\Gamma} N_i \frac{\partial N_j}{\partial n} u_j ds - \mu \int_A \left(\frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} u_j + \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial y} u_j \right) dA$$

$$\int_A N_i \left(\mu \left(\frac{\partial^2 v_j}{\partial x^2} + \frac{\partial^2 v_j}{\partial y^2} \right) \right) dA = \mu \int_{\Gamma} N_i \frac{\partial N_j}{\partial n} v_j ds - \mu \int_A \left(\frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} v_j + \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial y} v_j \right) dA$$
(4.18)

os termos de integral de superfície são divididos em dois outros termos que representam as condições de contorno essencial, que não são aplicadas através desta forma, e as condições de contorno naturais:

$$\mu \int_{\Gamma} N_i \frac{\partial N_j}{\partial n} u_j ds = \mu \int_{\Gamma_1} N_i \frac{\partial N_j}{\partial n} u_j d\Gamma_1 + \mu \int_{\Gamma_2} N_i \frac{\partial u_j}{\partial n} d\Gamma_2$$

$$\mu \int_{\Gamma} N_i \frac{\partial N_j}{\partial n} v_j = \mu \int_{\Gamma_1} N_i \frac{\partial N_j}{\partial n} v_j d\Gamma_1 + \int_{\Gamma_2} N_i \frac{\partial v_j}{\partial n} d\Gamma_2$$
(4.19)

As condições de contorno essenciais são aplicadas substituindo o valor da condição na matriz que representa o lado direito das equações.

Substituindo (4.19) em (3.3):

$$\int_A N_i N_k u_k \rho \frac{\partial N_j}{\partial x} u_j + N_i N_k v_k \rho \frac{\partial N_j}{\partial y} u_j + N_i \frac{\partial M_l}{\partial x} p_l - N_i \rho g_x + \mu \left(\frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} u_j + \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial y} u_j \right) dA - \mu \int_{\Gamma} N_i \frac{\partial u_j}{\partial n} d\Gamma = 0$$

$$\int_A N_i N_k u_k \rho \frac{\partial N_j}{\partial x} v_j + N_i N_k v_k \rho \frac{\partial N_j}{\partial y} v_j + N_i \frac{\partial M_l}{\partial y} p_l - N_i \rho g_y + \mu \left(\frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} v_j + \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial y} v_j \right) dA - \mu \int_{\Gamma} N_i \frac{\partial v_j}{\partial n} d\Gamma = 0$$
(4.20)

e a equação da continuidade fica:

$$\int_A M_i \left(\frac{\partial N_j}{\partial x} u_j + \frac{\partial N_j}{\partial y} v_j \right) dA = 0 \quad (4.21)$$

Novamente aqui, como na geração da malha, as equações estão escritas de uma forma em relação a um sistema de coordenadas globais, plano físico (x,y) , e devem ser transformadas para um sistema de coordenadas locais, plano normalizado (λ,κ) onde foram encontradas as funções de forma.

Como já foram trabalhadas na secção 4-1, as transformações de um sistema global para um local, de forma análoga aqui podem ser aplicadas todas as transformações encontradas.

$$\begin{aligned} \frac{\partial N}{\partial x} &= \hat{J}_{11} \frac{\partial N}{\partial \lambda} + \hat{J}_{12} \frac{\partial N}{\partial \kappa} \\ \frac{\partial N}{\partial y} &= \hat{J}_{21} \frac{\partial N}{\partial \lambda} + \hat{J}_{22} \frac{\partial N}{\partial \kappa} \end{aligned}$$

\hat{J}_{ij} são os coeficientes da matriz jacobiana inversa, que agora é escrita da forma:

$$[\mathbf{J}] = \begin{bmatrix} \frac{\partial x}{\partial \lambda} & \frac{\partial y}{\partial \lambda} \\ \frac{\partial x}{\partial \kappa} & \frac{\partial y}{\partial \kappa} \end{bmatrix}$$

$$[\mathbf{J}]^{-1} = \frac{1}{\begin{pmatrix} \frac{\partial x}{\partial \lambda} \frac{\partial y}{\partial \kappa} - \frac{\partial y}{\partial \lambda} \frac{\partial x}{\partial \kappa} \end{pmatrix}} \times \begin{bmatrix} \frac{\partial y}{\partial \kappa} & -\frac{\partial y}{\partial \lambda} \\ -\frac{\partial x}{\partial \kappa} & \frac{\partial x}{\partial \lambda} \end{bmatrix}$$

$$dxdy = \mathbf{J}d\lambda d\kappa$$

Como em todas as transformações já apresentadas, para a integração das equações também será usado o método de integração numérica usado na geração da malha. Então as integrais de área são escritas da forma:

$$\int_A dA = \int_{-1}^1 \int_{-1}^1 \mathbf{J} d\lambda d\kappa = \sum_{i=1}^3 \sum_{j=1}^3 \varpi_i \varpi_j$$

Manipulando as equações (4.6) tem-se termos que podem ser tratados de forma especial, alguns valores podem ser considerados como dados ou já calculados em uma iteração anterior à corrente, então estes termos são passados para o lado direito das equações.

Para o lado direito são passados os termos que representam as forças de campo e as condições de contorno de derivada. Na equação (3.19) os termos u e v são considerados como valores de condições iniciais ou já calculados em uma iteração anterior.

Separando os termos das equações para a montagem da matriz do elemento, desenvolvendo o termo de integral de superfície, as equações são escritas da forma:

$$\begin{aligned} \int_A N_i N_k u_k \rho \frac{\partial N_j}{\partial x} u_j + N_i N_k v_k \rho \frac{\partial N_j}{\partial y} u_j + N_i \frac{\partial M_l}{\partial x} p_l + \mu \left(\frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} u_j + \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial y} u_j \right) dA &= \int_A N_i \rho g_x dA + \mu \int_{\Gamma} N_i \frac{\partial u_j}{\partial n} d\Gamma \\ \int_A N_i N_k u_k \rho \frac{\partial N_j}{\partial x} v_j + N_i N_k v_k \rho \frac{\partial N_j}{\partial y} v_j + N_i \frac{\partial M_l}{\partial y} p_l + \mu \left(\frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} v_j + \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial y} v_j \right) dA &= \int_A N_i \rho g_y dA + \mu \int_{\Gamma} N_i \frac{\partial v_j}{\partial n} d\Gamma \end{aligned} \quad (4.22)$$

Então a matriz do elemento é representada por:

$$\int_A \begin{bmatrix} N_1 u \frac{\partial N_1}{\partial x} + \dots & N_1 \frac{\partial M_1}{\partial x} & 0 & N_1 u \frac{\partial N_2}{\partial x} + \dots & 0 & \dots \\ M_1 \frac{\partial N_1}{\partial x} & 0 & M_1 \frac{\partial N_1}{\partial y} & M_1 \frac{\partial N_2}{\partial x} & M_1 \frac{\partial N_2}{\partial y} & \dots \\ 0 & N_1 \frac{\partial M_1}{\partial x} & N_1 u \frac{\partial N_1}{\partial x} + \dots & 0 & N_1 u \frac{\partial N_2}{\partial x} + \dots & \dots \end{bmatrix} \begin{bmatrix} u_1 \\ p_1 \\ v_1 \\ u_2 \\ v_2 \end{bmatrix} = \int_A \begin{bmatrix} N_1 \rho g_x \\ 0 \\ N_1 \rho g_y \\ N_2 \rho g_x \\ N_2 \rho g_y \end{bmatrix} dA + \int_{\Gamma} \begin{bmatrix} \mu N_1 \frac{\partial}{\partial n} \\ 0 \\ \mu N_1 \frac{\partial}{\partial n} \\ \mu N_2 \frac{\partial}{\partial n} \\ \mu N_2 \frac{\partial}{\partial n} \end{bmatrix} d\Gamma \quad (4.23)$$

Nota-se que a matriz resultante não é uma matriz simétrica, como na geração da malha.

A matriz apresentada em (4.23), por ser relativa ao elemento, está com uma numeração local das variáveis, que devem ser adaptadas à numeração global da malha para formação da matriz global.

Simplificando a representação das equações (4.20) e (4.21) pode-se representá-las por:

$$\mathbf{A}\omega = \mathbf{C} + \mathbf{D}$$

onde ω é da forma:

$$\omega_i = \begin{bmatrix} u_i \\ p_i \\ v_i \end{bmatrix}$$

os coeficientes da matriz **A** são dados por:

$$a_{ij} = \int_A \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix} dA$$

onde

$$\begin{aligned} b_{11} &= N_i N_k u_k \rho \frac{\partial N_j}{\partial x} + N_i N_k v_k \rho \frac{\partial N_j}{\partial y} + \mu \left(\frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial y} \right) \\ b_{12} &= N_i \frac{\partial M_i}{\partial x} & b_{13} &= 0 & b_{21} &= M_i \frac{\partial N_j}{\partial x} \\ b_{22} &= 0 & b_{23} &= M_i \frac{\partial N_j}{\partial y} & b_{31} &= 0 \\ b_{32} &= N_i \frac{\partial M_j}{\partial y} & b_{33} &= b_{11} \end{aligned}$$

os coeficientes da matriz **C** são dados por:

$$c_i = \int_A \begin{bmatrix} N_i \rho g_x \\ 0 \\ N_i \rho g_y \end{bmatrix} dA$$

e os termos da matriz **D** são dados por:

$$d_i = \mu \int_{\Gamma} \begin{bmatrix} N_i \frac{\partial u_j}{\partial n} \\ 0 \\ N_i \frac{\partial v_j}{\partial n} \end{bmatrix} d\Gamma$$

4-3 Equação da Energia

Após a aproximação da equação de Navier-Stokes, temos todo o campo de velocidades e pressão do problema. A equação da Energia é implementada ao problema, para casos em convecção forçada, onde se calcula o campo de temperatura separado do campo de velocidades.

A equação da Energia aqui utilizada foi a apresentada no capítulo 1, tratada no plano físico (x,y), com as seguintes considerações, Burmeister (1983):

- Caso bidimensional
- Propriedades constantes
- Sem fonte de calor imposta
- Regime permanente
- Convecção forçada

A equação da energia simplificada tratada será a representada pela equação (1.3).

O tratamento numérico para equação da Energia, será semelhante ao dado à equação da geração da malha e às equações de Navier-Stokes.

Para a aproximação da equação foi utilizado o elemento de oito nós.

Aplicando aqui todas as aproximações já desenvolvidas para as variáveis e suas derivadas, temos para a temperatura:

$$t = \sum_{j=1}^8 N_j t_j$$

$$\frac{\partial t}{\partial x} = \sum_{j=1}^8 \frac{\partial N}{\partial x_j} t_j \quad \frac{\partial t}{\partial y} = \sum_{j=1}^8 \frac{\partial N}{\partial y_j} t_j$$

Nesta equação, a variável dependente é t , temperatura, e todos os termos envolvendo as velocidades são conhecidos, então estes termos são tratados como constantes e/ou passados para o lado direito da equação.

$$\rho C_p \left(u \frac{\partial}{\partial x} + v \frac{\partial}{\partial y} \right) - k \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) = 2 * \mu \left[\left(\frac{\partial u}{\partial x} \right)^2 + \left(\frac{\partial v}{\partial y} \right)^2 \right] + \left[\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right]^2 \quad (4.24)$$

As velocidades e derivadas são aproximadas pelas expressões apresentadas no desenvolvimento das equações de Navier-Stokes:

$$u = \sum_{i=1}^8 N_i u_i \quad v = \sum_{i=1}^8 N_i v_i$$

$$\frac{\partial u}{\partial x} = \sum_{i=1}^8 \frac{\partial N}{\partial x_i} u_i \quad \frac{\partial u}{\partial y} = \sum_{i=1}^8 \frac{\partial N}{\partial y_i} u_i$$

$$\frac{\partial v}{\partial x} = \sum_{i=1}^8 \frac{\partial N}{\partial x_i} v_i \quad \frac{\partial v}{\partial y} = \sum_{i=1}^8 \frac{\partial N}{\partial y_i} v_i$$

Aplicando o método de Galerkin na equação da Energia e o teorema de Green nos termos de segunda ordem tem-se:

$$\int_A (\text{função de forma}) \times (\text{resíduo}) dA = 0$$

$$\int_A N \left(\rho C_p \left(u \frac{\partial}{\partial x} + v \frac{\partial}{\partial y} \right) - k \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) \right) dA = \int_A N \left(2 * \mu \left[\left(\frac{\partial u}{\partial x} \right)^2 + \left(\frac{\partial v}{\partial y} \right)^2 \right] + \left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right)^2 \right) dA \quad (4.25)$$

Escrevendo para um elemento, substituindo as aproximações para as variáveis:

$$\int_A N_i \left(\rho C_p \left(u \frac{\partial N_j}{\partial x} t_j + v \frac{\partial N_j}{\partial y} t_j \right) - k \left(\frac{\partial t}{\partial x^2} + \frac{\partial t}{\partial y^2} \right) \right) dA = \int_A N_i \left(2^* \mu \left(\left(\frac{\partial u}{\partial x} \right)^2 + \left(\frac{\partial v}{\partial y} \right)^2 \right) + \left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right)^2 \right) dA \quad (4.26)$$

Nos termos de segunda ordem, aplica-se o teorema de Green para redução,

$$\int_A N_i \left(k \left(\frac{\partial^2 t_j}{\partial x^2} + \frac{\partial^2 t_j}{\partial y^2} \right) \right) dA = k \int_{\Gamma} N_i \frac{\partial N_j}{\partial n} t_j d\Gamma + k \int_{\Gamma} N_i \frac{\partial t_j}{\partial n} d\Gamma - k \int_A \frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} t_j + \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial y} t_j dA \quad (4.27)$$

Substituindo na equação original, encontra-se a forma final para o tratamento numérico:

$$\int_A N_i \left(\rho C_p \left(u \frac{\partial N_j}{\partial x} t_j + v \frac{\partial N_j}{\partial y} t_j \right) + k \left(\frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} t_j + \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial y} t_j \right) \right) dA = k \int_{\Gamma} N_i \frac{\partial t_j}{\partial n} d\Gamma + \int_A N_i \left(2^* \mu \left(\left(\frac{\partial u}{\partial x} \right)^2 + \left(\frac{\partial v}{\partial y} \right)^2 \right) + \left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right)^2 \right) dA \quad (4.28)$$

As transformações necessárias para integração numérica são iguais às das equações de Navier-Stokes, e o procedimento de cálculo também é o mesmo.

Da equação final, tem-se a formação da matriz do elemento, de forma semelhante às já apresentadas.

$$\int_A \begin{bmatrix} N_1 \rho C_p u \frac{\partial N_1}{\partial x} + \dots & N_1 \rho C_p u \frac{\partial N_2}{\partial x} + \dots \\ N_2 \rho C_p u \frac{\partial N_1}{\partial x} + \dots & N_2 \rho C_p u \frac{\partial N_2}{\partial x} + \dots \\ \vdots & \vdots \\ N_8 \rho C_p u \frac{\partial N_1}{\partial x} + \dots & N_8 \rho C_p u \frac{\partial N_2}{\partial x} + \dots \end{bmatrix} dA \times \begin{bmatrix} t_1 \\ t_2 \\ \vdots \\ t_8 \end{bmatrix} = \int_{\Gamma} \begin{bmatrix} N_1 \frac{\partial}{\partial x} \\ N_2 \frac{\partial}{\partial x} \\ \vdots \\ N_8 \frac{\partial}{\partial x} \end{bmatrix} d\Gamma + \int_A \begin{bmatrix} N_1 (2 * \mu (\frac{\partial u}{\partial x})^2 + \dots) \\ N_2 (2 * \mu (\frac{\partial u}{\partial x})^2 + \dots) \\ \vdots \\ N_8 (2 * \mu (\frac{\partial u}{\partial x})^2 + \dots) \end{bmatrix} dA$$

(4.29)

A montagem da matriz global será feita seguindo o mesmo procedimento até agora aplicado, respeitando a numeração global de cada nó.

4-4 Técnica de solução das matrizes

Para todas as equações tratadas numericamente aqui, o que foi obtido como produto final foi um sistema de equações que deve ser resolvido para se obter a solução procurada.

Vários são os métodos que poderiam ser utilizados para esta tarefa, mas o método aqui empregado foi o mesmo também aplicado por Taylor (1981), e que foi primeiramente desenvolvido por Hood (1976).

O método frontal de solução para matrizes não simétricas apresenta algumas vantagens sobre os outros métodos de solução. Hood (1976) em seu trabalho de apresentação deste método, o comparou com outros mais frequentemente usados e pode relatar algumas destas vantagens:

- Menor uso de memória para solução.
- Tempo de cálculo menor.

A filosofia frontal de solução, de maneira geral, pode ser explicada seguindo o raciocínio:

- A rotina frontal começa alocando uma matriz, chamada matriz frontal ($m_{\text{fron}} \times m_{\text{fron}}$), que reserva o espaço onde serão feitas todas as reduções necessárias. Nota-se logo uma primeira vantagem que é a possibilidade de solução de um sistema de 100×100 usando uma matriz frontal de 10×10 , por exemplo.

- Com esta matriz alocada, para cada elemento da malha, é juntada à matriz frontal a matriz do elemento, respeitando a numeração dos nós.

- Para cada elemento, há as variáveis associadas, e para cada variável i quando não houver mais contribuições de nenhum elemento, a matriz frontal é reduzida eliminando-se a equação para aquela variável, e a equação eliminada é guardada.

- Isto se repete até que todas as variáveis sejam completadas e as respectivas equações forem guardadas, então é usada uma substituição tipo Gauss-Seidel para encontrar a solução.

Para este método, detalhes como pivotamento, armazenamento de equações, substituição, podem ser acompanhados em Taylor (1981) e Hood (1976).

4-5 Critério de convergência

Todo o procedimento mostrado é repetido a cada iteração, e o processo é encerrado quando for obtida uma tolerância requerida para convergência. A convergência é atingida segundo o critério:

$$\frac{(i_{iter} - i_{iter-1})}{i_{iter}} < \text{tolerância}$$

esta verificação é feita para todas as variáveis em todos os nós da malha.

Para os casos tratados foi usada $\text{tolerância}=0.001$.

Caso a tolerância não seja atingida, as variáveis são atualizadas como:

$$\hat{i}_{iter-1} = (i_{iter-1} + i_{iter}) * \text{relaxaçã}$$

Para a solução de todas as equações tratadas, foi usada $\text{relaxação}=0.4$.

Capítulo 5

Tópicos computacionais

Como já foi mencionado, foram desenvolvidos três programas computacionais, *GridGen*, *Gridg* e *Therm* e adaptado o programa *Fluid* que juntos foram os responsáveis por todos os cálculos dos resultados.

De forma mais detalhada, serão agora descritos alguns dos mais importantes tópicos, relativos à programação dos programas apresentados, tais como: funções usadas, estrutura de programação, estrutura de dados, alocação e uso de memória.

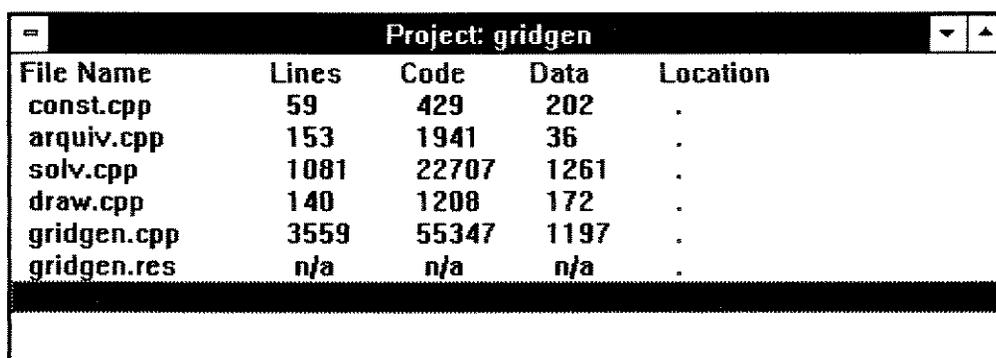
5-1 Programas *GridGen*, *Gridg* e *Therm*

Arquivo de projeto

GridGen é um programa desenvolvido para trabalhar sob ambiente Windows® e que oferece uma interatividade gráfica ao usuário para maior controle e visualização de dados e resultados. Com este programa é possível, pelo uso das funções gráficas desenvolvidas, a geração da geometria física do problema, imposição das condições de contorno necessárias à solução do campo de escoamento e de temperatura e visualização de resultados. Como é um programa para trabalhar sob ambiente Windows®, ele foi desenvolvido utilizando o conceito de programação orientada a objetos e foi utilizado o compilador Borland® C++ 3.1.

A programação orientada a objetos apresenta uma nova estruturação dos programas e a definição de classes que definem quais as particularidades que serão trabalhadas e o programa se concentra nelas. A programação para Windows® também apresenta uma diferenciação que é a programação por eventos, ou seja, algo só é realizado se for dado um comando antes ou se houver acontecido algum evento específico, e requer uma estruturação diferente da estruturação clássica da programação linear.

O programa GridGen é composto de vários arquivos cabeçalhos *.h* já disponíveis e alguns sub-programas *.cpp* mostrados na figura 7.1 que são compilados separadamente e agrupados pelo recurso de projeto disponível pelo compilador usado. É criado um arquivo de projeto *gridgen.prj* o qual reúne todas informações de todos os arquivos que compõem o programa principal. São utilizadas também várias bibliotecas padrão do C++ necessárias para criação de janelas, quadros de diálogo, menus, etc, e o arquivo *gridgen.res* é um arquivo de recurso onde são montados todos os diálogos, menus, cursor, etc.



File Name	Lines	Code	Data	Location
const.cpp	59	429	202	.
arquiv.cpp	153	1941	36	.
solv.cpp	1081	22707	1261	.
draw.cpp	140	1208	172	.
gridgen.cpp	3559	55347	1197	.
gridgen.res	n/a	n/a	n/a	.

Figura 5.1: janela do arquivo de projeto *gridgen.prj*

Arquivos cabeçalhos

1	#include <owl.h>	disponível em C++
2	#include <stdio.h>	"
3	#include <stdlib.h>	"
4	#include <string.h>	"
5	#include <math.h>	"
6	#include <fstream.h>	"
7	#include <iomanip.h>	"
8	#include <filedialog.h>	"
9	#include <inputdialog.h>	"
10	#include <dir.h>	"
11	#include <dos.h>	"
12	#include <window.h>	"
13	#include <time.h>	"
14	#include <edit.h>	"
15	#include <radiobutton.h>	"
16	#include <alloc.h>	"
17	#include "const.h"	desenvolvida
18	#include "draw.h"	"
19	#include "dimen.h"	"
20	#include "extern.h"	"

Tabela 5.1: principais arquivos cabeçalhos utilizados

Classes

As principais classes que foram utilizadas são:

- TGridGenApp::TApplication → classe da aplicação
define parâmetros gerais à aplicação
- TGridGenWindow::TWindow → classe da janela principal
cria e define parâmetros gerais da janela principal
- TToolsPalette::TWindow → classe da paleta de ferramentas
cria e define parâmetros gerais da paleta de ferramentas
- TD(número)Dlg::TDialog → classes de diálogo
cria quadros de diálogo
- TDrawingTool → classe das ferramentas de
desenho cria e define parâmetros gerais das ferramentas

Janela principal

A janela principal do programa é mostrada abaixo, figura 5.2.

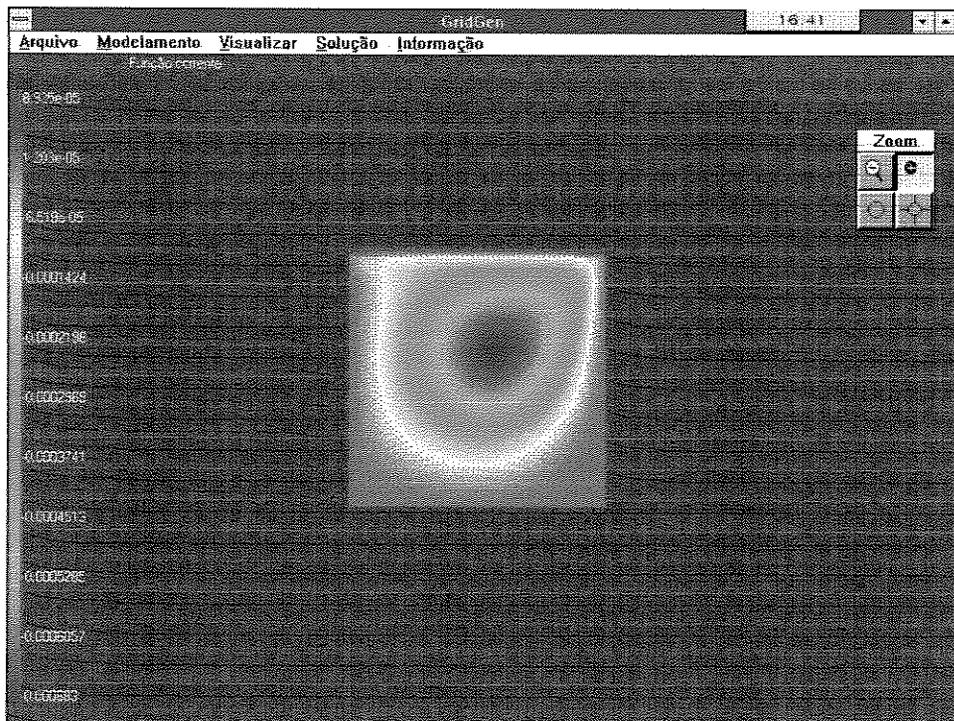


Figura 5.2: janela principal com exemplo para cavidade- *GridGen*

Comandos disponíveis

Todos os comandos disponíveis no programa são acessados a partir do menu principal pelos itens: Arquivo, Modelamento, Visualizar, Solução, Informação. Cada item do menu principal apresenta outros sub-ítems como mostra a figura 5.3.

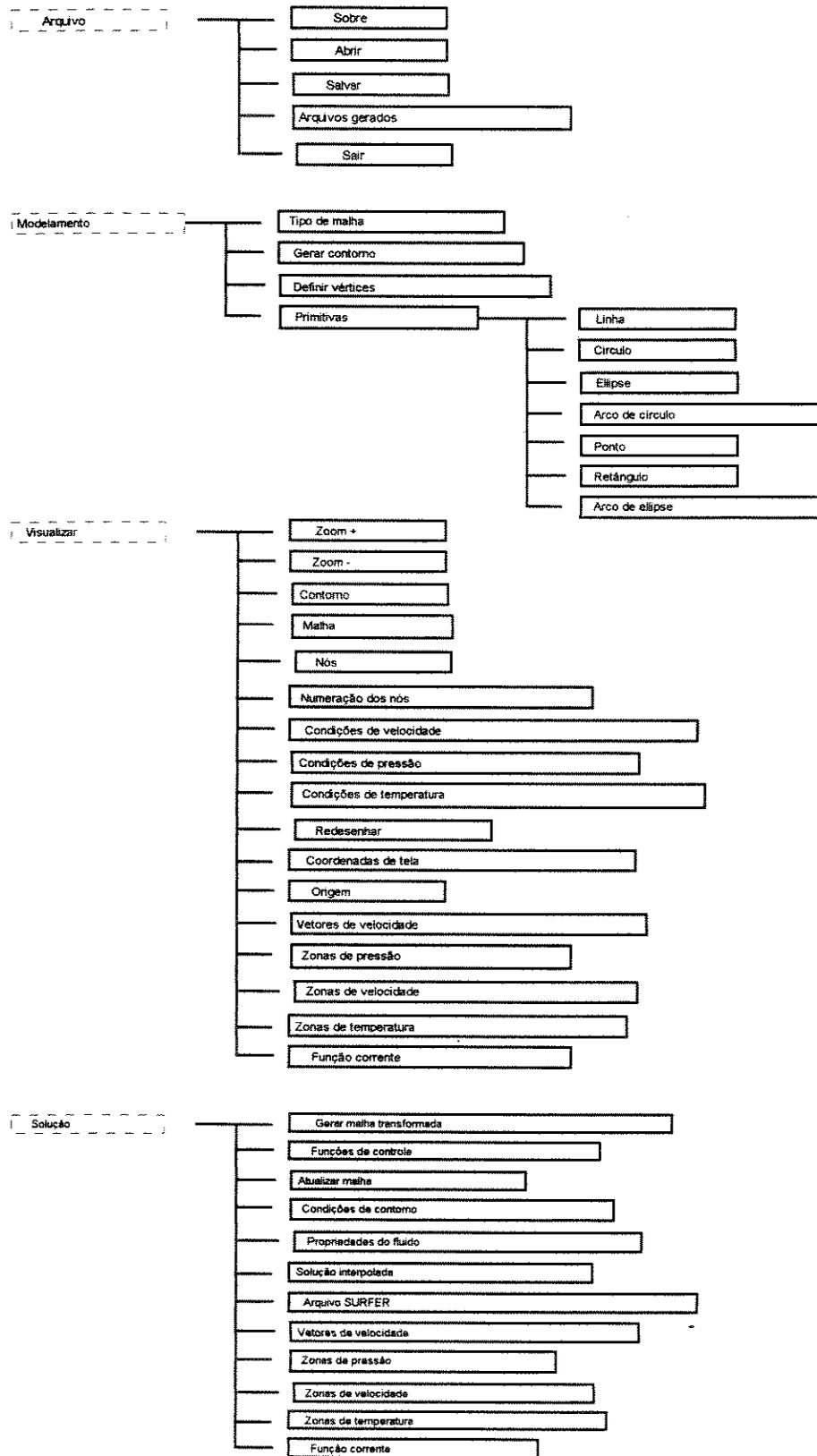


Figura 5.3: Relação dos comandos disponíveis no programa

Descrição de alguns dos comandos existentes.

Abrir: Abre um arquivo referente a uma malha escolhida por meio de um quadro de diálogo abrir padrão. Os arquivos contendo valores de velocidade, pressão e temperatura são lidos separadamente.

Salvar: Salva todos os dados referentes à malha gerada.

Os comandos sob o item *Modelagem* são para a definição das propriedades da geometria física e para a geração da malha tais como: escolha do tipo de malha simples ou multi conectada, informação dos pontos que formam os vértices e também o uso das ferramentas para construção da geometria. As primitivas geométricas são acessadas de forma semelhantes, como por exemplo o caso de uma linha, pelo quadro de diálogo, figura 5.4.

O diálogo 'Linha' possui as seguintes seções:

- De:** Novo ponto; Último ponto gerado. Campos para x1 e y1.
- Para:** Novo ponto; Primeiro ponto gerado. Campos para x2 e y2.
- Número de pontos:** Campo para n.
- Distribuição dos pontos:** Uniforme; Crescente; Decrescente.
- Botões OK e Cancel.

Figura 5.4: quadro de diálogo para construção de linhas

No caso de linha são informados os pontos iniciais e finais, a distribuição que se deseja ter e a quantidade de pontos formadores desta linha.

Sob o item *Visualizar* é escolhido o que será mostrado na janela principal segundo o esquema da figura 5.3. No item *Solução* são dados os comandos para geração da malha transformada, são definidas as funções de controle para geração da malha, são impostas as condições de

contorno para escoamento e energia e também são escolhidos quais gráficos dos resultados serão apresentados.

Todos os gráficos apresentados pelo programa são obtidos pelo uso de combinações de funções gráficas já existentes em C++. Algumas das funções gráficas mais utilizadas foram as de desenho de linhas, pexes e polígonos, dentre outras. Algumas das rotinas do programa, que serão apresentadas abaixo, utilizam as funções gráficas e demonstram a sua aplicação.

Rotina para desenho de linha.

```
void TGridGenWindow::DisplayContorno(HDC hdc)
{
    int i;
    float x,y;
    HPEN hWhitePen=CreatePen(PS_SOLID,1,RGB(255,255,255));
    HPEN hOldPen=SelectObject(hdc,hWhitePen);
    if(nbcon==0)return;
    for(i=1;i<=nbcon/2-1;i++)
    {
        x=posxy[i][1];
        y=posxy[i][2];
        MoveTo(hdc,x*Zoom,y*Zoom);
        x=posxy[i+1][1];
        y=posxy[i+1][2];
        LineTo(hdc,x*Zoom,y*Zoom);
        if(i==nbcon/2-1)
        {
            MoveTo(hdc,x*Zoom,y*Zoom);
            x=posxy[1][1];
            y=posxy[1][2];
            LineTo(hdc,x*Zoom,y*Zoom);
        }
    }
    SelectObject(hdc,hOldPen);
    DeleteObject(hWhitePen);
}
```

Parte de uma rotina de pintura de pixels

```
for(m=0;m<256;m++)
    if(vali>=valmax-(m+1)*deltaval)
    {
        cor=255-m;
        break;
    }
SetPixel(hdc,x*Zoom,y*Zoom,PALETTE_RGB(LogPal->palPalEntry[cor].peRed,
LogPal->palPalEntry[cor].peGreen,LogPal->palPalEntry[cor].peBlue));
```

Rotina para desenho de poligonos

```

void TGridGenWindow::DisplayMalha(HDC hdc)
{
    HBRUSH hGreenBrush=CreateSolidBrush(RGB(64,130,130));
    HBRUSH hOldBrush=SelectObject(hdc,hGreenBrush);
    HPEN hWhitePen=CreatePen(PS_SOLID,1,RGB(255,255,255));
    HPEN hOldPen=SelectObject(hdc,hWhitePen);
    POINT p[8];
    int i,j,itotx,itoty,ipoin;
    float x,y;
    if(nelem==0)return;
    for(i=1;i<=nelem;i++)
    {
        for(j=1;j<=8;j++)
        {
            ipoin=abs(lnods[i][j]);
            itotx=nadfm[ipoin];
            itoty=itotx+1;
            x=varbl[itotx];
            y=varbl[itoty];
            p[j-1].x=x*Zoom;
            p[j-1].y=y*Zoom;
        }
        Polygon(hdc,p,8);
    }
    SelectObject(hdc,hOldBrush);
    DeleteObject(hGreenBrush);
    SelectObject(hdc,hOldPen);
    DeleteObject(hWhitePen);
}

```

Estas e outras rotinas que estão no programa só são acionadas quando há algum evento específico, como já foi dito. Na maioria das rotinas o acionamento se dá a partir dos comandos que estão no menu principal, situado na parte de cima da janela principal. Cada comando especificado, (figura 5.3), está relacionado a um código que irá desencadear uma série de chamadas de funções ou irá chamar um quadro de diálogo. A seguir está um exemplo de chamadas de comandos que acessam os quadros de diálogos *sobre e abrir*.

Chamada dos quadros de diálogo sobre e abrir

```

void TGridGenWindow::WMCommand(TMessage& Msg)
{
    switch (COMMANDMSG(Msg))
    {
        HMENU hmenu;

    case 101:
        if(GetApplication()->ExecDialog(new TD101Dlg(this,101))==IDOK)
        {
            InvalidateRect(HWindow, NULL, TRUE);
        }
        break;

    case 102:
        int hcrOld;
        extern void LerArquivo(char FileName[80]);
        HCURSOR hcrWait;
        if ( GetApplication()->ExecDialog(new TFileDialog(this, SD_FILEOPEN,
            strcpy(FileName, "*.GRD"))) == IDOK )
        {
            hcrWait=LoadCursor(NULL, IDC_WAIT);
            hcrOld=SetCursor(hcrWait);
            LerArquivo(FileName);
            SetCursor(hcrOld);
            if(FALHA==TRUE)
            {
                MessageBox(HWindow, "Erro de leitura ou arquivro inexistente", "Erro", MB_OK |
                    MB_ICONHAND);
                FALHA=FALSE;
            }
            InvalidateRect(HWindow, NULL, TRUE);
        }
        break;
    }
}

```

Alocação dinâmica

Em todo o programa existem variáveis que são acessíveis a todas as rotinas e outras que são acessíveis a algumas rotinas específicas, isto é, o escopo da variável. As variáveis que ocupam maiores dimensões na memória são alocadas dinamicamente, ou seja, só são criadas quando necessitadas e logo após o uso são destruídas e liberada a memória que ocupavam. Esta prática de alocação dinâmica dá ao programa a capacidade de poder ser utilizado para maior número de variáveis que se fosse construído da forma clássica. As variáveis locais, que são utilizadas somente dentro de uma rotina, são todas criadas dinamicamente e logo quando a rotina é finalizada elas são destruídas.

Algumas das variáveis que são alocadas dinamicamente estão apresentadas logo a seguir. Neste trecho da rotina é mostrada a criação das variáveis e logo após é feita uma verificação para certificar da perfeita alocação destas na memória.

Parte de uma rotina de alocação dinâmica

```
float (*posxy)[3]=(float (*)[3])farcalloc((MPOIN+1)*3,sizeof(float));
float *varb1=(float *)farcalloc(MTOTV+1,sizeof(float));
float *pres=(float *)farcalloc(MPOIN+1,sizeof(float));
float (*vel)[3]=(float(*)[3])farcalloc((MPOIN+1)*3,sizeof(float));
float *temp=(float *)farcalloc(MPOIN+1,sizeof(float));
float (*interp)[3]=(float(*)[3])farcalloc((MPOIN+1)*3,sizeof(float));
float (*vector)[3]=(float(*)[3])farcalloc((MPOIN+1)*3,sizeof(float));
float *corrente=(float *)farcalloc(MPOIN+1,sizeof(float));

void verificar(void)
{
  if(!posxy||!varb1||!pres||!vel||!temp||!interp||!vector||!corrente)
  {
    MessageBox(NULL, "Sem memória suficiente para alocar", "Erro", MB_OK |
      MB_ICONHAND);
    exit(0);
  }
}
```

Cores

Com relação às cores disponíveis, o Windows® oferece uma paleta própria de cores que é a paleta do sistema default. Em um adaptador gráfico tipo SuperVGA esta paleta contém no máximo 20 cores puras. Para simular as outras variedades de cores o Windows® usa um processo chamado ditherização, que pinta pixels de cores diferentes lado a lado para criar a ilusão de cores variadas. Mas este processo não pode ser usado quando aplicado para desenhar linhas ou pintar pixels, e quando aplicado para pintar regiões a textura apresentada não é satisfatória. Por estas razões foi utilizado outro processo para acessar mais variações de cores criando uma paleta lógica de cores de forma que para o caso de um adaptador tipo SuperVGA no modo 256 cores, é possível acessar e mostrar 256 tonalidades de cores puras simultaneamente, tornando possível uma melhor qualidade na mostra dos resultados, suavizando a transição de uma cor para outra.

A paleta que foi utilizada contém 256 tonalidade de cores puras variando do azul para o vermelho, e foi criada como mostra o trecho da rotina de criação a seguir.

Parte da rotina de criação da paleta de cores

```

LogPal=(LPLOGPALETTE)(new char[sizeof(LOGPALETTE)+
    sizeof(PALETTEENTRY)*(NumColors-1)]);
LogPal->palVersion=0x300;
LogPal->palNumEntries=NumColors;
for(i=0;i<64;i++)          //blue->cyan
{
    LogPal->palPalEntry[i].peRed=0;
    LogPal->palPalEntry[i].peGreen=i*4;
    LogPal->palPalEntry[i].peBlue=255;
    LogPal->palPalEntry[i].peFlags=0;
}
for(i=0;i<64;i++)          //cyan->green
{
    LogPal->palPalEntry[i+64].peRed=0;
    LogPal->palPalEntry[i+64].peGreen=255;
    LogPal->palPalEntry[i+64].peBlue=255-i*4;
    LogPal->palPalEntry[i+64].peFlags=0;
}
for(i=0;i<64;i++)          //green->yellow
{
    LogPal->palPalEntry[i+2*64].peRed=i*4;
    LogPal->palPalEntry[i+2*64].peGreen=255;
    LogPal->palPalEntry[i+2*64].peBlue=0;
    LogPal->palPalEntry[i+2*64].peFlags=0;
}
for(i=0;i<64;i++)          //yellow->red
{
    LogPal->palPalEntry[i+3*64].peRed=255;
    LogPal->palPalEntry[i+3*64].peGreen=255-i*4;
    LogPal->palPalEntry[i+3*64].peBlue=0;
    LogPal->palPalEntry[i+3*64].peFlags=0;
}
hLogPal=CreatePalette(LogPal);
SelectPalette(hdc,hLogPal,0);
RealizePalette(hdc);

```

Modo de mapeamento

Para se desenhar na tela qualquer entidade é necessário antes determinar quais as coordenadas que a definirão e depois plotá-las. O Windows® oferece alguns modos de mapeamento que poderão ser escolhidos de acordo com a facilidade que eles apresentam para cada caso de uso, e entre estes os mais comuns são: MM_TEXT, MM_ISOTROPIC, MM_LOMERIC. Estes modos de mapeamento definirão a maneira de como serão relacionadas as coordenadas lógicas com as coordenadas físicas do hardware. Isto é necessário já que todos

os comandos da GDI(*Graphic Device Independent*) usam os valores de coordenadas lógicas.

Por uma série de fatores favoráveis o modo de mapeamento utilizado foi o MM_ISOTROPIC, que possibilita usar fatores de escala variáveis e equivalentes nas direções x e y. Logo a seguir está uma parte da rotina utilizada para definição das escalas correntes, origem do sistema e dimensão lógica da janela.

Parte da rotina para configuração do modo de mapeamento

```
RECT rect;
GetClientRect(HWindow.&rect);
SetMapMode(hdc,MM_ISOTROPIC);
SetWindowExt(hdc,WndExt,-WndExt);
SetViewportExt(hdc,rect.right,rect.bottom);
SetViewportOrg(hdc,rect.right*.1,rect.bottom*.9);
c.x=rect.right/2;
c.y=rect.bottom/2;
z.x=ZoomX;
z.y=ZoomY;
if((z.x>=c.x)&&(z.y<=c.y))
{
    d.x=c.x-z.x;
    d.y=c.y-z.y;
}
else
if((z.x<=c.x)&&(z.y<=c.y))
{
    d.x=c.x-z.x;
    d.y=c.y-z.y;
}
else
if((z.x<=c.x)&&(z.y>=c.y))
{
    d.x=c.x-z.x;
    d.y=c.y-z.y;
}
else
if((z.x>=c.x)&&(z.y>=c.y))
{
    d.x=c.x-z.x;
    d.y=c.y-z.y;
}
o.x=o.x+d.x;
o.y=o.y+d.y;
SetViewportOrg(hdc,o.x,o.y);
```

Paleta de ferramentas

A paleta de ferramentas utilizada é uma facilidade a mais encontrada para agilizar os comandos de enquadramento da janela por meio de três comandos disponíveis: Zoom +, Zoom - e Centralizar. Cada operação é acessada quando selecionado o ícone na paleta. A paleta é uma janela tipo pop-up onde estarão alocados os ícones para acesso das ferramentas. A sua criação é semelhante a criação da janela principal, deriva da mesma classe pai TWindow, mas tem uma configuração especial porque é necessário relacionar cada ícone com uma função dentro do programa além de outros recursos que devem estar interligados. De uma maneira simplificada, a listagem a seguir descreve a definição da classe TToolsPalette::TWindow e também a sua configuração.

Descrição da classe TToolsPalette

```
class TToolsPalette:public TWindow
{
public:
    int IconWd,IconHt;
    int WndWd,WndHt;
    HICON hIcons[NUMICONWIDE*NUMICONHIGH];
    TPaintContext *PC;
    PTDrawingTool Tools[NUMICONWIDE*NUMICONHIGH];
    TToolsPalette(PTWindowsObject AParent,LPSTR ATitle,TPaintContext *ThePC);
    void HighlightIcon(HDC hDC,int i);
    virtual void Paint(HDC hDC,PAINTSTRUCT&);
    virtual void WMLButtonDown(TMessage&){WM_FIRST+WM_LBUTTONDOWN};
};
```

Construtor do objeto da classe da paleta

```

TToolsPalette::TToolsPalette(PWindowsObject AParent,LPSTR ATitle,TPaintContext
*ThePC):TWindow(AParent,ATitle)
{
  Attr.Style=WS_CHILD|WS_CAPTION|WS_VISIBLE;
  Tools[0]=new TZoomMenos;
  Tools[1]=new TZoomMais;
  Tools[2]=new TCircleTool;
  Tools[3]=new TCenterTool;
  for(int i=0;i<NUMICONHIGH*NUMICONWIDE;i++)
    hIcons[i]=Tools[i]->GetIcon();
  IconWd=GetSystemMetrics(SM_CXICON);
  IconHt=GetSystemMetrics(SM_CYICON);
  WndWd=IconWd*NUMICONWIDE+GetSystemMetrics(SM_CXBORDER)+2;
  WndHt=IconHt*NUMICONHIGH+GetSystemMetrics(SM_CYBORDER)
    +GetSystemMetrics(SM_CYCAPTION);
  ToolNdx=0;
  PC=ThePC;
  PC->Tool=Tools[ToolNdx];
}

```

Em relação ao programa *GridGen* estes são alguns dos pontos mais salientes específicos à programação.

Os programas *Gridg* e *Therm* são escritos em FORTRAN e são construídos de uma forma semelhante ao *Fluid*, (Taylor, 1981), seguindo o mesmo padrão de programação. As principais diferenças entre eles estão em como são formadas as matrizes elementares para cada equação e na presença de algumas rotinas específicas necessárias para cada caso.

Gridg

O programa *Gridg* apresenta algumas rotinas específicas que tratarão das condições de contorno com as funções P e Q não nulas. Os valores de P e Q são informados pelo *GridGen* e armazenados em um arquivo temporário que será lido pelo *Gridg*. A rotina CONTROLF listada a seguir descreve esta operação de leitura e armazenamento dos valores de P e Q para cada ponto da malha.

Rotina de leitura dos valores das funções de controle P e Q

```

SUBROUTINE CONTROLF(MPOIN,NPOIN,COORD,CFP,CFQ)

IMPLICIT REAL*8(A-H,O-Z)
IMPLICIT INTEGER*4 (I-N)
DIMENSION COORD(MPOIN,2),CFP(MPOIN),CFQ(MPOIN)
READ(8,*)PQSI,PNI
READ(8,*)PQSIX,PNIY
READ(8,*)AP,BP,CP,DP
READ(8,*)AQ,BQ,CQ,DQ
SGNQL=0.0
SGNQP=0.0
SGNNL=0.0
SGNNP=0.0
DO 10 I=1,NPOIN
  PQS=(COORD(I,1)+1.)
  PN=(COORD(I,2)+1.)
  CALL SINALQL(PQS,PQSI,SGNQL)
  CALL SINALQP(PQS,PQSIX,SGNQP)
  CFP(I)=-AP*SGNQL*EXP(-CP*ABS(PQS-PQSI))-
1    BP*SGNQP*EXP(-DP*((PQS-PQSIX)**2.+(PN-PNIY)**2.))**0.5)
  CALL SINALNL(PN,PNI,SGNNL)
  CALL SINALNP(PN,PNIY,SGNNP)
  CFQ(I)=-AQ*SGNNL*EXP(-CQ*ABS(PN-PNI))-
1    BQ*SGNNP*EXP(-DQ*((PQS-PQSIX)**2.+(PN-PNIY)**2.))**0.5)
10 CONTINUE
DO 20 I=1,NPOIN
  WRITE(9,*)I,CFP(I),CFQ(I)
20 CONTINUE
RETURN
END

```

As derivadas cruzadas, equação 4.13, seguindo a figura 4.5 que mostra uma célula em torno de um ponto, são calculadas para cada ponto de Gauss pela rotina listada a seguir.

Rotina para cálculo das derivadas cruzadas

```

SUBROUTINE DERIVCRU(D2XDQSDN,D2YDQSDN,IELEM,IGAUS,LNODS,VARB2,
1  MELEM,MTOTV,NADFM,MPOIN)
IMPLICIT REAL*8(A-H,O-Z)
IMPLICIT INTEGER*4(I-N)
DIMENSION LNODS(MELEM,8),VARB2(MTOTV),SHAPE(8),DERIV(2,8),
1  NADFM(MPOIN)

```

```

D2XDQSDN=0.0
D2YDQSDN=0.0
PQSA=0.0
PNA=0.0
PQSB=0.0
PNB=0.0
PQSC=0.0
PNC=0.0
PQSD=0.0
PND=0.0
XA=0.0
YA=0.0
XB=0.0
YB=0.0
XC=0.0
YC=0.0
XD=0.0
YD=0.0
IF(IGAUS.EQ.1)THEN
  PQSA=1.0
  PNA=1.0
  PQSB=1.0
  PNB=0.5491933384
  PQSC=0.5491933384
  PNC=1.0
  PQSD=0.5491933384
  PND=0.5491933384
ELSE IF(IGAUS.EQ.2)THEN
  PQSA=1.0
  PNA=0.2254033307
  PQSB=1.0
  PNB=-0.2254033307
  PQSC=0.5491933384
  PNC=0.2254033307
  PQSD=0.5491933384
  PND=-0.2254033307
ELSE IF(IGAUS.EQ.3)THEN
  PQSA=1.0
  PNA=-0.5491933384
  PQSB=1.0
  PNB=-1.0
  PQSC=0.5491933384
  PNC=-0.5491933384
  PQSD=0.5491933384
  PND=-1.0
ELSE IF(IGAUS.EQ.4)THEN
  PQSA=0.2254033307
  PNA=1.0
  PQSB=0.2254033307
  PNB=0.5491933384
  PQSC=-0.2254033307
  PNC=1.0
  PQSD=-0.2254033307
  PND=0.5491933384
ELSE IF(IGAUS.EQ.5)THEN
  PQSA=0.2254033307
  PNA=0.2254033307
  PQSB=0.2254033307
  PNB=-0.2254033307
  PQSC=-0.2254033307
  PNC=0.2254033307
  PQSD=-0.2254033307
  PND=-0.2254033307
ELSE IF(IGAUS.EQ.6)THEN
  PQSA=0.2254033307
  PNA=-0.5491933384
  PQSB=0.2254033307
  PNB=-1.0
  PQSC=-0.2254033307
  PNC=-0.5491933384
  PQSD=-0.2254033307
  PND=-1.0
ELSE IF(IGAUS.EQ.7)THEN
  PQSA=-0.5491933384
  PNA=1.0
  PQSB=-0.5491933384
  PNB=0.5491933384
  PQSC=-1.0
  PNC=1.0
  PQSD=-1.0
  PND=0.5491933384
ELSE IF(IGAUS.EQ.8)THEN
  PQSA=-0.5491933384
  PNA=0.2254033307
  PQSB=-0.5491933384
  PNB=-0.2254033307
  PQSC=-1.0
  PNC=0.2254033307
  PQSD=-1.0
  PND=-0.2254033307
ELSE IF(IGAUS.EQ.9)THEN
  PQSA=-0.5491933384
  PNA=-0.5491933384
  PQSB=-0.5491933384
  PNB=-1.0
  PQSC=-1.0
  PNC=-0.5491933384
  PQSD=-1.0
  PND=-1.0
ENDIF
DO 10 I=1,4
  IF(I.EQ.1)THEN
    XEQIV=PQSA
    YEQIV=PNA
  ELSE IF(I.EQ.2)THEN
    XEQIV=PQSB
    YEQIV=PNB
  ELSE IF(I.EQ.3)THEN
    XEQIV=PQSC
    YEQIV=PNC
  ELSE IF(I.EQ.4)THEN
    XEQIV=PQSD
    YEQIV=PND
  ENDF

```

```

CALL SHAPE8(DERIV.SHAPE,XEQIV,YEQIV)
IF(I.EQ.1)THEN
  DO 20 J=1,8
    JPOIN=IABS(LNODS(IELEM,J))
    ITOTX=NADFM(JPOIN)
    ITOTY=ITOTX+1
    XA=XA+SHAPE(J)*VARB2(ITOTX)
    YA=YA+SHAPE(J)*VARB2(ITOTY)
20  CONTINUE
  ELSE IF(I.EQ.2)THEN
    DO 30 J=1,8
      JPOIN=IABS(LNODS(IELEM,J))
      ITOTX=NADFM(JPOIN)
      ITOTY=ITOTX+1
      XB=XB+SHAPE(J)*VARB2(ITOTX)
      YB=YB+SHAPE(J)*VARB2(ITOTY)
30  CONTINUE
    ELSE IF(I.EQ.3)THEN
      DO 40 J=1,8
        JPOIN=IABS(LNODS(IELEM,J))
        ITOTX=NADFM(JPOIN)
        ITOTY=ITOTX+1
        XC=XC+SHAPE(J)*VARB2(ITOTX)
        YC=YC+SHAPE(J)*VARB2(ITOTY)
40  CONTINUE
      ELSE IF(I.EQ.4)THEN
        DO 50 J=1,8
          JPOIN=IABS(LNODS(IELEM,J))
          ITOTX=NADFM(JPOIN)
          ITOTY=ITOTX+1
          XD=XD+SHAPE(J)*VARB2(ITOTX)
          YD=YD+SHAPE(J)*VARB2(ITOTY)
50  CONTINUE
        ENDIF
10  CONTINUE
D2XDQSDN=4.92061459138*(XA-XC-XB+XD)
D2YDQSDN=4.92061459138*(YA-YC-YB+YD)
RETURN
END

```

A alocação da matriz elementar para as equações da geração da malha se dá, como no programa *Fluid*, na rotina MATRIX. O que estão sendo calculados e armazenados são os termos da matriz elementar, equação 4.12. E para a equação da geração da malha, considerando o sistema dado pela equação 4.10, a rotina MATRIX é escrita da seguinte forma listada a seguir.

Rotina de alocação da matriz elementar

```

SUBROUTINE MATRIX(EQRHS,FLUMX,LNODS,MELEM,
1      MPOIN,MTOTV,NADFM,NEVAB,NNODP,
2      KELEM,UNIT10,IUNIT10,VARB2,IITER,CFP,CFQ)
IMPLICIT REAL*8(A-H,O-Z)
REAL UNIT10
DIMENSION AREAW(9),CARPG(2,72),
1      CARTP(2,8),EQRHS(MTOTV),ERHSX(8),ERHSY(8),
2      FLUMX(16,16),LNODS(MELEM,8),NADFM(MPOIN),
3      SHAPG(72),
4      SHAPP(8),UNIT10(MELEM,340),
5      IUNIT10(MELEM,4),VARB2(MTOTV),CFP(MPOIN),CFQ(MPOIN)
DO 10 INODP=1,NNODP
ERHSX(INODP)=0.0
ERHSY(INODP)=0.0
10  CONTINUE
DO 20 IEVAB=1,NEVAB
DO 20 JEVAB=1,NEVAB
FLUMX(IEVAB,JEVAB)=0.0
20  CONTINUE
IELEM=IUNIT10(KELEM,1)
LGAUS=IUNIT10(KELEM,2)
NGAPA=IUNIT10(KELEM,3)
N=4
DO 130 IGAPA=1,NGAPA
N=N+1
SHAPG(IGAPA)=UNIT10(KELEM,N)
DO 140 IDIME=1,2
N=N+1
CARPG(IDIME,IGAPA)=UNIT10(KELEM,N)
140 CONTINUE
130 CONTINUE
DO 170 IGAUS=1,LGAUS
N=N+1
AREAW(IGAUS)=UNIT10(KELEM,N)
170 CONTINUE
DO 100 IGAUS=1,LGAUS
DAREA=AREAW(IGAUS)
DO 30 INODP=1,NNODP
SHAPP(INODP)=SHAPG(NNODP*(IGAUS-1)+INODP)
DO 30 IDIME=1,2
CARTP(IDIME,INODP)=CARPG(IDIME,NNODP*(IGAUS-1)+INODP)
30  CONTINUE
DXDQS=0.0
DXDN=0.0
DYDQS=0.0
DYDN=0.0
D2XDQSDN=0.0
D2YDQSDN=0.0
FP=0.0
FQ=0.0

```

```

X=0.0
Y=0.0
DO 50 INODP=1,NNODP
KPOIN=IABS(LNODS(IELEM,INODP))
ITOTX=NADFM(KPOIN)
ITOTY=ITOTX+1
SHAPE=SHAPP(INODP)
DXDQS=DXDQS+CARTP(1,INODP)*VARB2(ITOTX)
DXDN=DXDN+CARTP(2,INODP)*VARB2(ITOTX)
DYDQS=DYDQS+CARTP(1,INODP)*VARB2(ITOTY)
DYDN=DYDN+CARTP(2,INODP)*VARB2(ITOTY)
FP=FP+SHAPE*CFP(KPOIN)
FQ=FQ+SHAPE*CFQ(KPOIN)
50  CONTINUE
CALL DERIVCRU(D2XDQSDN,D2YDQSDN,IELEM,IGAUS,LNODS,VARB2,
1  MELEM,MTOTV,NADFM,MPOIN)
IF(IITER.EQ.1)THEN
DXDQS=1.0
DXDN=1.0
DYDQS=1.0
DYDN=1.0
ENDIF
VJACOB2=(DXDQS*DYDN-DXDN*DYDQS)**(2.)
DO 60 INODP=1,NNODP
kpoin=iabs(lnods(ielem,inodp))
itotx=nadfm(kpoin)
itoty=itotx+1
ERHSX(INODP)=ERHSX(INODP)+SHAPP(INODP)*DAREA*(
1  (-2.*DXDQS*DXDN-2.*DYDQS*DYDN)*D2XDQSDN+
2  VJACOB2*(FP*DXDQS+FQ*DXDN))
ERHSY(INODP)=ERHSY(INODP)+SHAPP(INODP)*DAREA*(
1  (-2.*DXDQS*DXDN-2.*DYDQS*DYDN)*D2YDQSDN+
2  VJACOB2*(FP*DYDQS+FQ*DYDN))
60  CONTINUE
INODP=0
DO 90 ICON1=1,15,2
IROWX=ICON1
IROWY=IROWX+1
INODP=INODP+1
SHAPI=SHAPP(INODP)
CARXI=CARTP(1,INODP)
CARYI=CARTP(2,INODP)
JNODP=0
DO 80 JCON1=1,15,2
JCOLX=JCON1
JCOLY=JCOLX+1
JNODP=JNODP+1
SHAPJ=SHAPP(JNODP)
CARXJ=CARTP(1,JNODP)
CARYJ=CARTP(2,JNODP)
FLUMX(IROWX,JCOLX)=FLUMX(IROWX,JCOLX)+DAREA*CARXI*CARXJ*
1  (DXDN**2+DYDN**2)+DAREA*CARYI*CARYJ*(DXDQS**2+DYDQS**2)

```

```

      FLUMX(IROWY,JCOLY)=FLUMX(IROWY,JCOLY)+DAREA*CARXI*CARXJ*
1     (DXDN**2+DYDN**2)+DAREA*CARYI*CARYJ*(DXDQS**2+DYDQS**2)
80    CONTINUE
90    CONTINUE
100   CONTINUE
      DO 110 INODP=1,NNODP
      KPOIN=IABS(LNODS(IELEM,INODP))
      ITOTX=NADFM(KPOIN)
      ITOTY=ITOTX+1
      EQRHS(ITOTX)=EQRHS(ITOTX)+ERHSX(INODP)
      EQRHS(ITOTY)=EQRHS(ITOTY)+ERHSY(INODP)
110   CONTINUE
      RETURN
      END

```

Therm

No programa *Therm* as modificações mais importantes se deram na rotina *MATRIX*, já que as equações a serem resolvidas são outras e a alocação agora será da equação 4.29.

Rotina de alocação da matriz elementar

```

SUBROUTINE MATRIX(COORD,DENSY,EQRHS,FLUMX,IAXSY,LNODS,MELEM,
1     MPOIN,MTOTV,NADFM,NEVAB,NNODP,
2     VARB2,VISCY,KELEM,
3     UNIT10,IUNIT10,CP,CK,VVELU,VVELV)
  IMPLICIT REAL*8(A-H,O-Z)
  IMPLICIT INTEGER*4 (I-N)
  REAL UNIT10
  DIMENSION AREAW(9),CARPG(2,72),
1     CARTP(2,8),COORD(MPOIN,2),EQRHS(MTOTV),
2     FLUMX(8,8),LNODS(MELEM,8),NADFM(MPOIN),
3     SHAPG(72),
4     SHAPP(8),VARB2(MTOTV),UNIT10(MELEM,340),
5     IUNIT10(MELEM,4),
6     VVELU(MELEM,8),VVELV(MELEM,8),ERHST(8)
  DO 10 INODP=1,NNODP
  ERHST(INODP)=0.0
10   CONTINUE
      DO 20 IEVAB=1,NEVAB
      DO 20 JEVAB=1,NEVAB
      FLUMX(IEVAB,JEVAB)=0.0
20   CONTINUE
  IELEM=IUNIT10(KELEM,1)

```

```

LGAUS=IUNIT10(KELEM,2)
NGAPA=IUNIT10(KELEM,3)
N=4

DO 130 IGAPA=1,NGAPA
N=N+1
SHAPG(IGAPA)=UNIT10(KELEM,N)
DO 140 IDIME=1,2
N=N+1
CARPG(IDIME,IGAPA)=UNIT10(KELEM,N)
140 CONTINUE
130 CONTINUE

DO 170 IGAUS=1,LGAUS
N=N+1
AREAW(IGAUS)=UNIT10(KELEM,N)
170 CONTINUE
DO 100 IGAUS=1,LGAUS
DAREA=AREAW(IGAUS)
DO 30 INODP=1,NNODP
SHAPP(INODP)=SHAPG(NNODP*(IGAUS-1)+INODP)
DO 30 IDIME=1,2
CARTP(IDIME,INODP)=CARPG(IDIME,NNODP*(IGAUS-1)+INODP)
30 CONTINUE
UVELY=0.0
TTEMP=0.0
VVELY=0.0
DUDX=0.0
DUDY=0.0
DVDX=0.0
DVDY=0.0
DUVELYX=0.0
DUVELYY=0.0
DVVELYX=0.0
DVVELYY=0.0
DO 50 INODP=1,NNODP
KPOIN=IABS(LNODS(IELEM,INODP))
ITOTT=NADFM(KPOIN)
SHAPE=SHAPP(INODP)
TTEMP=TTEMP+VARB2(ITOTT)*SHAPE
UVELY=UVELY+VVELU(KELEM,INODP)*SHAPE
VVELY=VVELY+VVELV(KELEM,INODP)*SHAPE
DUVELYX=DUVELYX+CARTP(1,INODP)*VVELU(KELEM,INODP)
DUVELYY=DUVELYY+CARTP(2,INODP)*VVELU(KELEM,INODP)
DVVELYX=DVVELYX+CARTP(1,INODP)*VVELV(KELEM,INODP)
DVVELYY=DVVELYY+CARTP(2,INODP)*VVELV(KELEM,INODP)
50 CONTINUE
DO 60 INODP=1,NNODP
ERHST(INODP)=0.0
60 CONTINUE

```

```
DO 90 ICON1=1,8
IROWT=ICON1
INODP=ICON1
SHAPI=SHAPP(INODP)
CARXI=CARTP(1,INODP)
CARYI=CARTP(2,INODP)
DO 80 JCON1=1,8
JCOLT=JCON1
JNODP=JCON1
SHAPJ=SHAPP(JNODP)
CARXJ=CARTP(1,JNODP)
CARYJ=CARTP(2,JNODP)
DIFFU=(CARXI*CARXJ+CARYI*CARYJ)*CK*DAREA
CONVC=(UVELY*CARXJ+VVELY*CARYJ)*SHAPI*CP*DENSY*DAREA
FLUMX(IROWT,JCOLT)=FLUMX(IROWT,JCOLT)+DIFFU+CONVC
70  CONTINUE
80  CONTINUE
90  CONTINUE
100 CONTINUE
DO 110 INODP=1,NNODP
KPOIN=IABS(LNODS(IELEM,INODP))
ITOTT=NADFM(KPOIN)
EQRHS(ITOTT)=EQRHS(ITOTT)+ERHST(INODP)
110 CONTINUE
RETURN
END
```

Capítulo 6

Apresentação e comparação de resultados

6-1 Geração da malha

Todo o trabalho apresentado é compreendido em quatro módulos computacionais que funcionam independentemente, mas que estão ligados de uma forma sequencial. O módulo *GridGen* roda sob ambiente Windows® e os módulos *Gridg*, *Fluid*, *Therm*, rodam sob ambiente MS-DOS®. Esta aparente incompatibilidade dos módulos é justificada pelo fato de que o módulo *GridGen* seja responsável principalmente pela visualização gráfica do problema e o ambiente Windows® oferece muitas vantagens e facilidades referentes a este tipo de trabalho, e os outros três módulos apenas pelo cálculo das variáveis, então estes podem ser usados em outras máquinas mesmo sob outros sistemas, já que estão escritos em FORTRAN e podem ser compilados em qualquer máquina sem muitas modificações, isto dá a possibilidade de serem trazidos somente os arquivos com resultados para análise de um problema maior em uma máquina pequena de mais fácil acesso.

Todas as malhas geradas e utilizadas estão apresentadas nas figuras 6.1, 6.2 e 6.3.

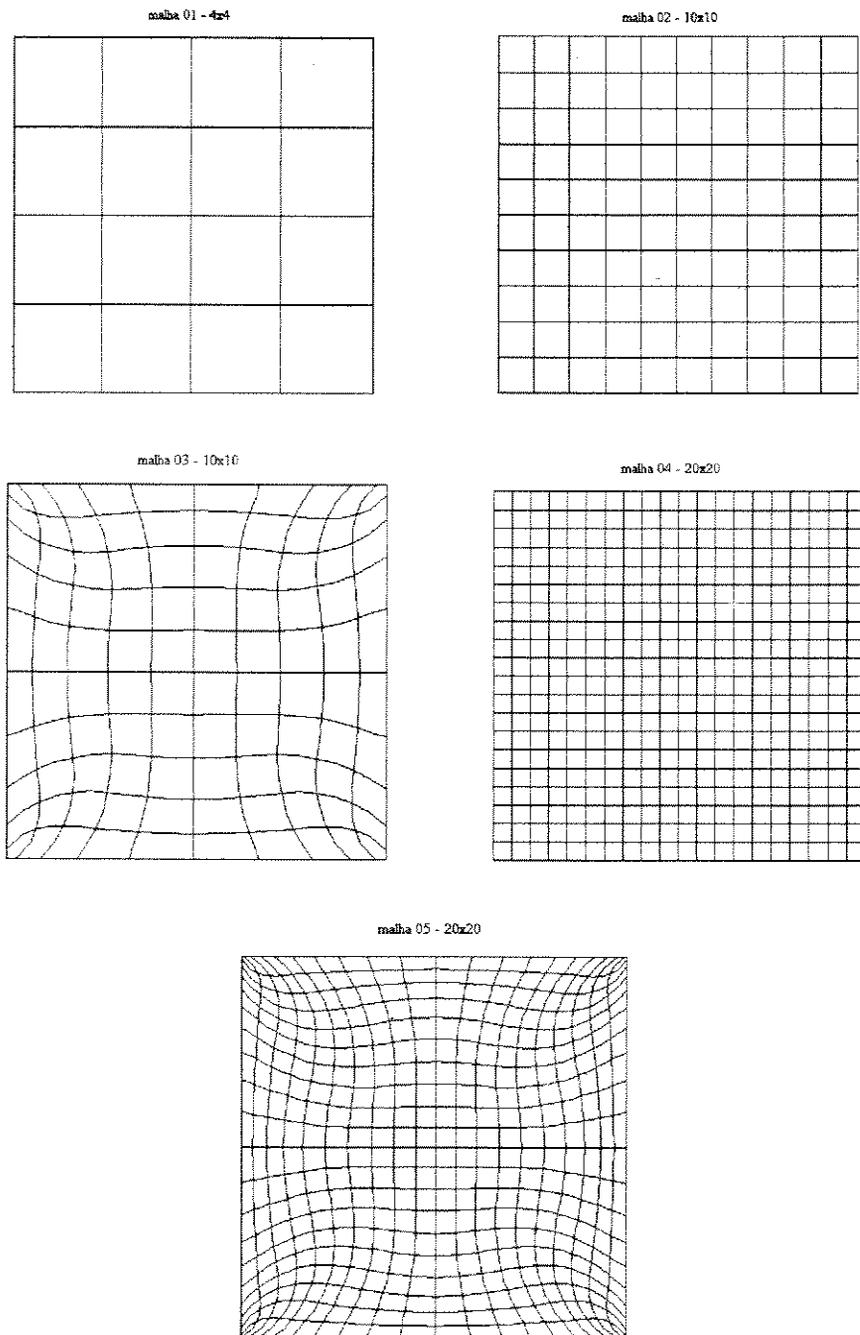


Figura 6.1: malhas utilizadas na cavidade

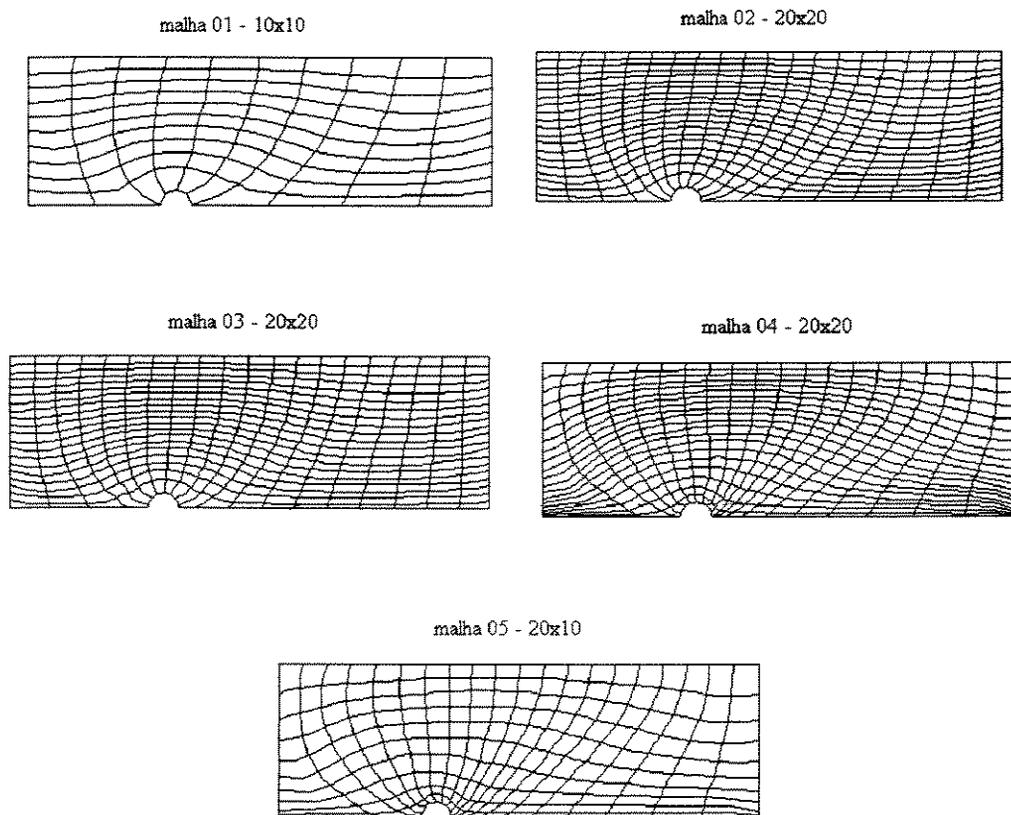


Figura 6.2: malhas utilizadas no cilindro

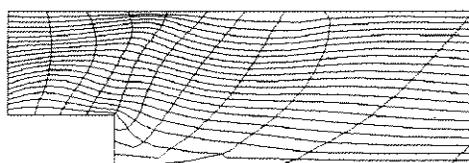
malha 01 - 10x10



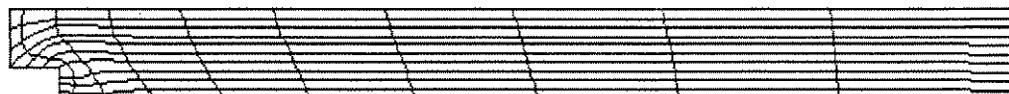
malha 02 - 20x20



detalhe



malha 03 - 10x10



malha 04 - 20x20

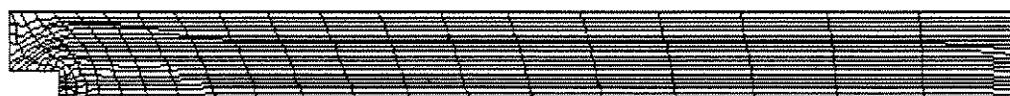


Figura 6.3: malhas utilizadas na expansão abrupta

A seguir estão algumas tabelas comparativas dos números de iterações do módulo *Gridg*.

Número de iterações para cálculo das malhas

malha	n° de nós	n° de iterações (equação 4.10)	n° de iterações (equação 4.11)
01	65	2	2
02	341	2	2
03	341	5	13
04	1281	2	2
05	1281	7	17

Tabela 6.1: iterações para cálculo das malhas para cavidade

Número de iterações para cálculo das malhas

malha	n° de nós	n° de iterações (equação 4.10)
01	341	6
02	1281	6
03	1281	6
04	1281	8
05	661	7

Tabela 6.2: iterações para cálculo das malhas para cilindro

Número de iterações para cálculo das malhas

malha	n° de nós	n° de iterações (equação 4.10)
01	341	7
02	1281	9
03	341	6
04	1281	6

Tabela 6.3: iterações para cálculo das malhas para expansão

Para todos os casos, nota-se que quanto mais regular a malha, menos iterações e conseqüentemente menor o tempo de cálculo necessário. O uso das funções de controle de malha também afeta o tempo de cálculo. Vale ressaltar que todas as malhas foram calculadas sem aproximações iniciais, sem condições iniciais que podem reduzir o número de iterações necessárias.

O programa *GridGen* gera automaticamente malhas com elementos de oito nós, mas para testar a qualidade dos resultados obtidos por uma malha, deve-se então resolver alguma equação que dependa da malha gerada para comparar as aproximações obtidas. Neste trabalho a malha gerada foi utilizada como parte dos dados de entrada necessários para o programa que aproxima as equações de Navier-Stokes apresentado por Taylor (1981) e posteriormente a equação da Energia.

Para o estudo do comportamento da malha, foram considerados três casos de escoamento de fluido, cujas aproximações são bem conhecidas e consideradas satisfatórias ou com soluções exatas, casos estes com geometrias conhecidas e definidas, permitindo várias configurações para geração da malha. Os casos tratados aqui foram:

- Escoamento em cavidade quadrada com fronteira deslizando,
- Escoamento em torno de um cilindro,
- Escoamento em uma expansão abrupta.

A seguir serão detalhados todos os casos tratados.

6-2 Escoamento em cavidade quadrada com fronteira deslizante

O escoamento em uma cavidade quadrada tem sido muito usado como um modelo para testes de algoritmos aproximadores das equações de Navier-Stokes. Rubin & Khosla (1977) e Nallasamy & Prasad (1977) entre outros obtiveram resultados para este modelo, até que Ghia (1982) obteve resultados que são considerados como solução padrão para o caso.

A geometria e condições de contorno deste escoamento estão representadas na figura 6.4:

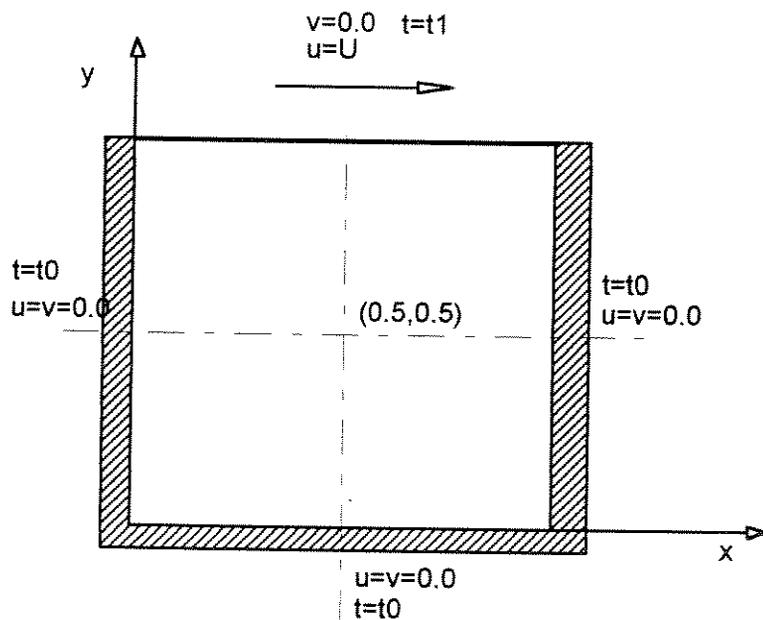


Figura 6.4: escoamento em cavidade, coordenadas e condições de contorno

Para comparação dos resultados, foram utilizadas diversas malhas, variando tanto na configuração quanto no refinamento. Todas as malhas utilizadas foram obtidas através do programa *GridGen* e em todas elas foram zeradas as funções de controle P e Q . Foram utilizadas as

equações (4.10) justificadas pelos resultados apresentados na tabela (6.1), que mostra que a solução pela equação (4.11) se torna mais demorada. As malhas estão apresentadas na figura 6.1 a seguir.

Tabela comparativa - malhas para cavidade

malha	n° de elementos	n° de nós
malha 01	16	65
malha 02	100	341
malha 03	100	341
malha 04	400	1281
malha 05	400	1281

Tabela 6.4: características das malhas - cavidade

Para efeito de comparação, foram consideradas os resultados em duas seções da cavidade, os resultados da componente u são apresentados ao longo da linha (0.5,0)(0.5,1) e os da componente v ao longo da linha (0,0.5)(1,0.5).

Foram obtidos resultados para vários números de Reynolds, referidos à velocidade da fronteira deslizante. Como o programa para aproximação das equações de Navier-Stokes, (Taylor,1981) trabalha com as equações de uma forma dimensional, e para obter um mesmo número de Reynolds para comparação dos resultados, foram consideradas propriedades para o fluido tal que variando a velocidade da fronteira superior fossem obtidos os valores de Re desejados. O fluido considerado para testes foi: ar a 50 °C, e as propriedades para este fluido são, segundo White (1991), tabela 6.5:

Propriedades do ar - 50°C

densidade ρ [kg/m ³]	viscosidade μ [μ Pa.s]	viscosidade ν [m ² /s]	condutividade térmica k [mW/mK]	calor específico C_p [J/kg.K]
1.093	19.5	0.0000178408	27.6	1006

Tabela 6.5: propriedades do ar

Com as propriedades acima, para conseguir os valores de Re , calculados em relação ao comprimento da fronteira superior, figura 6.5, deve-se então variar o valor da velocidade na fronteira superior, então para cada caso tem-se as velocidades referentes, tabela 6.6:

$$Re_l = \frac{\rho * U * l}{\mu}$$

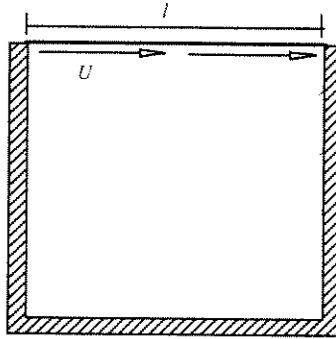


Figura 6.5: fronteira de referência para cálculo do número de Re

Velocidade U da fronteira superior

Re_l	U [m/s]
100	0.00178408051
400	0.00713632204
1000	0.01784080512

Tabela 6.6: velocidades U da fronteira superior

A seguir estão os gráficos comparativos das componentes u e v para cada Re e para cada malha utilizada.

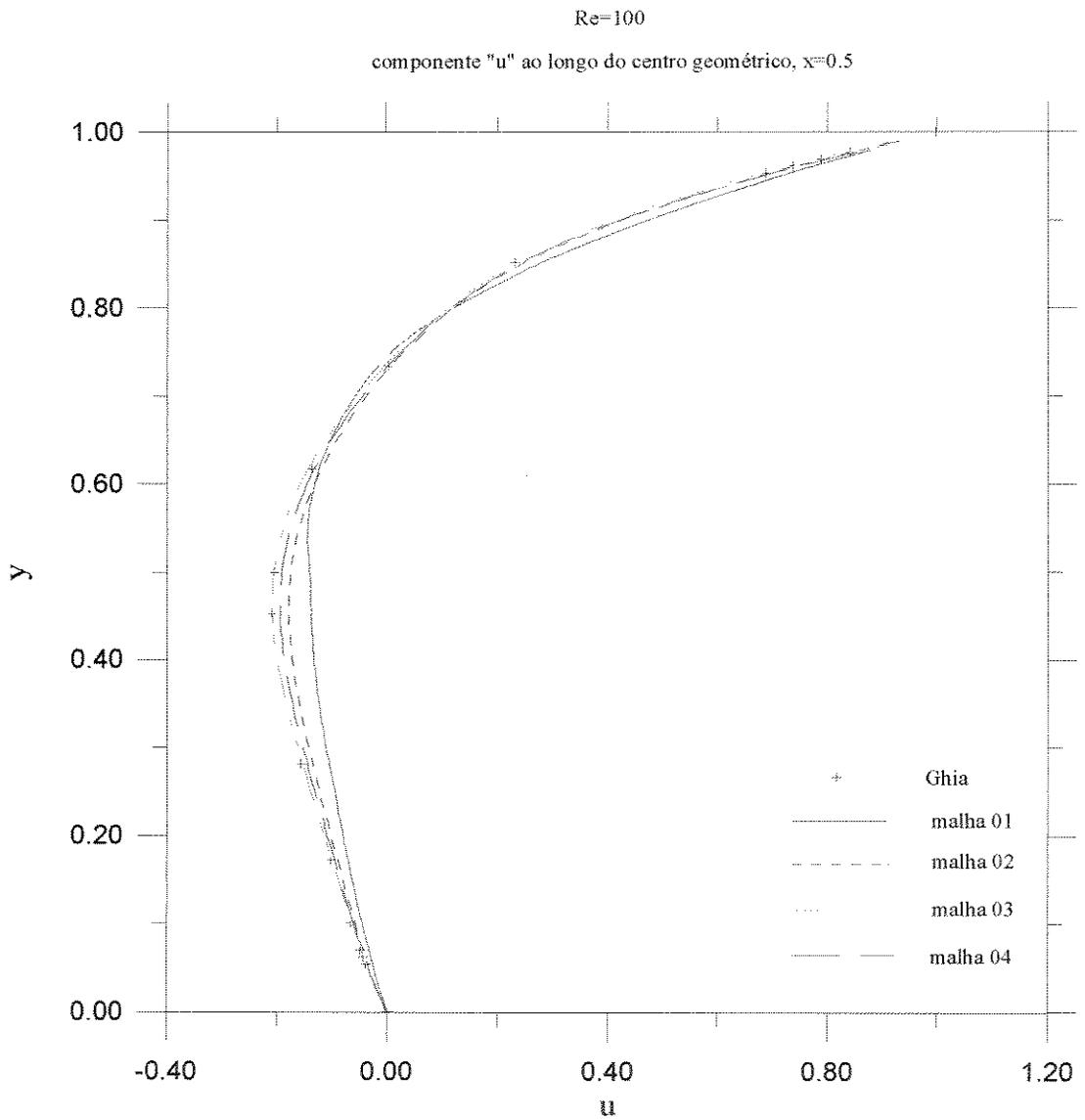


Figura 6.6: componente "u" ao longo do centro geométrico, x=0.5, Re=100

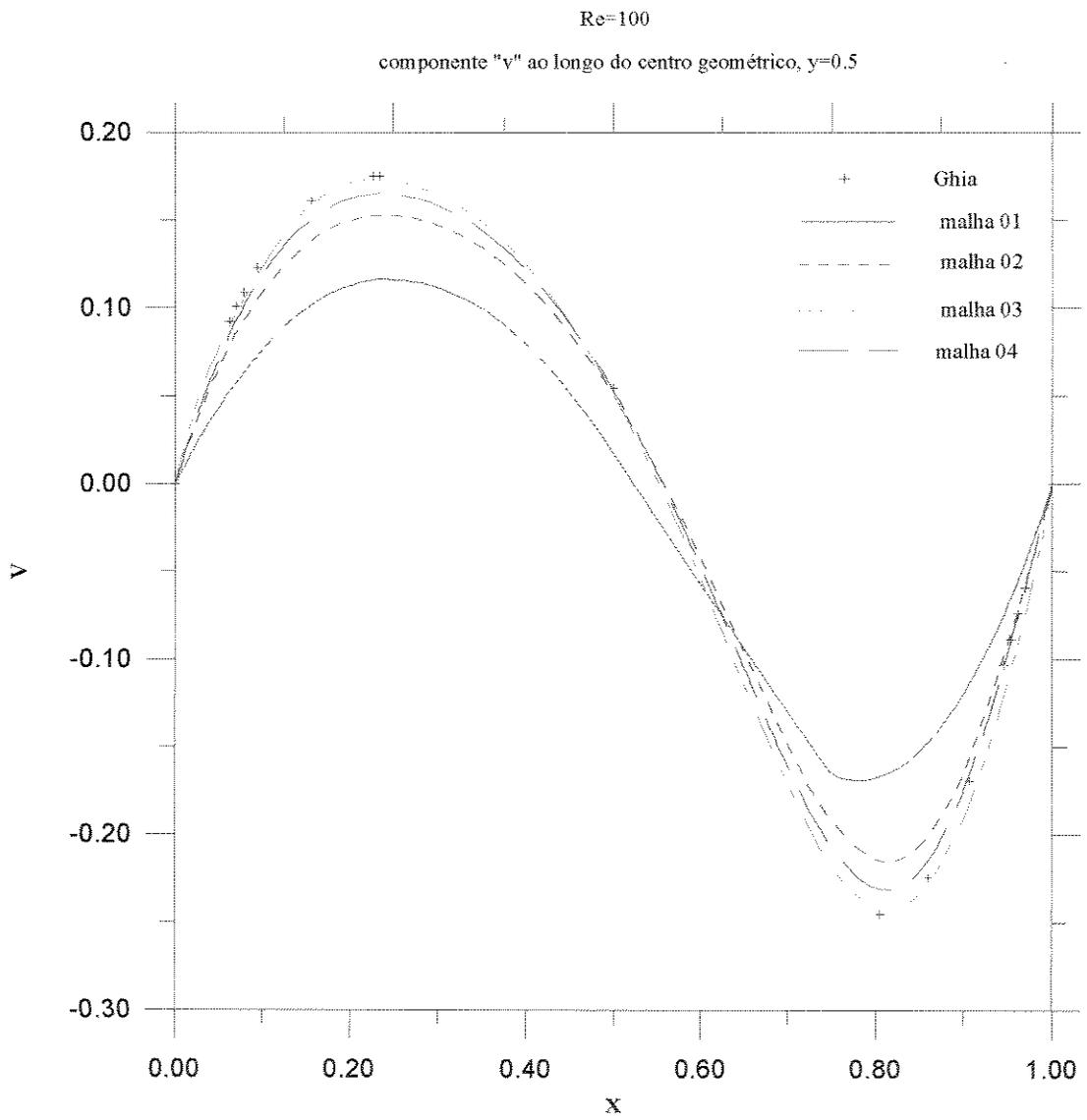
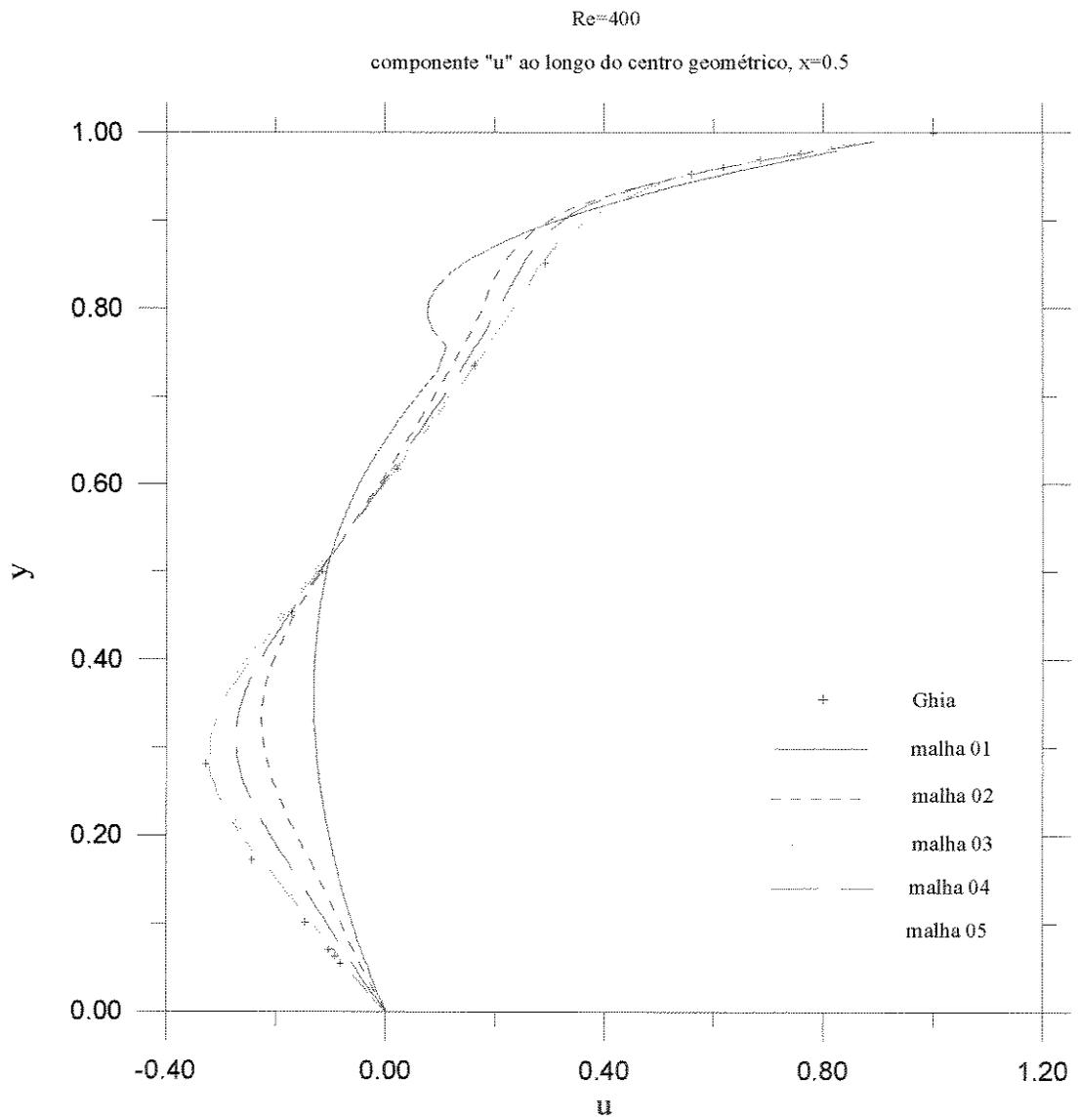


Figura 6.7: componente "v" ao longo do centro geométrico, y=0.5, Re=100

Figura 6.8: componente "u" ao longo do centro geométrico, $x=0.5$, $Re=400$

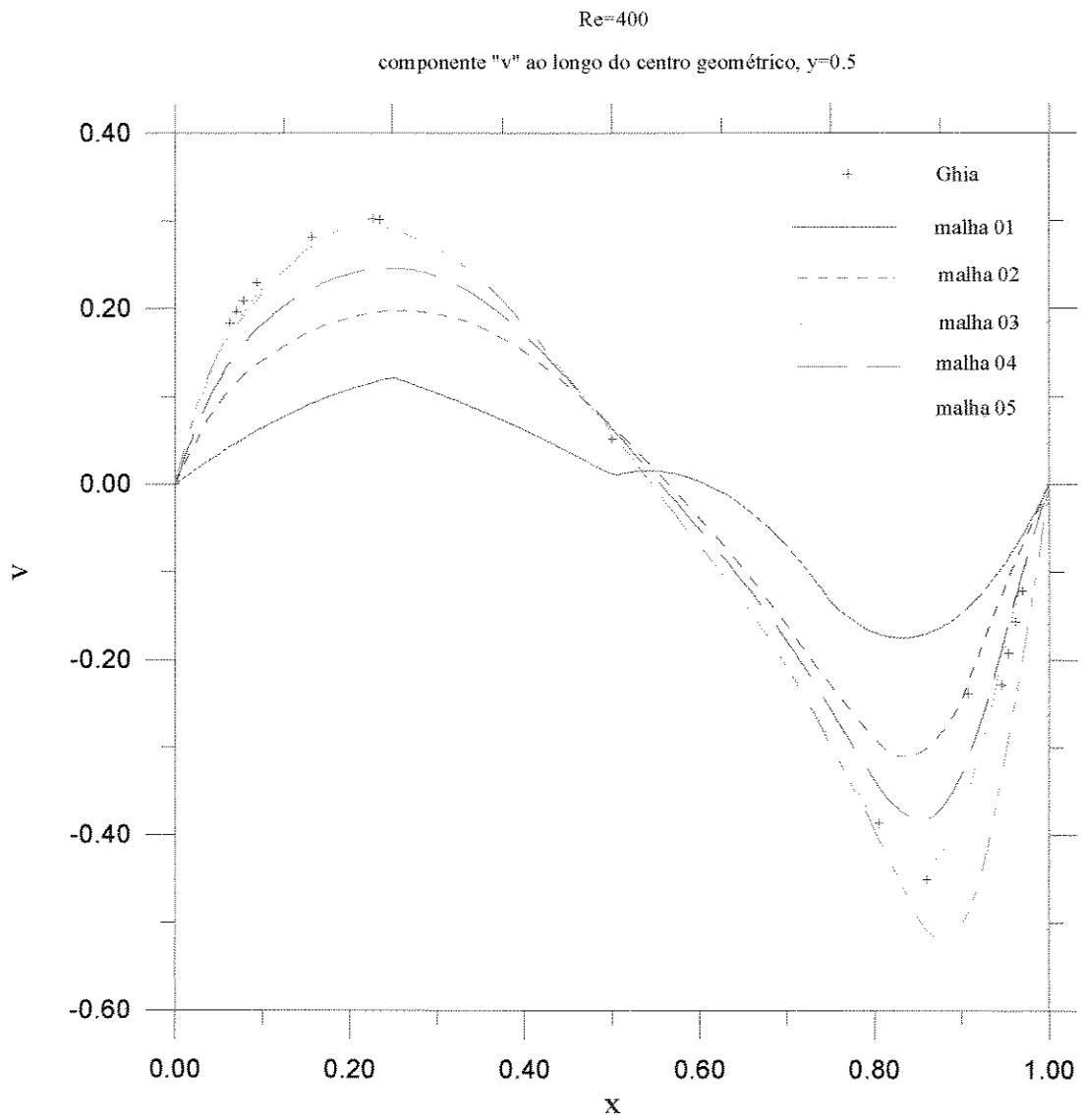
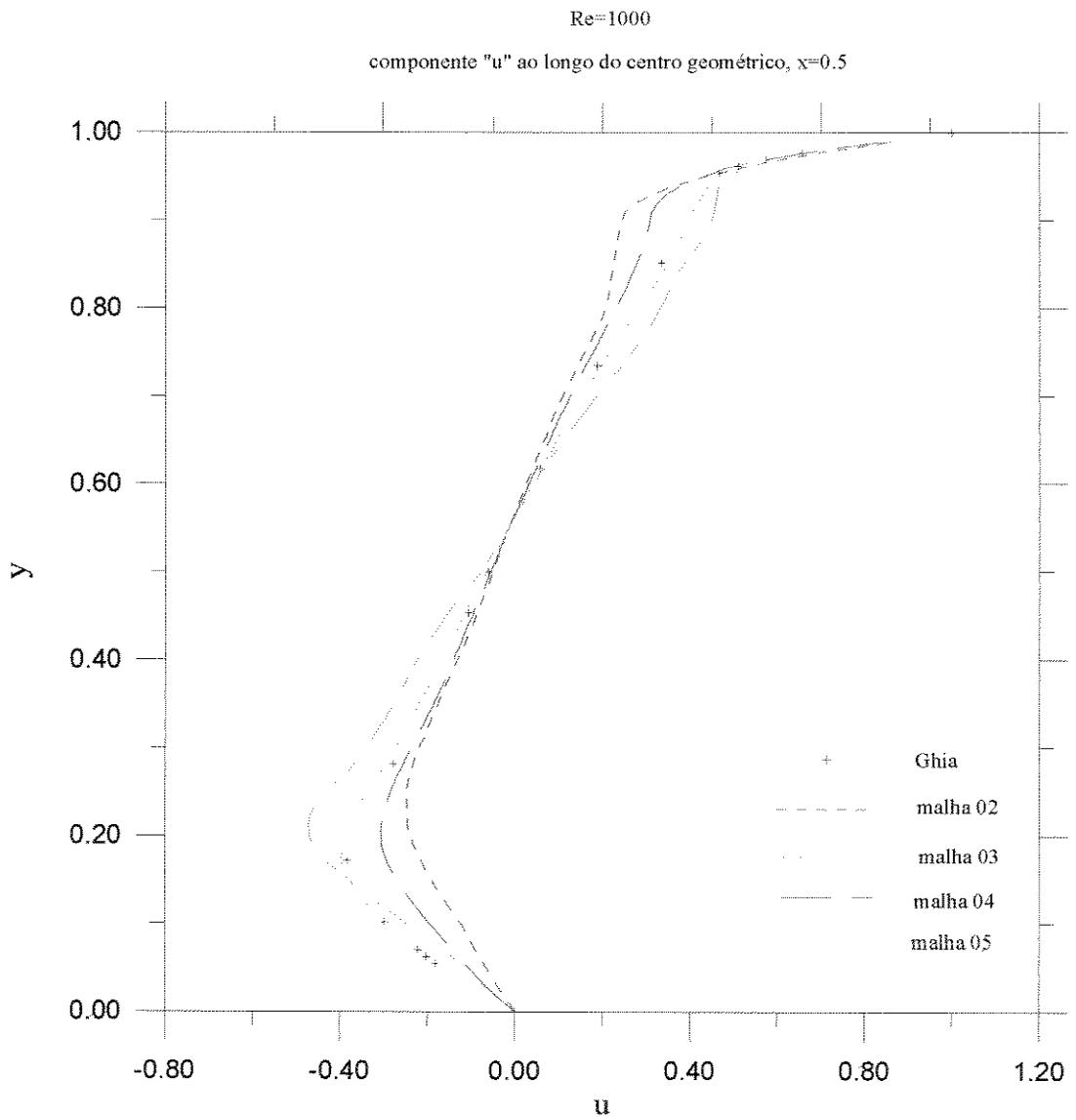


Figura 6.9: componente "v" ao longo do centro geométrico, y=0.5, Re=400

Figura 6.10: componente "u" ao longo do centro geométrico, $x=0.5$, $Re=1000$

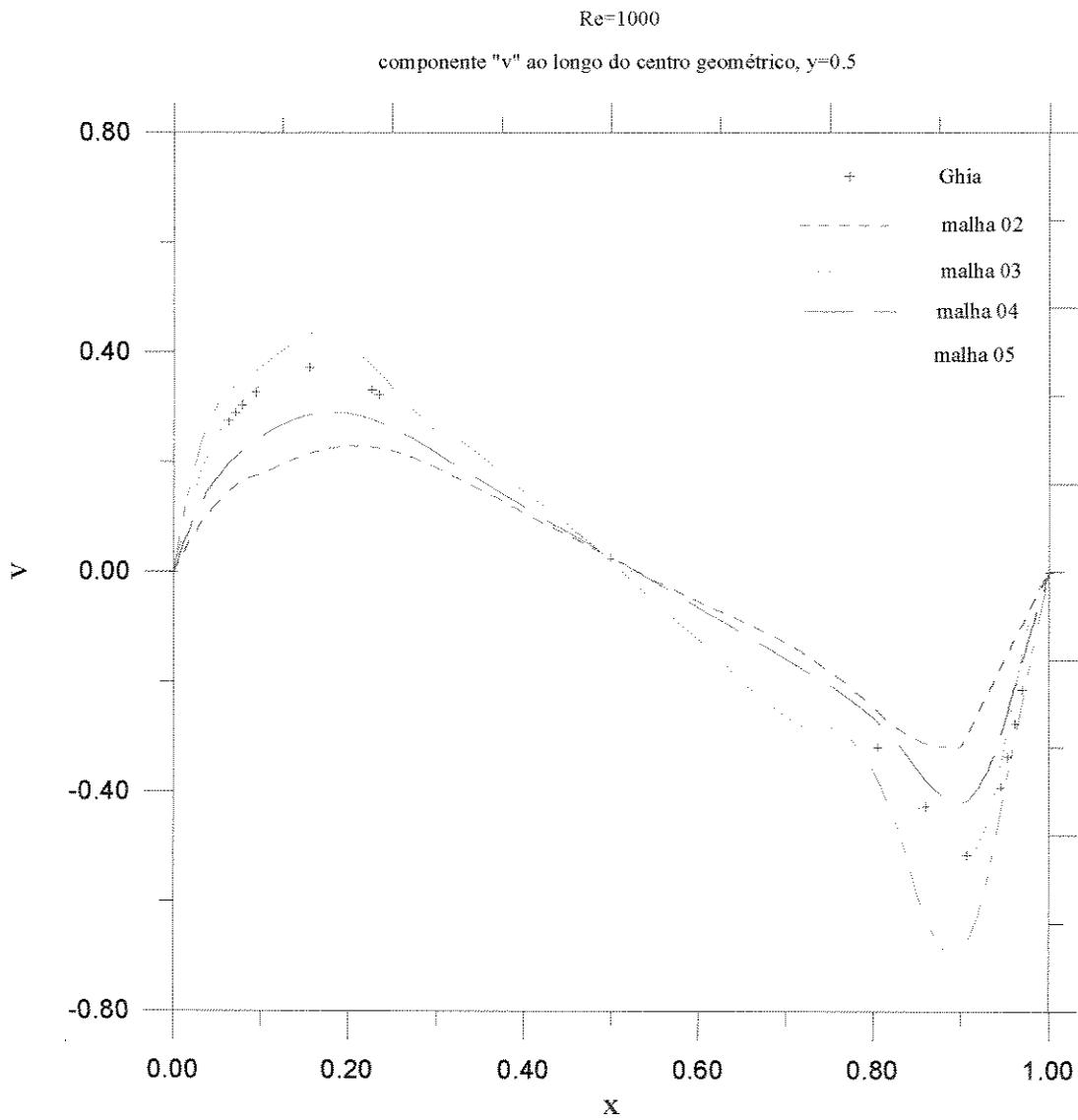


Figura 6.11: componente "v" ao longo do centro geométrico, y=0.5, Re=1000

A seguir, estão gráficos gerados pelo programa *GridGen*, que representam zonas de valores das variáveis: temperatura, velocidade, pressão; e vetores de velocidade.

A escala apresentada traz valores específicos calculados das variáveis entre as regiões. No gráfico dos vetores de velocidade, o seu tamanho está normalizado para todos os pontos, não representando assim velocidades maiores ou menores, mas dando mais atenção à direção do escoamento.

Cavidade :

malha 04

Re=1000

Pr=0.71

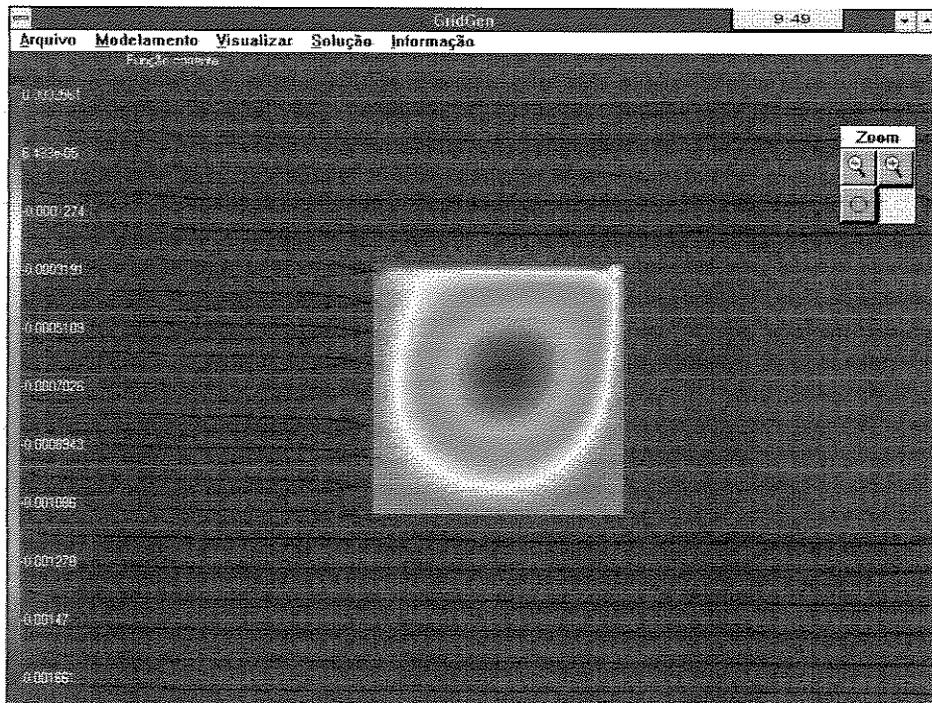


Figura 6.12: função corrente - cavidade



Figura 6.13: zonas de velocidades absolutas constantes - cavidade

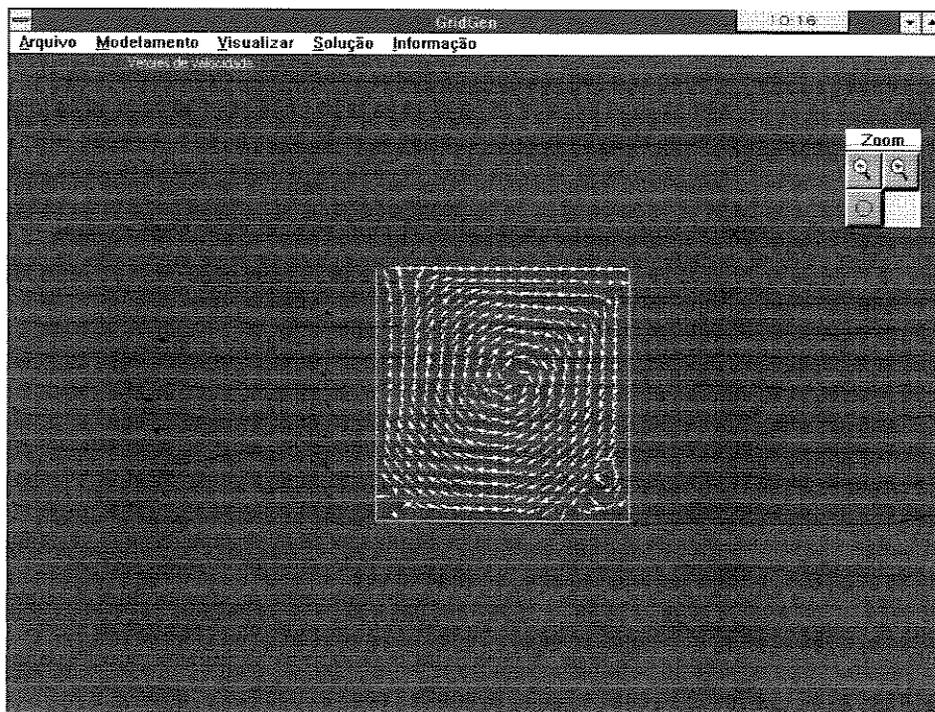


Figura 6.14: vetores de velocidade - cavidade

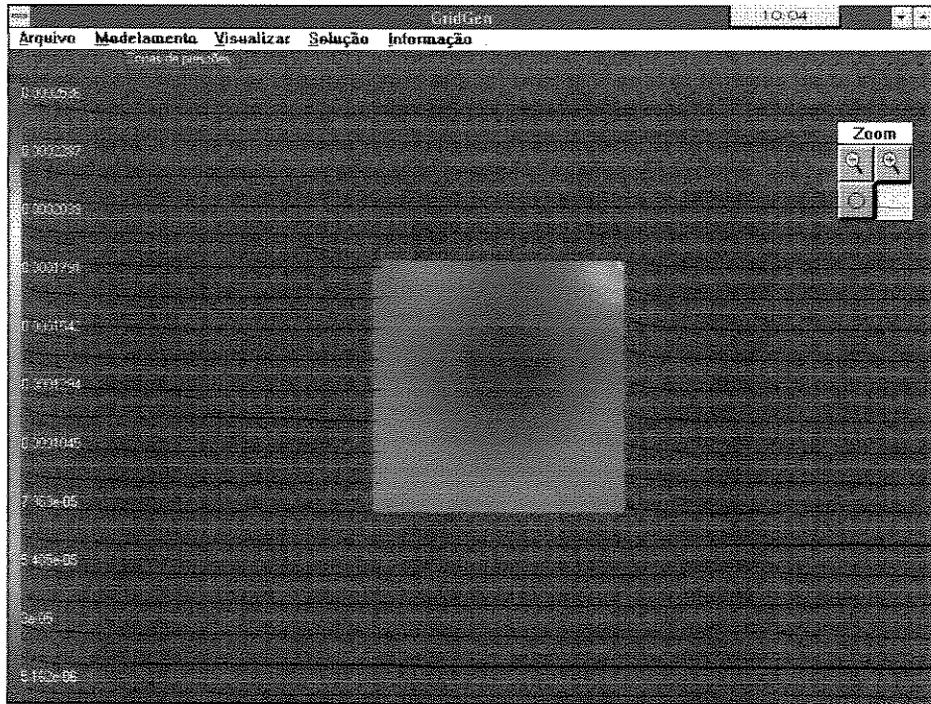


Figura 6.15: zonas de pressão constante - cavidade

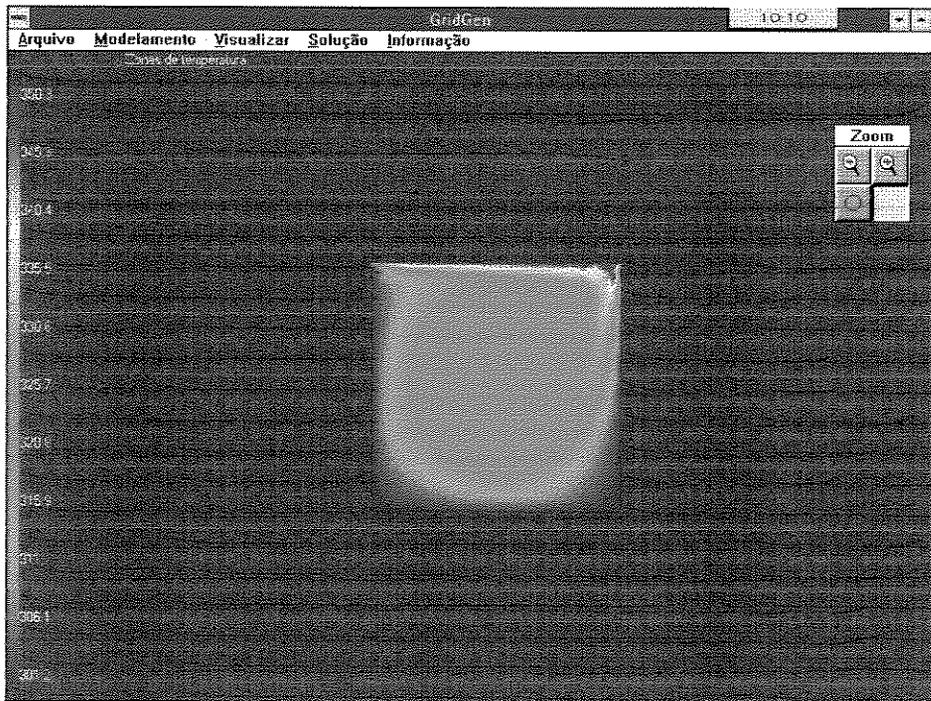


Figura 6.16: zonas de temperatura constante - cavidade

No estudo da cavidade observou-se que a qualidade dos resultados, para velocidade, aumentou com o refinamento da malha e com a deformação da malha. Foram utilizados dois tipos de malhas de configurações diferentes: com espaçamento regular, malhas 01, 02, 04; e com distribuição irregular nas fronteiras, mais refinadas próximas aos vértices, malhas 03, 05.

Foi constatado que para malhas com o mesmo número de elementos, mas com distribuição diferente, as malhas irregulares proporcionaram um melhor resultado que as regulares para valores das velocidades.

Este fato ocorre porque nas regiões próximas aos vértices e fronteiras, há uma concentração maior de elementos, de nós, e a distribuição de velocidade nestas regiões é calculada com menor erro por efeito do maior número de elementos, por isso a propagação dos efeitos viscosos do escoamento é passada com mais fidelidade aos nós, justamente nas regiões onde são maiores os gradientes de velocidade, e regiões onde há recirculação. Os melhores resultados foram obtidos com a malha 05. Como já esperado o refinamento deve ocorrer nas regiões de maiores gradientes, como foi aplicado.

Nos gráficos para $Re=100$, não foram colocados resultados para malha 05, porque não se julgou necessário já que com a malha 03 foram obtidos resultados bastante satisfatórios. Para $Re=1000$, não há resultados para malha 01 porque não houve convergência na aproximação do campo de escoamento.

Com o aumento do número de Re , a qualidade dos resultados obtidos com uma malha cai, o que torna necessário o refinamento da malha para o caso.

6-3 Escoamento em torno de um cilindro

Assim como o escoamento em cavidade, o escoamento em torno de um cilindro também tem sido utilizado como modelo para comparação de resultados para verificar a qualidade dos métodos de aproximação surgidos.

Com algumas simplificações, para baixos valores de Re , este modelo admite solução analítica, que pode ser encontrada em White (1991). Para comparação dos resultados foram utilizados dados e configurações publicados por White (1991), Dennis (1970) e Chang (1987). As dimensões e condições de contorno deste modelo estão representadas na figura 6.17.

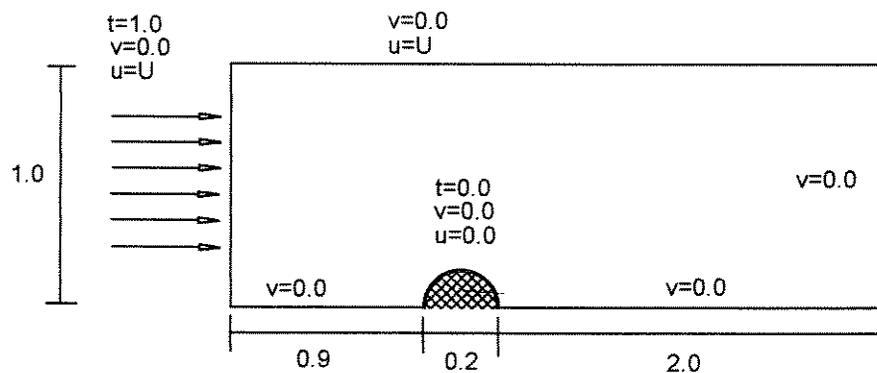


Figura 6.17: escoamento em torno de cilindro, coordenadas e condições de contorno

As malhas utilizadas para o modelo estão representadas na, figura 6.2.

Tabela comparativa - malhas para cilindro

malha	nº de elementos	nº de nós
malha 01	100	341
malha 02	400	1281
malha 03	400	1281
malha 04	400	1281
malha 05	200	661

Tabela 6.7: características das malhas - cilindro

Para a geração das malhas 04 e 05 foram utilizadas as funções P e Q para atração de linhas e de pontos.

Atração das linhas η para linha $\eta=1$ e dos pontos $\xi\eta$ para o ponto $\xi=21 \eta=1$, com fatores de decaimento:

•Função P

$$a=100 \quad b=10 \quad c=0.5 \quad d=0.5$$

•Função Q

$$a=100 \quad b=10 \quad c=0.5 \quad d=0.5$$

Para efeito de comparação dos resultados, foram analisados os comprimentos da área de recirculação no cilindro, figura 6.18, que são tabelados em White (1991) e Dennis (1970) para vários valores de Re . Neste modelo também foi utilizado o ar como fluido, e as suas propriedades estão tabeladas na tabela (6.4); já as velocidades U devem

ter valores tais que sejam obtidos os Re necessários, e estão na tabela (6.8),

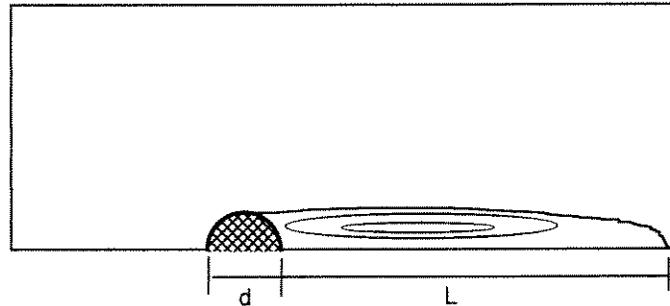


Figura 6.18: comprimento da zona de recirculação

$$Re_d = \frac{\rho * U * d}{\mu}$$

Velocidade U na fronteira esquerda

Re_d	U [m/s]
5	0.00044602
10	0.00089204
25	0.0022301
40	0.00356816

Tabela 6.8: velocidades U na fronteira esquerda

Também para comparação, foi analisada a distribuição de pressão ao longo da superfície do cilindro, figura 6.19, por meio do coeficiente de pressão.

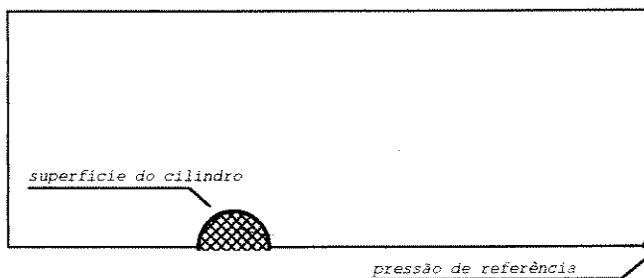


Figura 6.19 : superfície do cilindro - cálculo da pressão

$$P_{coef} = \frac{\Delta p}{0.5 * \rho * U^2}$$

onde: Δp - diferença entre pressão no ponto e pressão de referência.

ρ - densidade do fluido.

U - velocidade média de corrente.

E ainda foram comparados os números de Nu em toda a superfície do cilindro, conforme Chang (1987), figura 6.20,

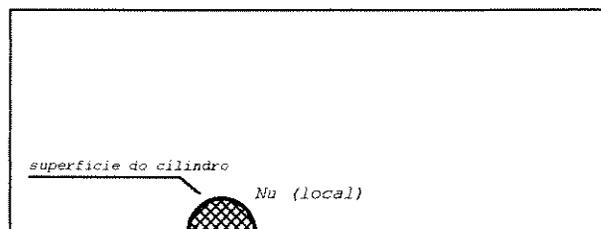


Figura 6.20: superfície do cilindro - cálculo do número de Nu

O cálculo do Nu local é dado por:

$$Nu_a = \frac{2 * (3 * \theta_1 - 4 * \theta_2 + \theta_3)}{\Delta r}$$

onde : θ são as temperaturas adimensionais

Δr é um passo adimensional na direção do raio.

e é calculado seguindo as posições dos pontos, figura 6.21,

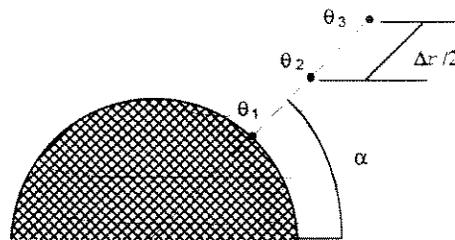


Figura 6.21: posição dos pontos para cálculo do número de Nu

A seguir estão os gráficos comparativos dos valores da razão L/d para cada Re e malha usada, dos coeficientes de pressão para cada caso e dos números de Nu locais na superfície do cilindro.

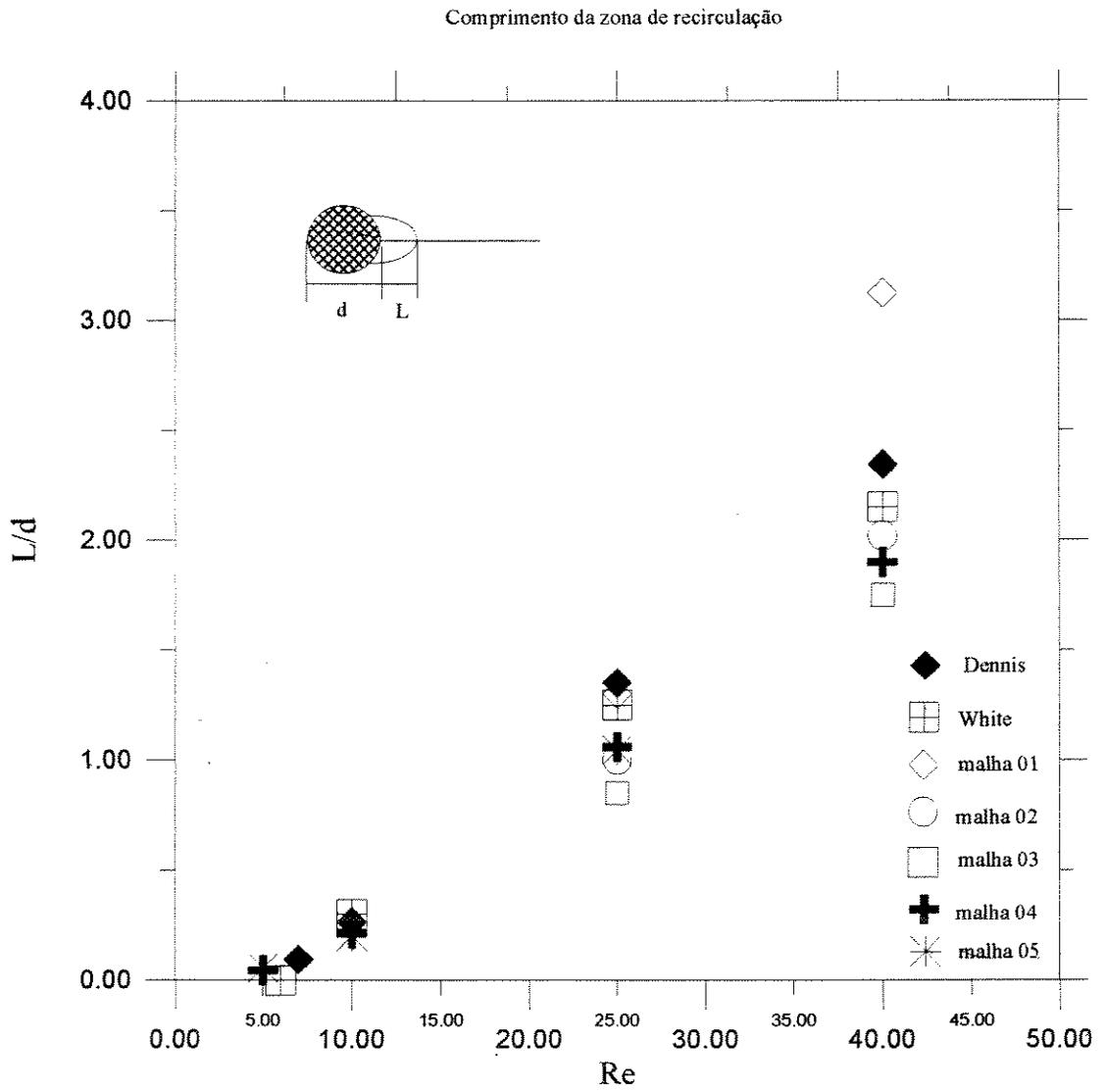


Figura 6.22: comprimento da zona de recirculação - resultados

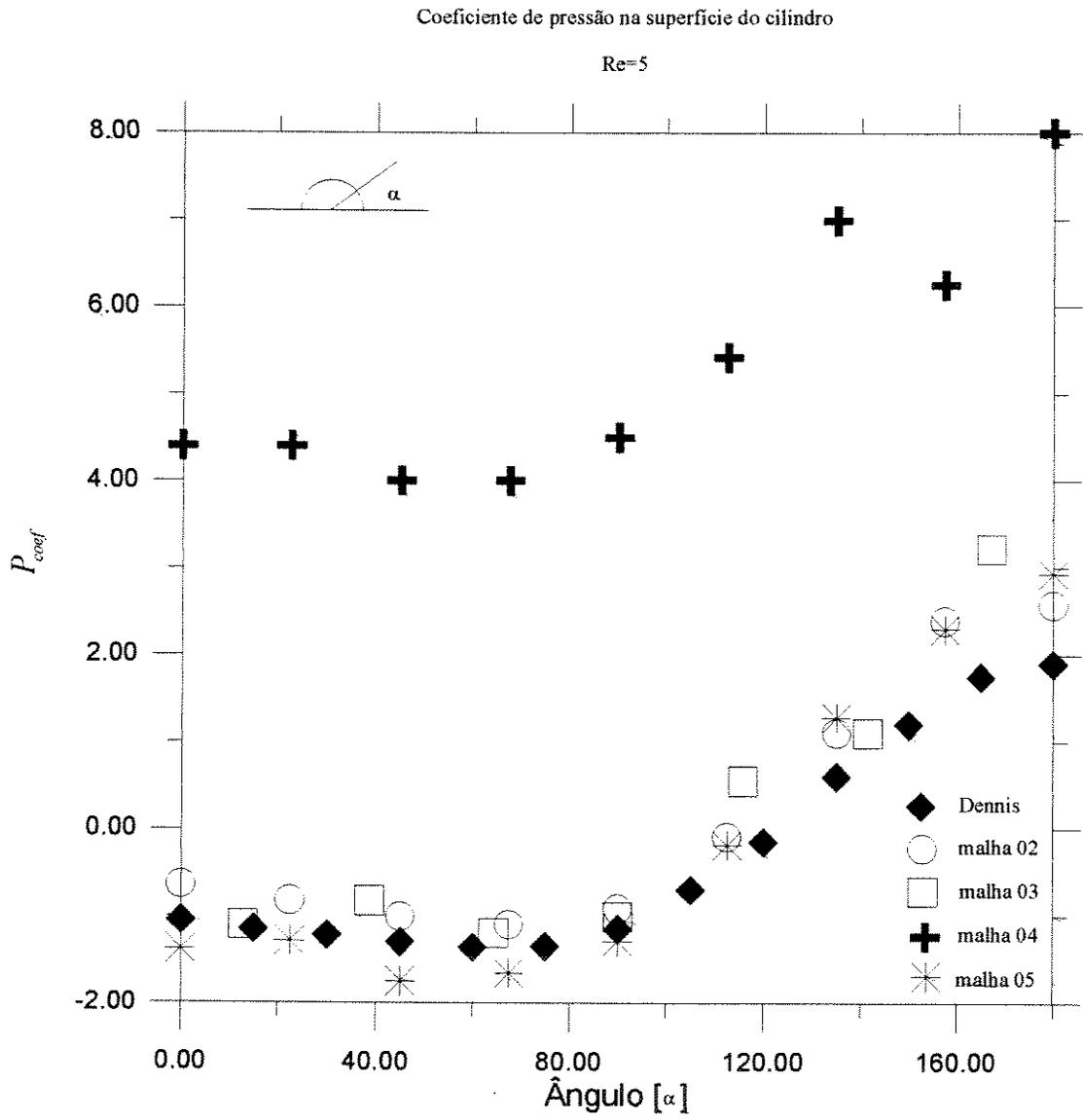


Figura 6.23: coeficiente de pressão na superfície do cilindro - Re=5

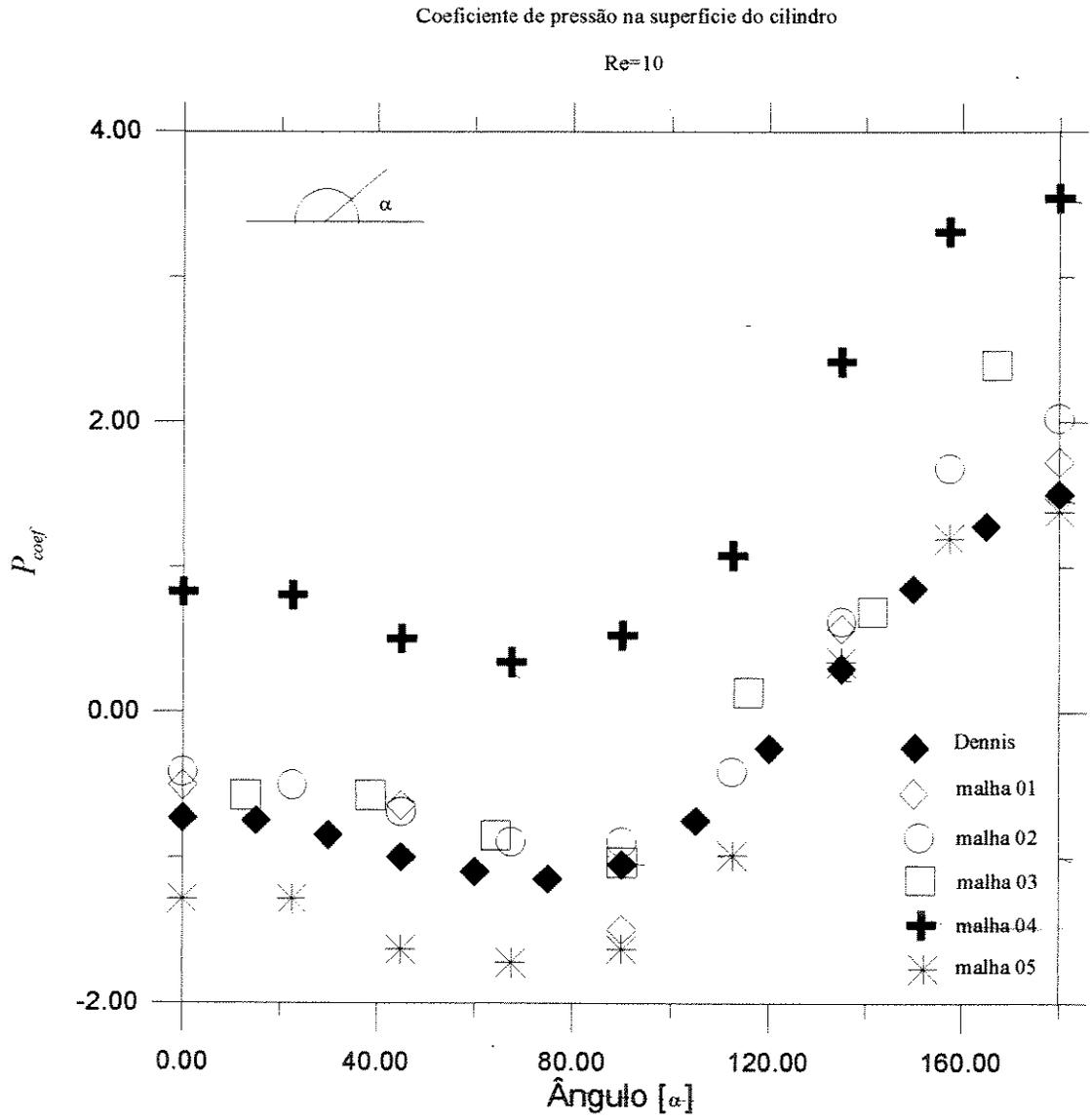


Figura 6.24: coeficiente de pressão na superfície do cilindro - Re=10

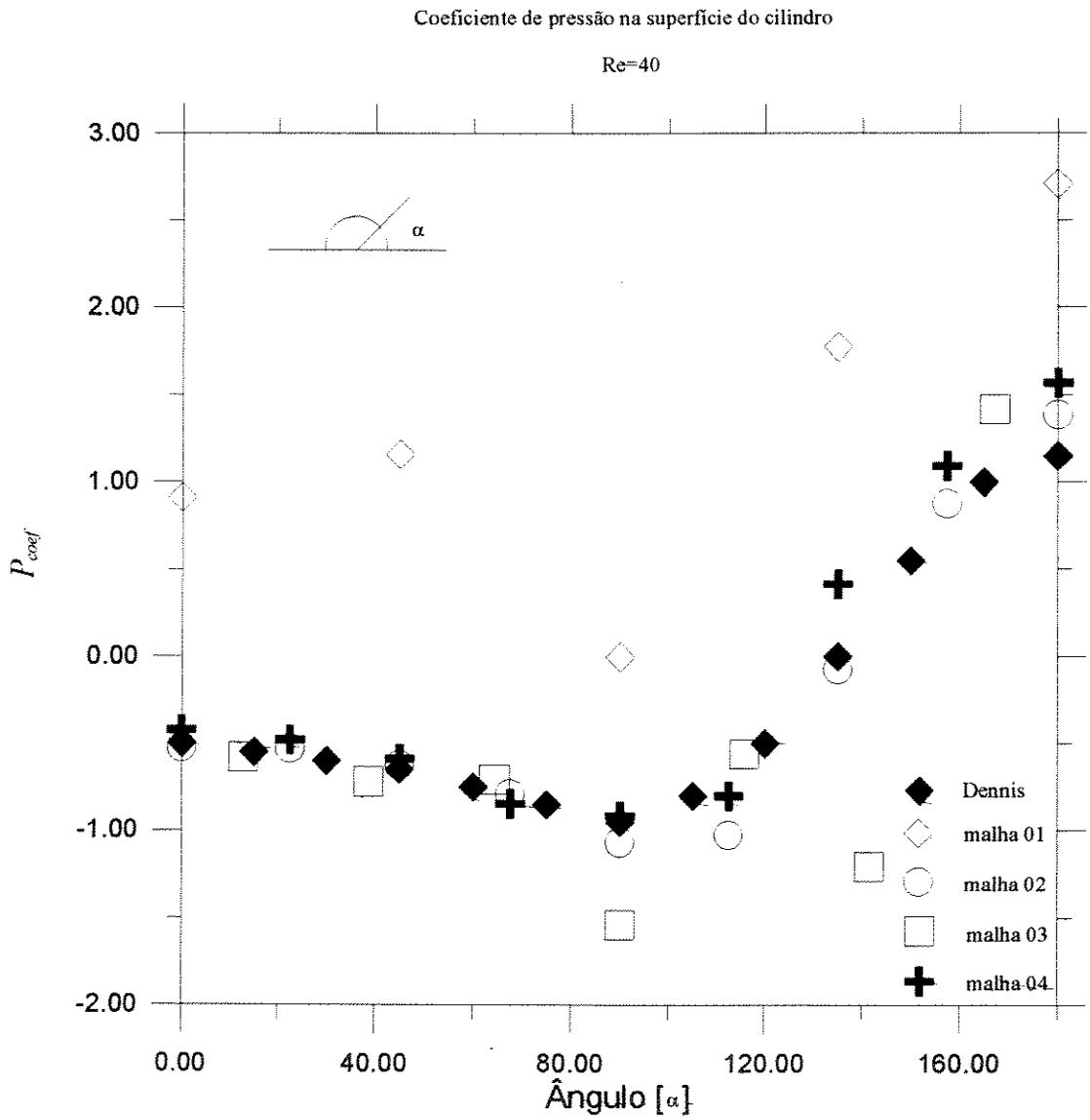


Figura 6.25: coeficiente de pressão na superfície do cilindro - Re=40

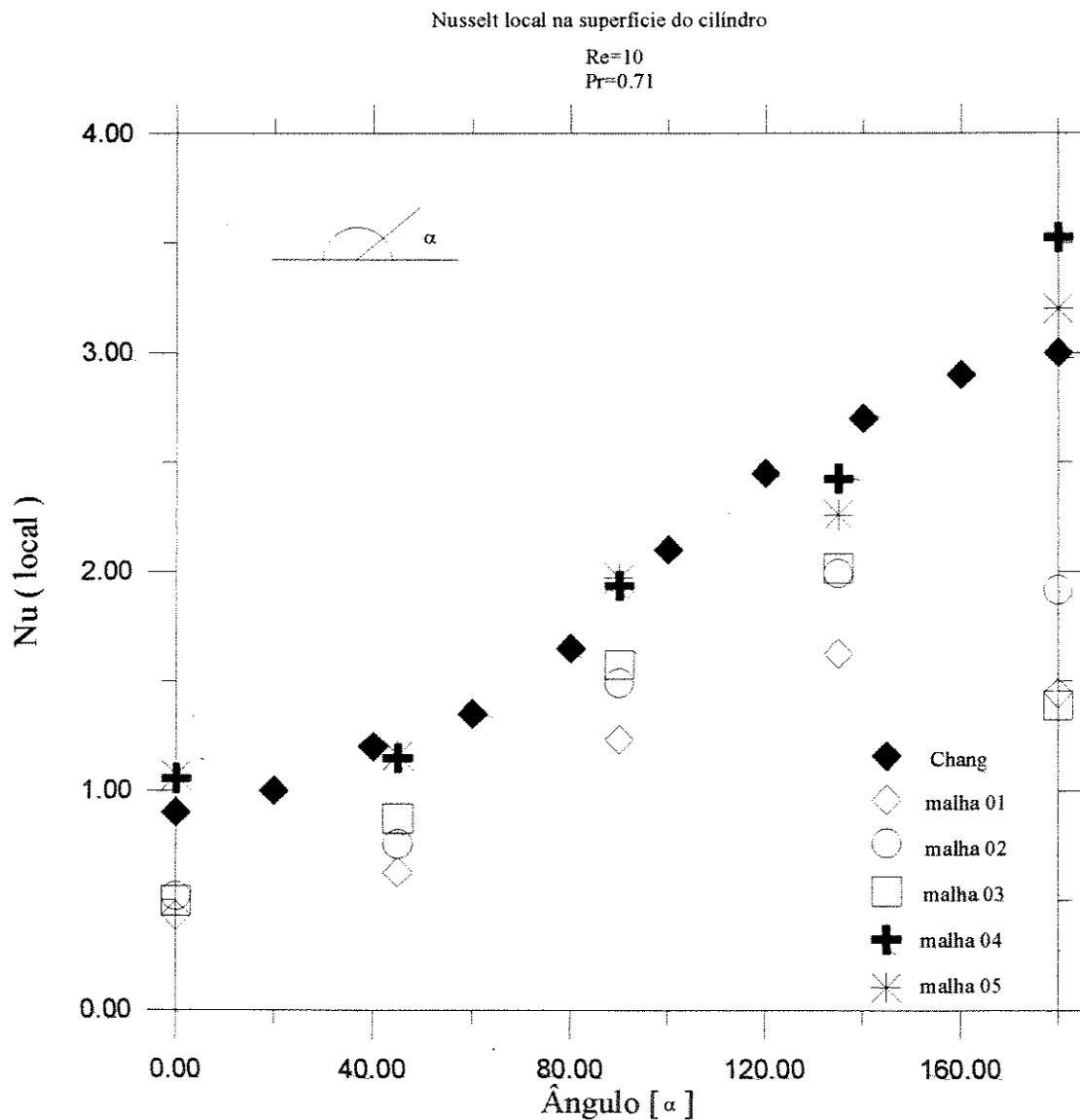


Figura 6.26: Nusselt local na superfície do cilindro - $Re=10$

A seguir, estão gráficos gerados pelo programa *GridGen*, que representam zonas de valores das variáveis: temperatura, velocidade, pressão; e vetores de velocidade.

A escala apresentada traz valores específicos calculados das variáveis entre as regiões. Nos vetores de velocidade, o seu tamanho está normalizado para todos os pontos, não representando assim velocidades maiores ou menores, mas dando mais atenção à direção.

Cilindro :

malha 04

Re=40

Pr=0.71

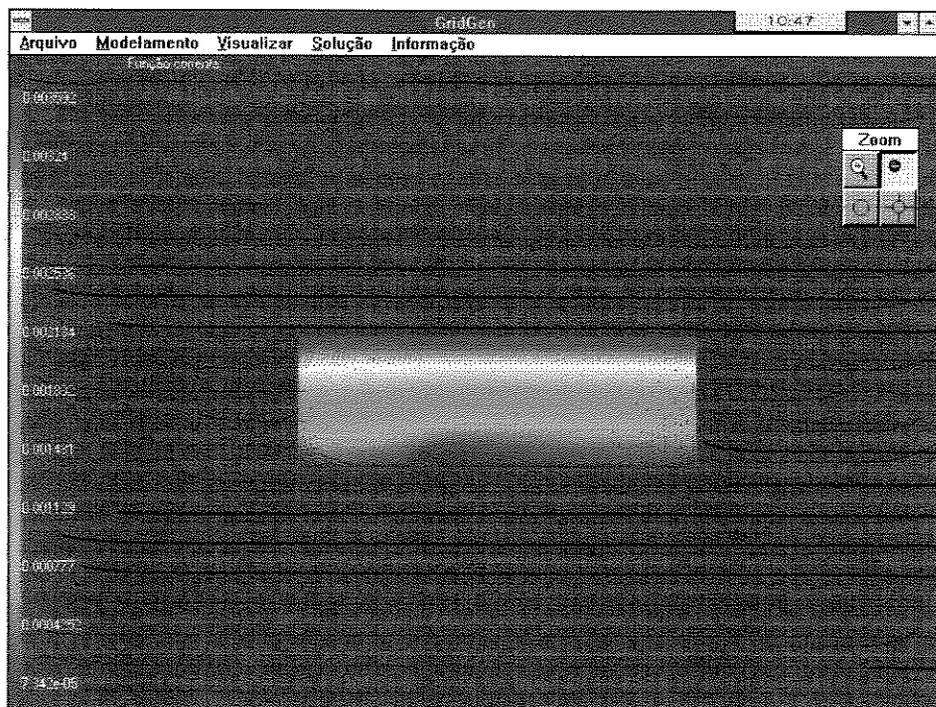


Figura 6.27: função corrente - cilindro

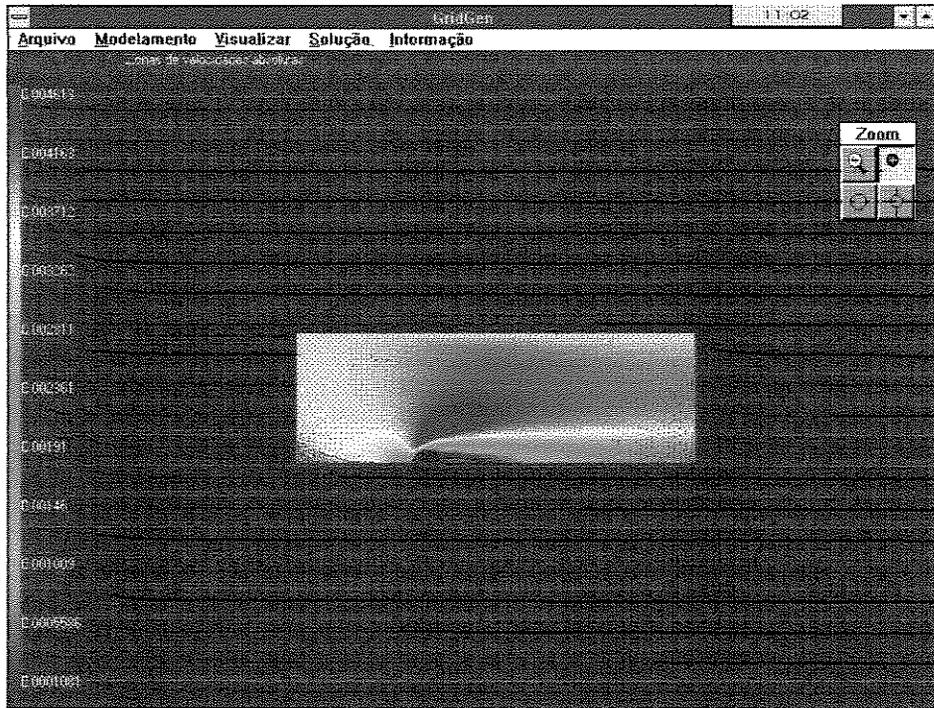


Figura 6.28: zonas de velocidades absolutas constantes - cilindro

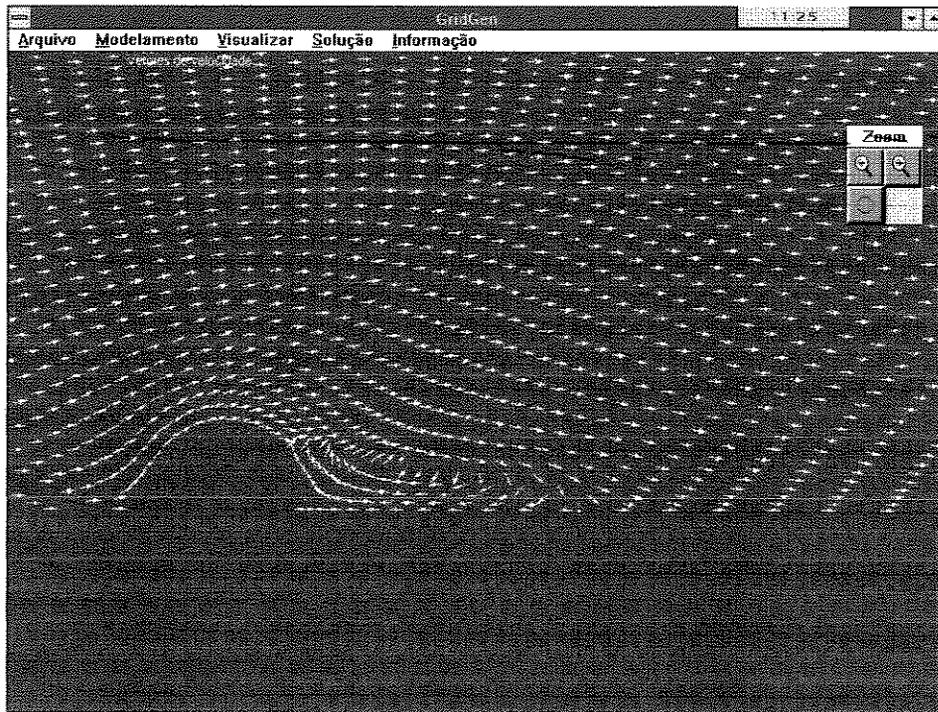


Figura 6.29: vetores de velocidade (detalhe) - cilindro

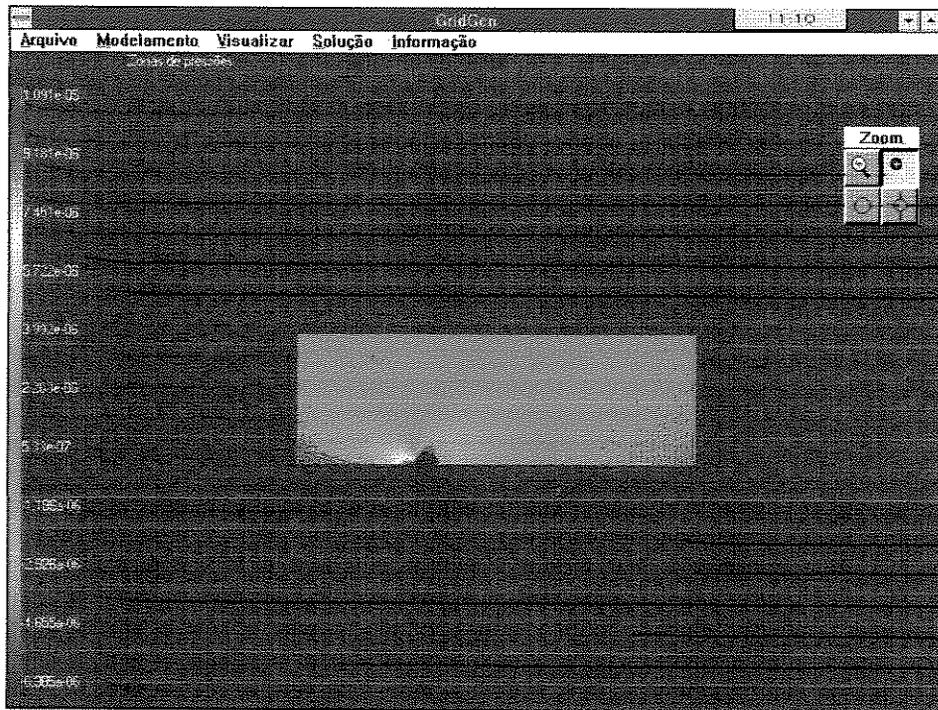


Figura 6.30: zonas de pressão constante - cilindro

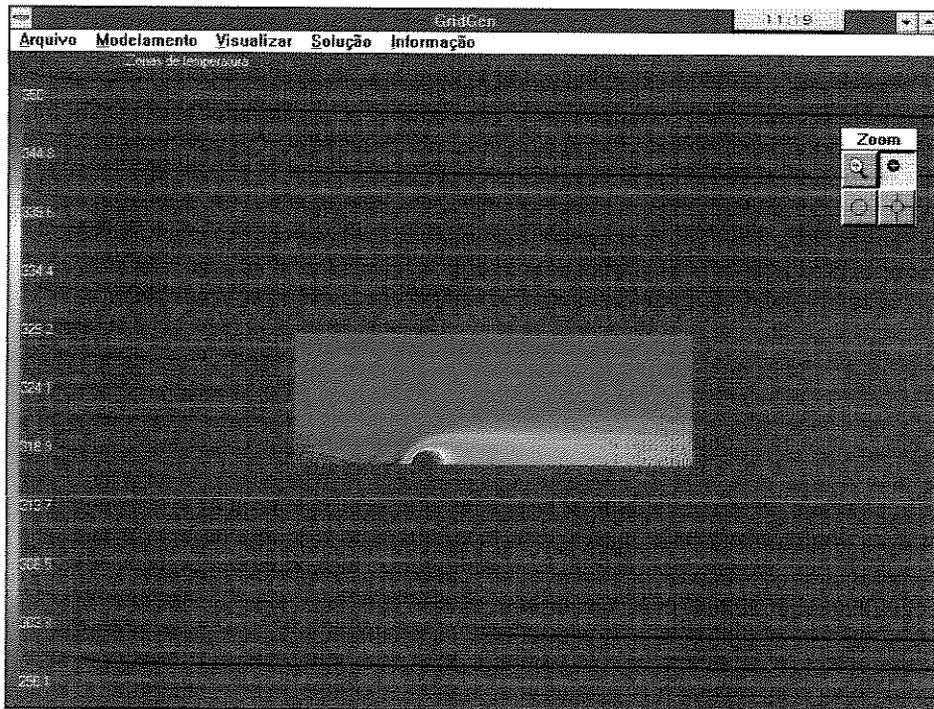


Figura 6.31: zonas de temperatura constante - cilindro

No caso do escoamento em torno de um cilindro, foram comparados três parâmetros: comprimento da zona de recirculação, coeficiente de pressão, número de Nu.

Para a zona de recirculação observou-se que para algumas malhas, 01, 02, 03, não foi possível a detecção do comprimento desta zona para $Re \approx 10$, valor médio conseguido pelos autores citados para comparação. Apenas com a aproximação das linhas da malha para a superfície do cilindro e linha de simetria, malhas 04, 05, foi possível a detecção da zona de recirculação. Este fato ocorre porque para as malhas 01, 02, 03, não há nós dentro da zona, então não há como obter velocidades características da recirculação, com a aproximação das linhas aparecem nós dentro da zona então há possibilidade de detecção da recirculação, figura 6.32.

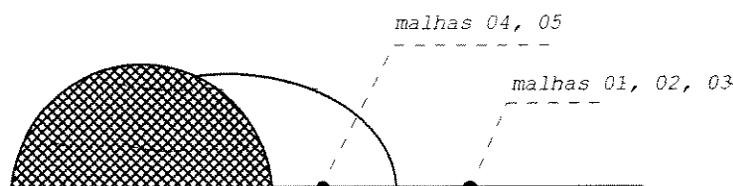


Figura 6.32: posição de nós na zona de recirculação - cilindro

Em relação ao coeficiente de pressão, a malha que teve um comportamento mais regular foi a malha 02, apresentando resultados bons para todos os Re testados, e a malha 05 apresentou melhora nos resultados de acordo com o aumento do número de Re . Mas em todas as malhas o comportamento geral foi apresentando uma região de maior pressão na face do cilindro frontal ao escoamento, 180° .

No cálculo do número de Nusselt, as malhas com aproximação à superfície do cilindro apresentaram resultados melhores, mostrando a importância da distribuição dos nós da malha também para o cálculo da distribuição da temperatura.

De um modo geral, para maiores Re a malha 04 mostrou-se mais confiável.

6-4 Escoamento em uma expansão abrupta

Ainda como modelo de comparação, foi usado o modelo de escoamento em uma expansão abrupta, onde de modo semelhante ao escoamento em torno de um cilindro, o comprimento da zona de recirculação é um parâmetro importante a determinar. Como fonte de resultados para comparação foram utilizados os trabalhos publicados por Atkins (1980), referentes a aproximação numérica das equações governantes, e por Denham (1974), a medições experimentais.

A geometria e as condições para o modelo estão representadas na figura 6.33, correspondendo àquelas empregadas por Atkins (1980) e Taylor (1981).

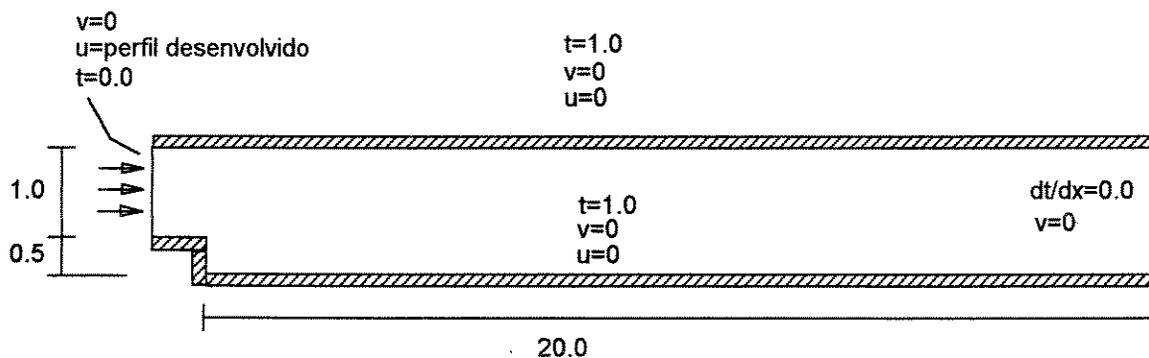


Figura 6.33 : escoamento em uma expansão, coordenadas e condições de contorno

As malhas utilizadas para o modelo estão representadas na figura 6.3.

Tabela comparativa - malhas para expansão

malha	n° de elementos	n° de nós
malha 01	100	341
malha 02	400	1281
malha 03	100	341
malha 04	400	1281

Tabela 6.9: características das malhas - expansão

O comprimento da zona de recirculação é medido segundo a figura 6.34. Neste modelo também para as equações de Navier-Stokes, foram utilizadas as propriedades do ar, já tabeladas, tabela (6.5). O número de Re é medido com referência ao comprimento h do salto e uma velocidade média U resultante do perfil desenvolvido de entrada, como sugere Atkins (1980).

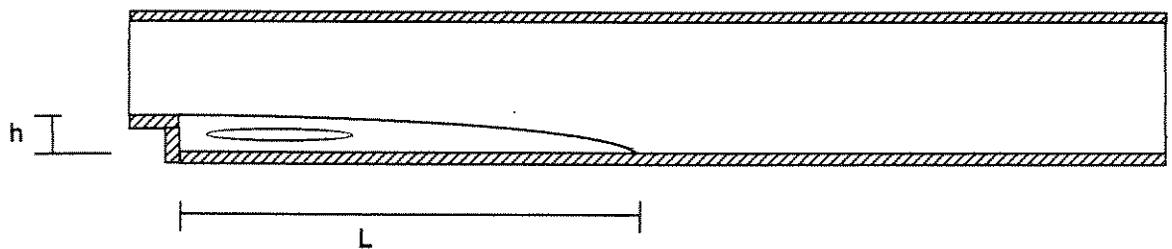


Figura 6.34: comprimento da zona de recirculação

Para cada valor de Re , obtem-se as velocidades médias, tabela 6.10.

$$Re_h = \frac{\rho^* U^* h}{\mu}$$

Velocidade U média

Re_h	U [m/s]
50	0.00178408
100	0.003568
150	0.005352

Tabela 6.10: velocidades U médias

Como no caso do cilindro, ainda foi analisada a distribuição de pressão ao longo da linha de referência, figura 6.35, por meio do coeficiente de pressão.



Figura 6.35 : posição da linha de referência

$$P_{coef} = \frac{\Delta p}{0.5 * \rho^* U^2}$$

onde: Δp - diferença entre pressão no ponto e pressão de referência.

ρ - densidade do fluido.

U - velocidade média de corrente.

A seguir estão os gráficos comparativos dos valores da razão L/h para cada Re e malha usada. Para o caso estudado, apenas a figura 6.36 apresenta comparações entre o presente estudo e estudos já publicados, porque nos estudos de Atkins (1980) e Denham(1974), que foram tomados como referência, não há todos os resultados para o regime laminar.

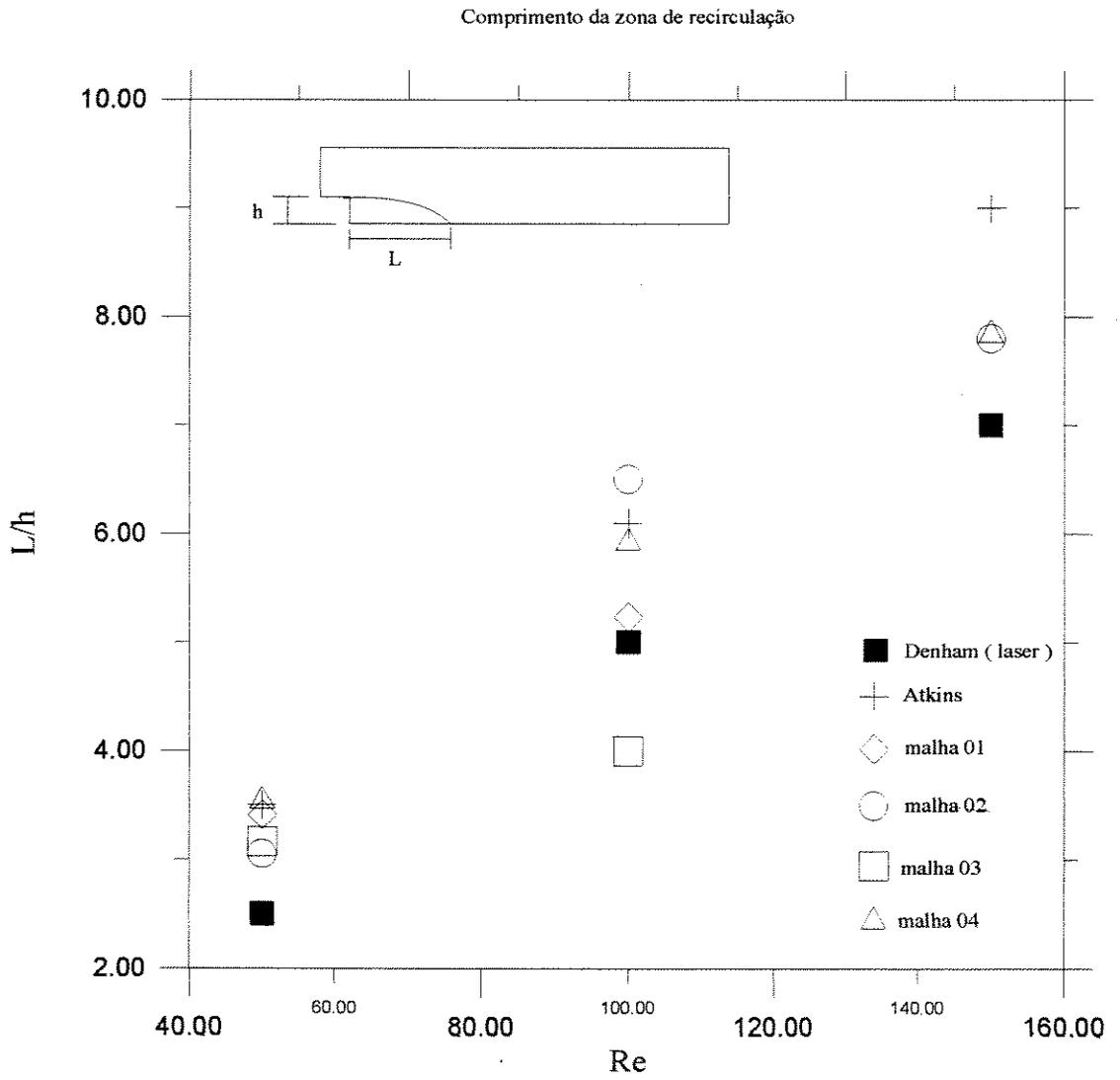


Figura 6.36: comprimento da zona de recirculação - resultados

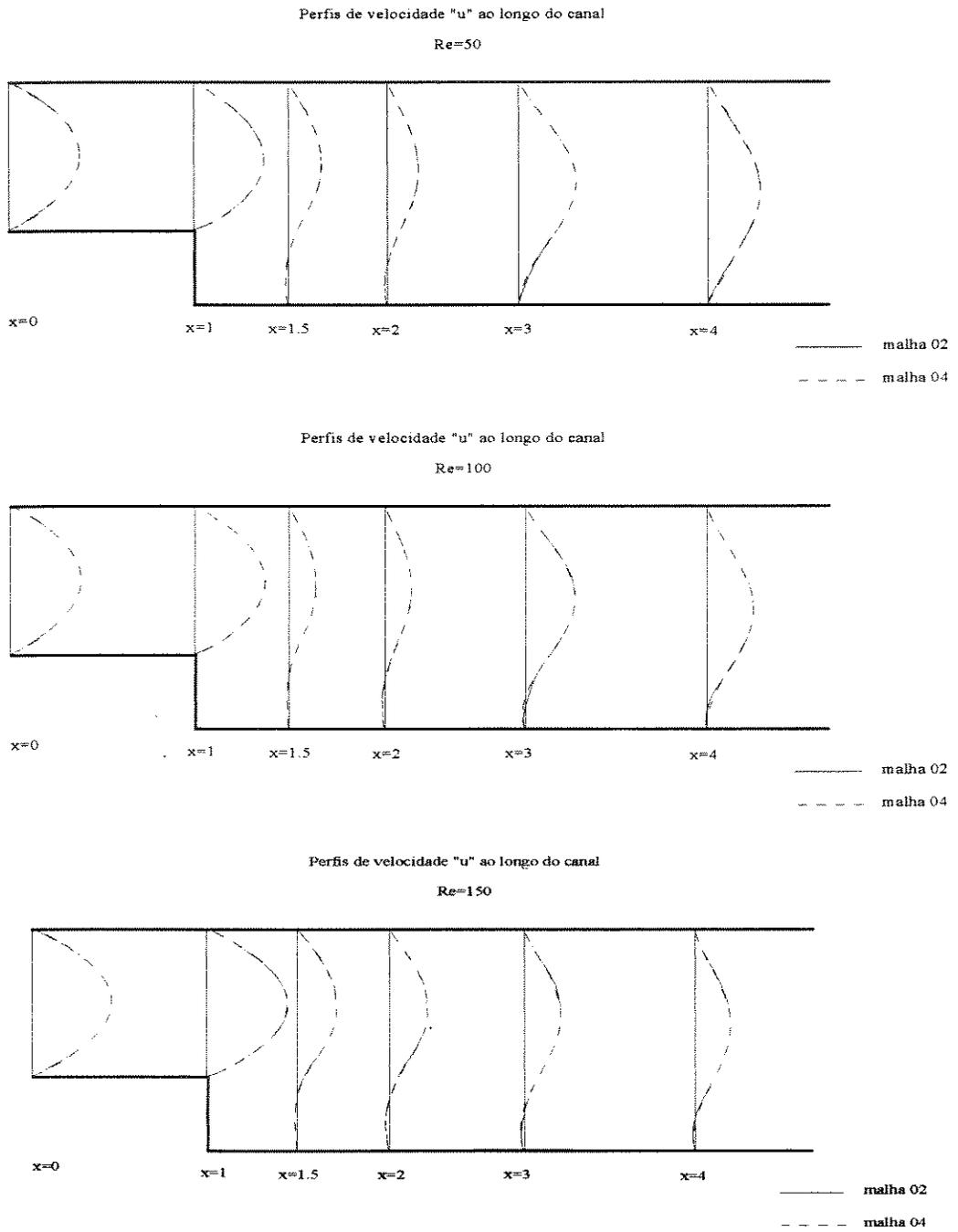


Figura 6.37: perfis de velocidade u ao longo do canal, $Re=50$; $Re=100$; $Re=150$

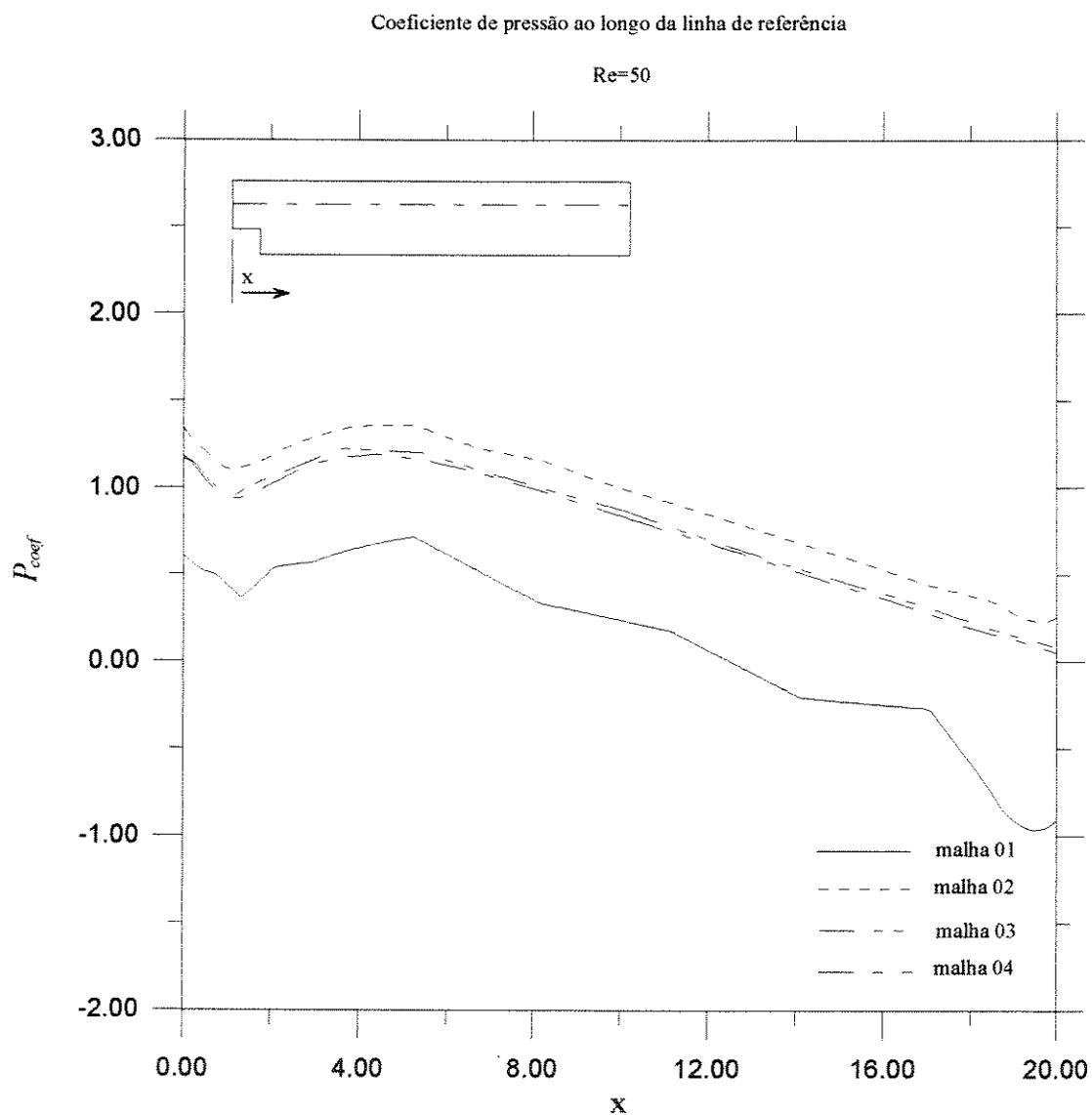


Figura 6.38: coeficiente de pressão ao longo da linha de referência - Re=50

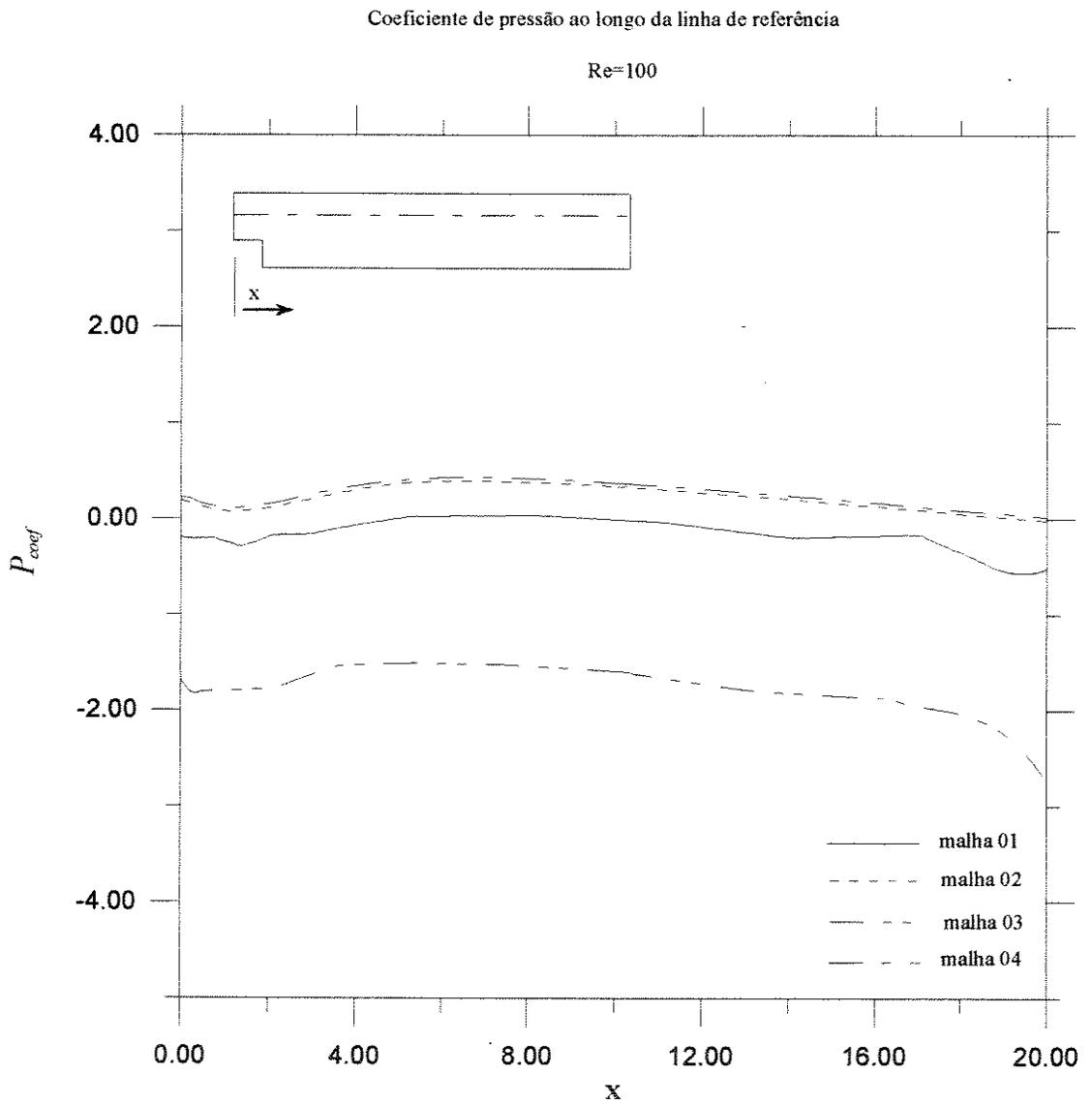


Figura 6.39: coeficiente de pressão ao longo da linha de referência - Re=100

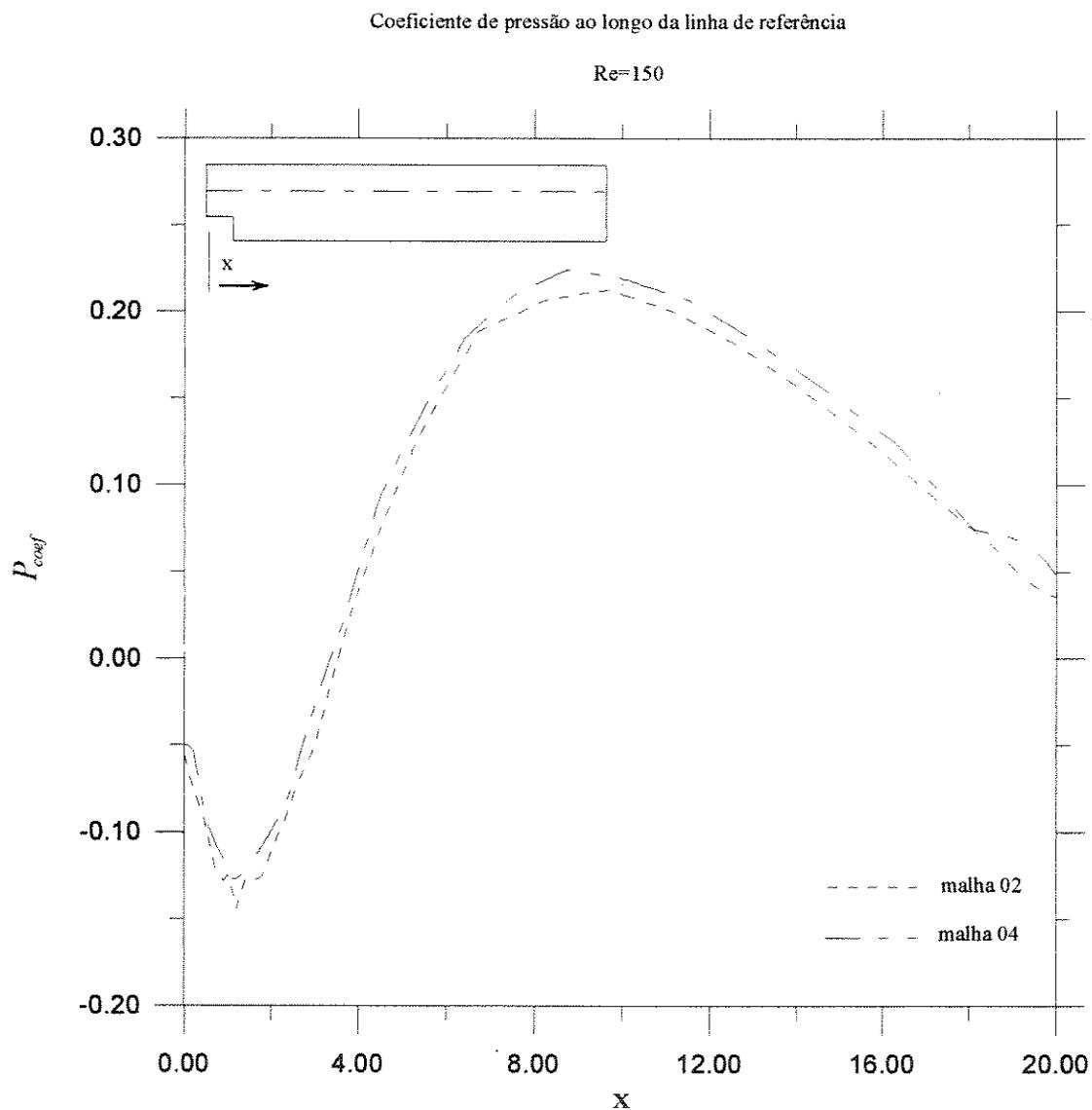


Figura 6.40: coeficiente de pressão ao longo da linha de referência - Re=150

A seguir, estão gráficos gerados pelo programa *GridGen*, que representam zonas de valores das variáveis: temperatura, velocidade, pressão; e vetores de velocidade.

A escala apresentada traz valores específicos calculados das variáveis entre as regiões. Nos vetores de velocidade, o seu tamanho está normalizado para todos os pontos, não representando assim velocidades maiores ou menores, mas dando mais atenção à direção.

Expansão :

malha 04

Re=150

Pr=0.71

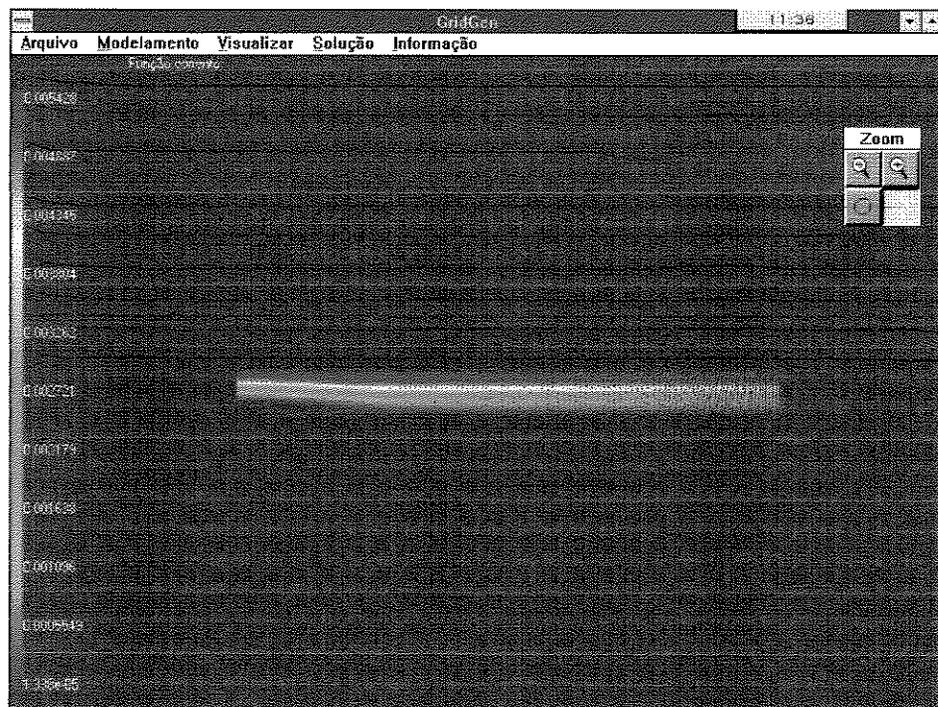


Figura 6.41: função corrente - expansão

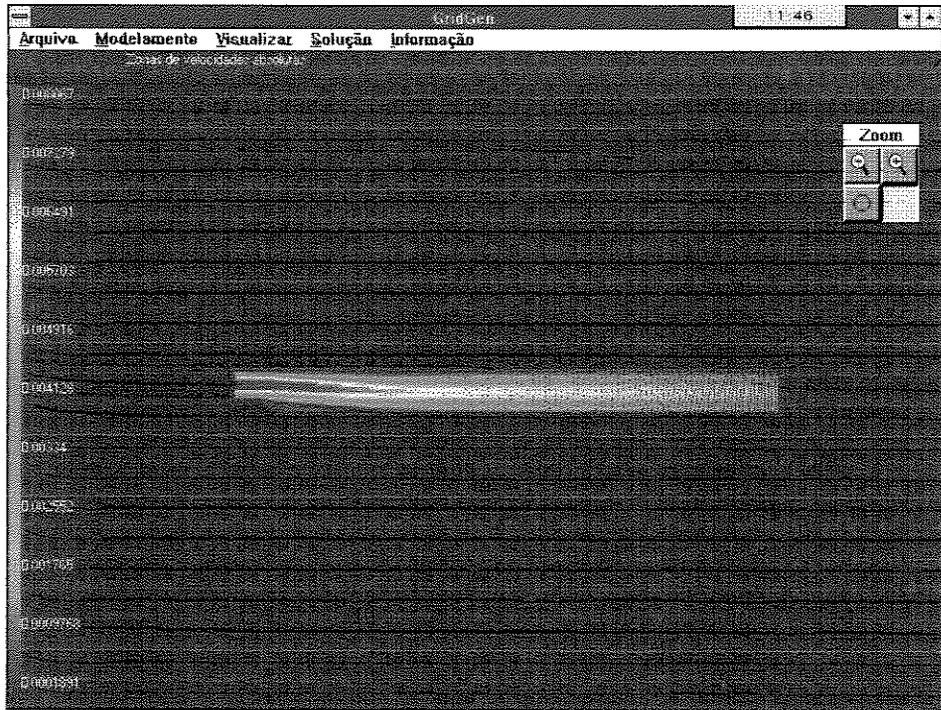


Figura 6.42: zonas de velocidades absolutas constantes - expansão

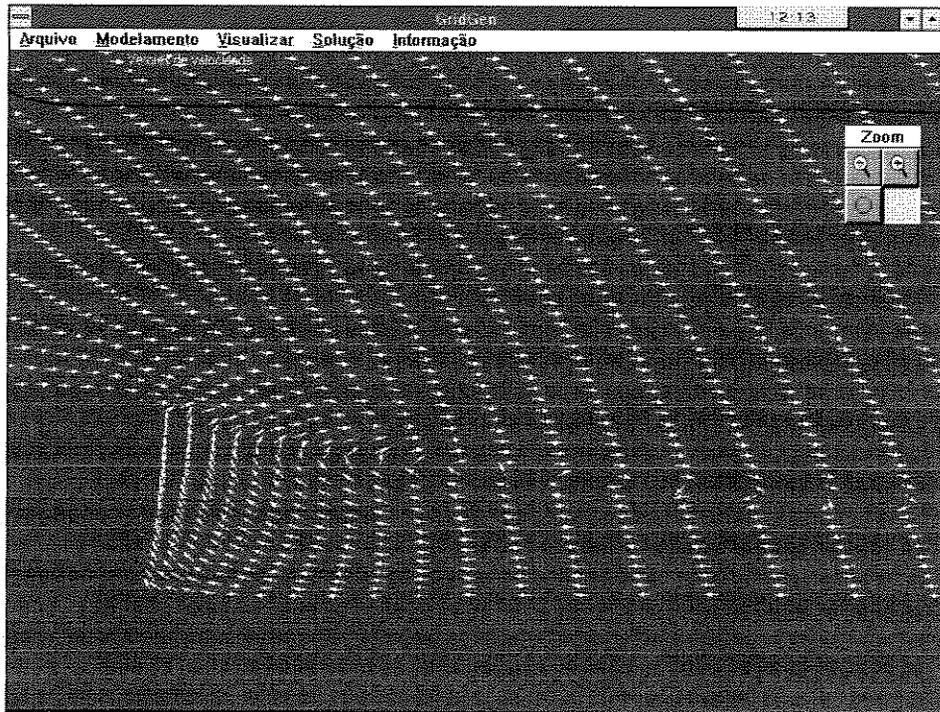


Figura 6.43: vetores de velocidade (detalhe) - expansão



Figura 6.44: zonas pressão constante - expansão

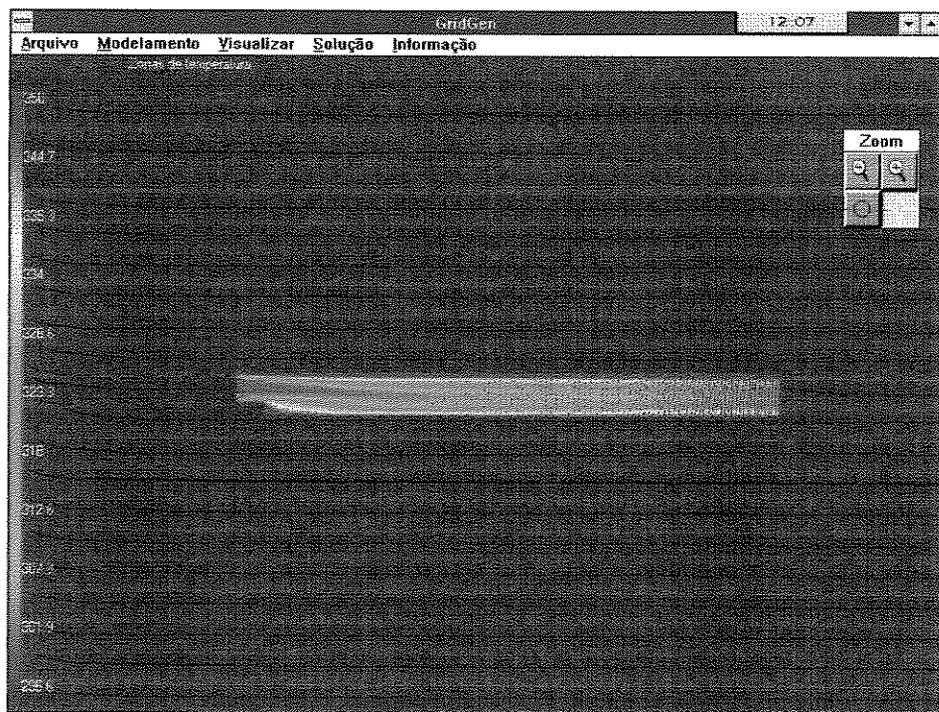


Figura 6.45: zonas de temperatura constante - expansão

De modo semelhante ao cilindro, o parâmetro aqui comparado também foi o comprimento da zona de recirculação, mas como esta zona é relativamente longa, não houve problemas com a distribuição de nós, para todas as malhas foram conseguidos nós dentro desta zona. Os resultados das malhas mais refinadas, 02 e 04, foram bem próximos para a zona de recirculação, distribuição de pressão e perfis de velocidade, mesmo nos gráficos onde há comparação somente entre as malhas usadas e não comparação com os autores citados.

Em todas as malhas foram apresentadas zonas de maior pressão logo após a zona de recirculação.

Nos gráficos apresentados para $Re=150$, não há resultados para malhas 01 e 03, porque não houve convergência para o campo de escoamento.

Capítulo 7

Conclusões

Baseado nos vários resultados apresentados ao longo do capítulo 6, conclui-se que com os programas *GridGen*, *Gridg*, *Fluid* e *Therm* em mãos, é possível se fazer um estudo introdutório com bons resultados para vários casos.

Mas os programas apresentados, além de facilidades, apresentam algumas limitações e pontos que podem e devem ser melhorados.

O programa *GridGen* apresenta um ponto desfavorável em relação à informação das condições de contorno para as equações de Navier-Stokes e da Energia. As condições de contorno são impostas ponto a ponto, ou seja, deve ser acessado cada ponto individualmente se desejar impor uma condição a ele, e isto pode se tornar um pouco cansativo a medida que estes pontos aumentam em quantidade, este é outro aspecto que deve ser melhorado também.

O programa *Gridg*, gera malhas estruturadas com elementos de oito e quatro nós, o que restringe o programa a poder ser usado também como fonte para gerar arquivos a casos que trabalham somente com estes tipos de elementos. Isto deve ser tratado posteriormente com o intuito de dar condições ao programa para gerar malhas com elementos triangulares e também malhas não estruturadas em geral.

Uma ferramenta que teve bastante importância no desenvolvimento do programa GridGen, foi a visualização gráfica, ainda que primitiva, mas foi algo que tornou mais interativo o uso do programa, deu mais controle e conhecimento do que está acontecendo quando se usa o GridGen, tanto para geração geométrica do problema quanto para visualização dos resultados obtidos.

Em suma, o conjunto dos programas apresentados forma um sistema de fácil acesso e controle, e que apresenta resultados satisfatórios, para casos com dimensões próximas às usadas aqui.

Referências bibliográficas

Amsden A. A., Hirt C. W. , A Simple Scheme for Generating General Curvilinear Grids, *Journal of Computational Physics*, 11, 348-359, 1973.

Atkins, D. J., Maskall, S. J. and Patrick, M. A., Numerical Prediction of Separated Flows, *International Journal for Numerical Methods in Engineering*, vol. 15, pp 129-144, 1980.

Burmeister, Louis C. , *Convective Heat Transfer*, John Wiley & Sons, 1983.

Chang, M. W., and B. A. Findlayson , Heat Transfer in Flow Past Cylinders at $Re < 150$: Part 1, Calculations for Constant Fluid Properties, *Numerical Heat Transfer*, vol. 12, pp. 179-195, 1987.

Denham, M. K. and Patrick M. A., Laminar Flow Over a Downstream-Facing Step in a Two-Dimensional Flow Channel, *Trans. Inst. Chem. Engrs*, 52(4), pp 361-367, 1974.

Dennis, S. R. and Chang, G. Z., Numerical Solutions for Steady Flow Past a Cylinder at Reynolds Numbers up to 100, *Journal of Fluid Mechanics*, 42, 1970.

Fox Robert W., Mc.Donald Alan T., *Introdução à Mecânica dos Fluidos*, Guanabara Koogan, 1988.

Hood P. , Frontal Solution Program for Unsymmetric Matrices, *International Journal for Numerical Methods in Engineering*, vol. 10, 379-399, 1976.

Kreyszig, Erwin , *Advanced Engineering Mathematics*, Seventh Edition, John Wiley & Sons, 1993.

Maliska C. R. , *Transferência de Calor e Mecânica dos Fluidos Computacional*, LTC Livros Técnicos e Científicos, 1995.

Reddy J. N., *An Introduction to the Finite Element Method*, Second Edition, Mc Graw Hill, 1993.

Smith G. D. , *Numerical Solutions of Partial Differential Equations: Finite Difference Methods*, third Edition, Oxford Press, 1985.

Taylor C., Hughes T. G. , *Finite Element Programing of the Navier-Stokes Equations*, Pineridge Press Swansea, 1981.

Thames Frank C. , Thompson Joe F. , Mastin C. Wayne , and Walker Ray L. , Numerical Solutions for Viscous and Potential Flow about Arbitrary Two-Dimensional Bodies-Using Body-Fitted Coordinate System, *Journal of Computational Physics*, 24, 245-273, 1977.

Thompson Joe F. , Thames Frank C. , and Mastin C. Wayne , TOMCAT-A code for Numerical Generation of Boundary-Fitted Curvilinear Coordinate Systems on Fields Containing Any Number of Arbitrary Two-Dimensional Bodies, *Journal of Computational Physics*, 24, 274-302, 1977.

Thompson Joe F. , Mastin C. Wayne , Automatic Numerical Generation of Body-Fitted Curvilinear Coordinates for a Field Containing any Number of Arbitrary Two-Dimensional Bodies, *Journal of Computational Physics*, 15, 299-319, 1974.

Thompson Joe F. , Warsi Z. U. A. , Mastin C. Wayne , *Numerical Grid Generation: Foundations and Applications*, Elsevier Science Publishing Co., 1985.

Truong Dinh Tien, J. M. , An Structured Mesh Generation Method for 2d Domains, *Proceedings of the XII Brazilian Congress of Mechanical Engineering*, Brasilia, Brasil, 145-148, December, 1993.

White Frank M. , *Viscous Fluid Flow*, Second Edition, Mc Graw Hill, 1991.

Anexo A

Procedimento para geração de malhas

O programa apresentado, *GridGen*, possibilita a construção de malhas estruturadas simples ou multi conectadas. Mas para a construção das malhas, são necessários alguns procedimentos específicos pelos quais o programa está baseado.

De início vale ressaltar o fato de que todas as primitivas gráficas tratadas no programa, linhas, arcos, círculos e elipses, são representadas por um conjunto de pontos discretos no plano que juntos formam uma aproximação para a primitiva.

A seguir serão apresentados os passos para geração de malhas simples e multi conectadas, e com o desejo de tornar mais clara a apresentação dos passos, será acompanhado todo o procedimento para geração das malhas, por meio de exemplos.

Malhas simples conectadas:

Como exemplo para malhas simples conectadas, será considerada a geometria apresentada na figura a1, que representa uma malha em torno de meio cilindro, com as dimensões especificadas no desenho.

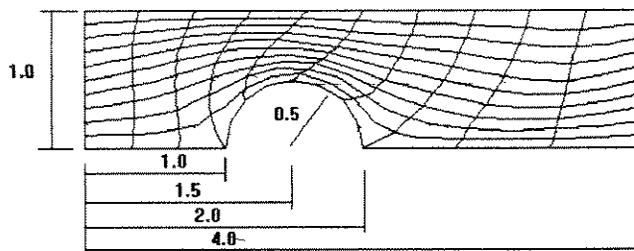


Figura a1: exemplo de malha simples conectada

Os passos necessários para a geração da malha são:

1- Ter o desenho da geometria do problema, sendo especificadas as dimensões. As dimensões são importantes para o uso das ferramentas de desenho disponíveis.

2- Ter pré-definido de quanto refinada será a malha.

3- Definir quantas divisões em ξ e em η terá cada segmento da malha transformada.

4- Ainda no desenho, identificar os pontos que serão tratados como vértices da malha transformada. Neste passo há várias considerações importantes a serem destacadas. Como já foi visto nos capítulos anteriores, a malha física é conformada em uma malha transformada, com elementos quadrangulares regulares. Então na malha física são identificados pontos chaves que servirão de vértices para a malha transformada. Considerando a figura a2, podemos identificar os pontos chaves da geometria.

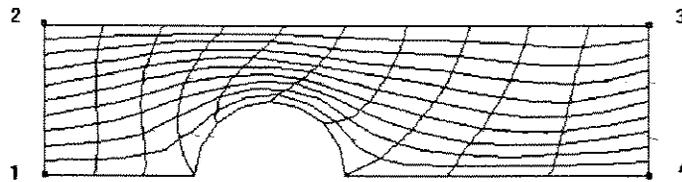


Figura a2: vértices da malha física - simples conectada

É importante ressaltar o fato de que o primeiro ponto gerado graficamente deverá ser também o primeiro vértice identificado para malha transformada. A malha transformada será formada com base na localização dos vértices, já que as linhas ξ e η constantes partem de uma região da malha para outra região, portanto a identificação dos vértices altera também a forma da malha calculada. No exemplo os vértices são os pontos 1,2,3,4; as linhas que partem do segmento 1-2 chegam ao segmento 3-4, e as linhas que partem do segmento 2-3 chegam ao segmento 4-1. Não importado a posição física dos pontos as linhas tomam este caminho.

A numeração utilizada na geração gráfica da geometria é simples. O primeiro ponto gerado é o ponto 1, o segundo 2, o terceiro 3, e assim por diante, portanto na identificação dos vértices devem ser usados os valores que identificam os pontos, figura a3. Os vértices para este exemplo são os pontos 1, 21, 41, 61. O número de pontos que compõem uma primitiva pode ser facilmente calculado da forma: $n = 2 * div + 1$, n é o número de pontos, div é em quantas divisões são formadas as primitivas, o segmento 1-21 é formado por 10 divisões em η o que dá um total de 21 pontos, e o segmento 21-41 é formado por 10 divisões em ξ o que dá um total de 21 pontos também, e o arco é formado por 4 divisões que dá um total de 9 pontos.

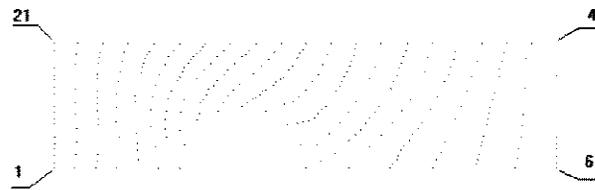


Figura a3: numeração na geração da geometria - simples conectada

Todos os passos anteriores podem ser feitos antes da utilização do programa. Agora através do programa os passos necessários para a geração gráfica serão demonstrados.

5- Identificar qual o tipo de malha que será criada através do comando no programa, figura a4.



Figura a4: comando para tipo de malha

6- Para malhas simples conectadas a geometria desenhada será identificada como contorno 1, que deve ser identificado através do comando no programa, figura a5.

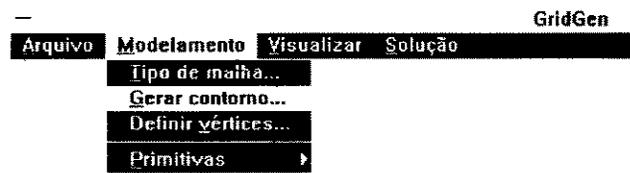


Figura a5: comando para identificar o contorno gerado

7- Através do comando, identificar os vértices da malha, figura a6.



Figura a6: comando para identificar os vértices

8- Com o número de divisões definidos, desenhar a geometria com as ferramentas disponíveis, figura a7.

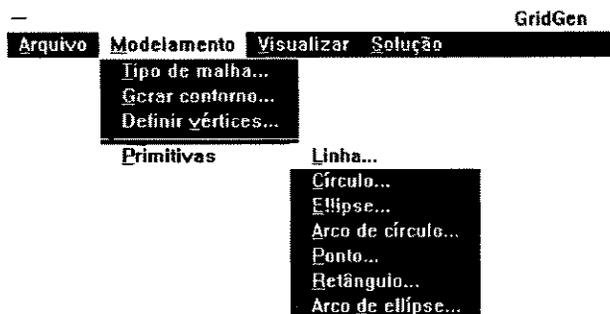


Figura a7: ferramentas de desenho

O exemplo citado foi construído da seguinte forma, seguindo a figura a8:

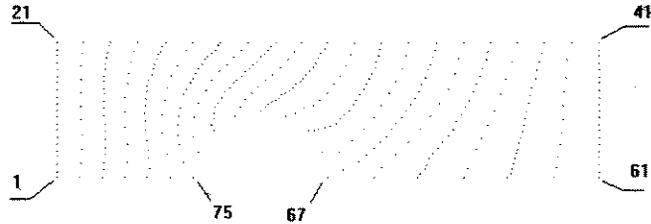


Figura a8: sequência de construção de geometria - simples conectada

- segmento de linha do ponto 1 ao 21;
- segmento de linha do ponto 21 ao 41;
- segmento de linha do ponto 41 ao 61;
- segmento de linha do ponto 61 ao 67;
- arco de círculo do ponto 67 ao 75;
- segmento de linha do ponto 75 ao 1;

9- Executar o comando de geração de malha transformada.

10- Executar o programa *Gridg*.

11- Executar o comando de atualização de malha, para visualizar a malha calculada.

Malhas multi conectadas:

Para malhas multi conectadas, será considerada a geometria descrita na figura a9, que é uma malha formada ao redor de um cilindro.

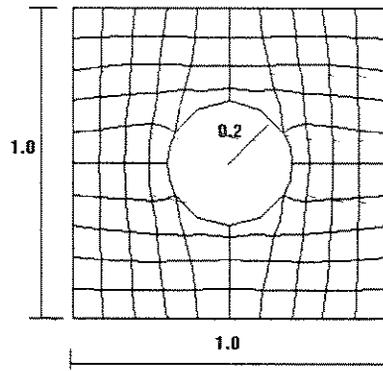


Figura a9: exemplo de malha multi conectada

Os passos para geração de malhas multi conectadas são semelhantes aos da malha simples conectada, descritos a seguir:

1- Ter o desenho da geometria do problema, sendo especificadas as dimensões. As dimensões são importantes para o uso das ferramentas de desenho disponíveis.

2- Ter idéia de quanto refinada será a malha.

3- Definir quantas divisões em ξ e em η terá cada segmento da malha transformada e do corpo.

4- Ainda no desenho, identificar os pontos que serão tratados como vértices da malha transformada e os vértices do corpo. Como para malha simples conectada, há várias considerações importantes a serem destacadas. Como já foi visto nos capítulos anteriores, a malha física é conformada em uma malha transformada, com elementos quadrangulares regulares. Então na malha física são identificados pontos chaves que servirão de vértices para a malha transformada e para o corpo. Considerando a figura a10, podemos identificar os pontos chaves da geometria.

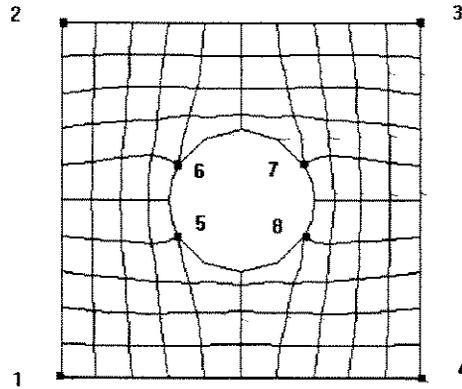


Figura a10: vértices da malha física - multi conectada

Como na malha simples conectada, aqui também o primeiro ponto gerado no plano físico será considerado da mesma forma, que deverá ser também o primeiro ponto da malha transformada. De modo semelhante ao exemplo anterior, também as linhas ξ e η têm o mesmo caminho, as que partem do segmento 1-2 chegam ao 3-4, e as que partem do segmento 2-3 chegam ao 4-1, mas há algumas linhas que partem do contorno externo e chegam ao corpo, segmentos 5-6, 6-7, 7-8, 8-5. Os pontos 5, 6, 7 e 8 são chamados de pontos de contato do corpo com o contorno externo. Eles determinam quantas divisões há no corpo, quantas divisões há abaixo, acima, à esquerda e à direita do corpo.

A numeração dos pontos também obedece à mesma ordem apresentada no caso anterior. Então assim sendo, os pontos que formam a geometria têm a seguinte numeração, figura a11, que deve ser usada para identificação dos vértices do contorno externo e do corpo. O número de pontos pode ser calculado da mesma forma já apresentada.

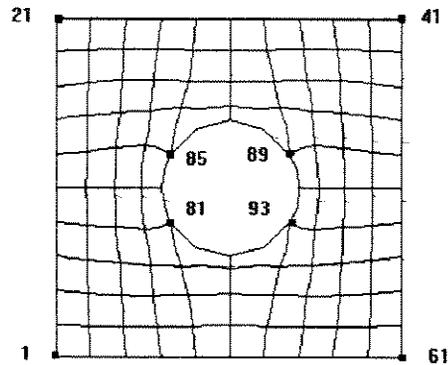


Figura a11: numeração na geração da geometria - multi conectada

Para o caso de malhas multi conectadas, o contorno externo e o corpo são identificados e gerados separadamente.

5- Agora no programa, deve-se identificar qual o tipo de malha que será construída, idem ao passo 5 para malha simples conectada.

6- Os passos 6, 7, 8 devem ser seguidos para cada contorno que for gerado. No comando de identificação do tipo de malha devem ser definidas quantas divisões na malha devem haver abaixo e à esquerda do corpo. Neste exemplo foram definidas 4 divisões abaixo e 4 à esquerda.

Para o exemplo, a sequência dos passos para geração gráfica da geometria pode ser seguida pela figura a12.

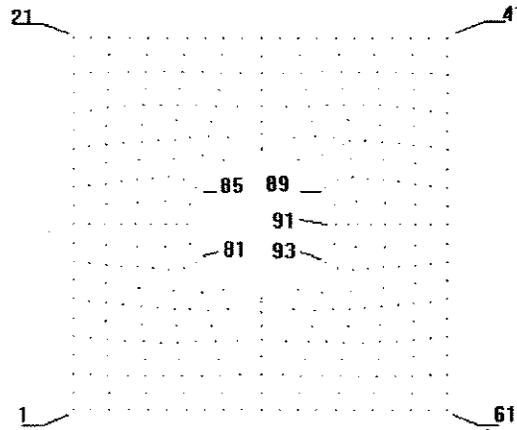


Figura a12: sequência de construção de geometria - multi conectada

- segmento de linha do ponto 1 ao 21;
- segmento de linha do ponto 21 ao 41;
- segmento de linha do ponto 41 ao 61;
- segmento de linha do ponto 61 ao 1;
- voltar ao passo 6;
- arco de círculo do ponto 81 ao 85;
- arco de círculo do ponto 85 ao 89;
- arco de círculo do ponto 89 ao 91;
- arco de círculo do ponto 91 ao 93;
- arco de círculo do ponto 93 ao 81;

7- Executar o comando de geração de malha transformada.

8- Executar o programa *Gridg*.

9- Executar o comando de atualização de malha, para visualizar a malha calculada.

As duas malhas apresentadas aqui, foram geradas sem o uso das funções de controle de atração P e Q , da equação 3.1, ou seja foram consideradas zero; $P=0$, $Q=0$. Para exemplificar o uso destas funções, considerando uma malha gerada para o caso da cavidade, figura a.13.

malha original (sem efeitos das funções de controle)

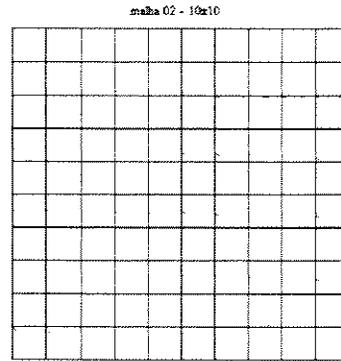
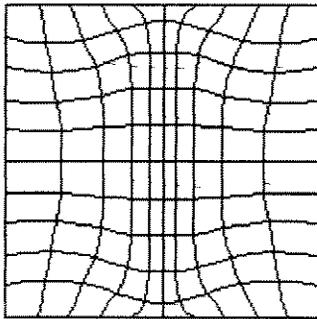


Figura a.13: malha original para cavidade

Considerando agora diferentes formas de atração,

Linhas ξ para linha $\xi=11$



Linhas η para linha $\eta=11$

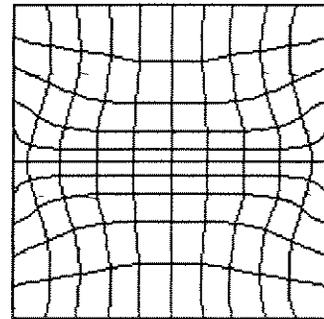


Figura a.14: atração para linha $\xi=11$

fatores de decaimento:

função	P	Q
$a=$	1000	0
$b=$	0	0
$c=$	0.5	0
$d=$	0	0

Figura a.15: atração para linha $\eta=11$

fatores de decaimento:

função	P	Q
$a=$	0	1000
$b=$	0	0
$c=$	0	0.5
$d=$	0	0

Linhas ξ para ponto $\xi=11, \eta=11$ Linhas η para ponto $\xi=11, \eta=11$

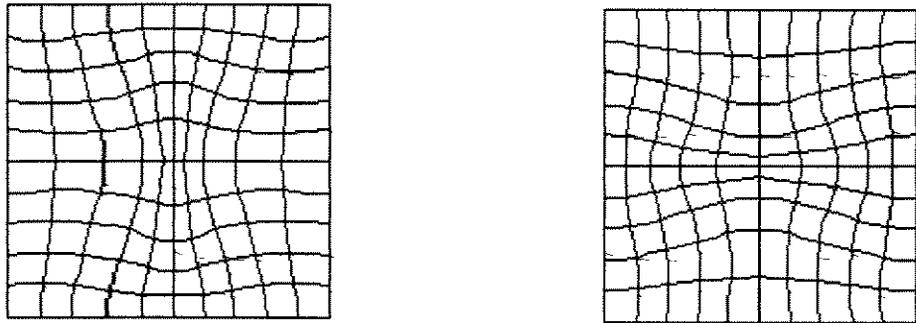


Figura a.16: atração para ponto $\xi=11, \eta=11$ Figura a.17: atração para ponto $\xi=11, \eta=11$

fatores de decaimento:

função	P	Q
$a=$	0	0
$b=$	1000	0
$c=$	0	0
$d=$	0.5	0

fatores de decaimento:

função	P	Q
$a=$	0	0
$b=$	0	1000
$c=$	0	0
$d=$	0	0.5

Linhas ξ e linhas η para ponto $\xi=11, \eta=11$

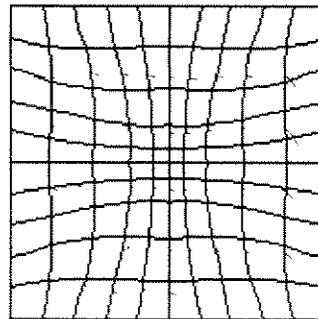


Figura a.18: atração para ponto $\xi=11, \eta=11$

fatores de decaimento:

função	P	Q
$a=$	0	0
$b=$	1000	1000
$c=$	0	0
$d=$	0.5	0.5

