

**UNIVERSIDADE ESTADUAL DE CAMPINAS**  
**Instituto de Matemática, Estatística e Computação Científica**  
**Departamento de Matemática Aplicada**

Dispersão Estacionária de Populações: Modelagem e  
Instrumentação Numérica.

Dissertação de Mestrado  
por  
**Roberto Henrique Heiderich**

**Orientador: Prof. Dr. Sílvio de Alencastro Pregolato**

Este trabalho contou com apoio financeiro da CAPES

# Dispersão Estacionária de Populações: Modelagem e Instrumentação Numérica.

Este exemplar corresponde à redação final da Dissertação devidamente corrigida, defendida por **Roberto Henrique Heiderich** e aprovada pela comissão julgadora.

Campinas, 17 de Dezembro de 2004

.....  
Prof. Dr. Sílvio de Alencastro Pregnotatto

Orientador

## **Banca Examinadora:**

1. Prof. Dr. Sílvio de Alencastro Pregnotatto
2. Prof. Dr. João Frederico da Costa Azevedo Meyer
3. Profa. Dra. Tânia Maria Vilela Salgado Lacaz

Dissertação apresentada ao Instituto de Matemática, Estatística e Computação Científica, UNICAMP, como requisito parcial para obtenção do Título de Mestre em Matemática Aplicada

**FICHA CATALOGRÁFICA ELABORADA PELA  
BIBLIOTECA DO IMECC DA UNICAMP**

Heiderich, Roberto Henrique

H362d      Dispersão estacionária de populações: modelagem e  
instrumentação numérica / Roberto Henrique Heiderich -- Campinas,  
[S.P. :s.n.], 2004.

Orientador : Silvio de Alencastro Pregolato

Dissertação (mestrado) - Universidade Estadual de Campinas,  
Instituto de Matemática, Estatística e Computação Científica.

1. Dinâmica (Modelagem). 2. Equações diferenciais parciais  
elípticas. 3. Métodos dos elementos finitos I. Pregolato, Silvio de  
Alencastro. II. Universidade Estadual de Campinas. Instituto de  
Matemática, Estatística e Computação Científica. III. Título.

**Dissertação de Mestrado defendida em 17 de dezembro de 2004**  
**e aprovada pela Banca Examinadora composta pelos Profs. Drs.**



---

**Prof (a). Dr (a). SILVIO DE ALENCASTRO PREGNOLATTO**



---

**Prof (a). Dr (a). TÂNIA MARIA VILELA SALGADO LACAZ**



---

**Prof (a). Dr (a). JOÃO FREDERICO DA COSTA AZEVEDO MEYER**

## Resumo

Neste trabalho propõe-se um método para o ajuste do coeficiente de dispersão em fenômenos de Dinâmica Populacional que se apresentem modelados por uma EDP elíptica não-linear aplicada a domínios com contornos suaves. Para um dos problemas de valor de contorno originados, obtém-se um resultado de existência de solução combinando o Método de Galerkin e o Teorema de ponto-fixo de Brower. As soluções aproximadas foram obtidas com o uso do Método dos Elementos Finitos, com as discretizações dos domínios feitas com auxílio de um gerador de malhas (no presente trabalho usa-se o Gmsh: aplicativo disponível em rede). Este procedimento foi transformado em algoritmo, programado no ambiente Matlab, com os resultados numéricos apresentados utilizando o ambiente gráfico do próprio Matlab.

## Abstract

The main purpose of this work is to present a method for adjusting dispersal coefficients in population phenomena that include the population dynamics of a studied species. The adopted model is that of a elliptic PDE with non-linear terms for the mentioned dynamics, on a domain with adequate smoothness conditions on the boundary. For one of the analyzed boundary-value problems, an existence result is obtained for the solution combining Galerkin's Method with Brouwer's Fixed Point theorem. Approximated solutions were obtained with the use of the Finite Element Method upon domains which were discretized with the use of a mesh-generator (in this work, the choice was for Gmesh, available on the internet). This described procedure was translated in terms of an algorithm, programmed in MATLAB, and the numerical results were also presented using MATLAB visualization tools.

*Este trabalho, dedico-o à minha mãe,  
meus irmãos, e à memória de meu pai.*

# Agradecimentos

A opção por escrever este texto na terceira pessoa não se refere apenas a uma escolha de forma literária, mas principalmente ao reconhecimento pela inestimável ajuda de tantos colegas e professores que me apontaram caminhos, ou por hora me incentivaram, na obtenção dos resultados que aqui figuram. Agradecê-los todos, nominalmente, seria por si só trabalho árduo, creio eu. Mas não me eximo da obrigação de agradecimentos pessoais aos que, juntamente comigo, sofreram ou saborearam, mesmo que brevemente, as etapas de evolução deste trabalho.

Ao amigo e orientador Sílvio, sem quem este trabalho não se completaria com certeza, pelo seu apoio transcendendo a ajuda técnica apenas.

Ao professor e amigo João Frederico (Joni), por me receber de braços abertos no grupo de pesquisa o qual lidera, local onde encontrei uma de minhas vocações, a de Biomatemático.

Às professoras Tânia Lacaz e Renata Sossae e ao professor Rodney, por suas presenças e sugestões para o aprimoramento deste trabalho.

Aos professores Yuri e Boldrini, por compartilharem seus inestimáveis conhecimentos matemáticos.

Aos colegas Geraldo Diniz e Paulo Lopes, pelas críticas e sugestões à este trabalho.

Às funcionárias Tânia, Cidinha e Fátima por terem assumidos os encargos burocráticos que viabilizaram minha vida acadêmica.

Agradecimentos também à CAPES, financiadora do trabalho.

Campinas, SP, Dezembro, 2004.

Roberto Henrique Heiderich

# Sumário

<b>Agradecimentos</b>	<b>v</b>
<b>Lista de Figuras</b>	<b>ix</b>
<b>Lista de Tabelas</b>	<b>x</b>
<b>Introdução</b>	<b>1</b>
<b>1 Uma Descrição Macroscópica dos Modelos de Dinâmica Populacional</b>	<b>5</b>
1 Euler e a Navalha de Ockham . . . . .	5
2 Conceitos de Espaço de Aspecto e Densidade de Distribuição . . . . .	7
3 Uma descrição fenomenológica do Princípio de Conservação . . . . .	9
3.1 O conceito de Fluxo e Densidade de Fluxo: Relações Constitutivas . . .	10
3.2 Fluxo por Transporte . . . . .	12
3.3 Fluxo Difusivo . . . . .	13
3.4 Fluxo por Taxia . . . . .	15
3.5 Densidade Superficial de Fonte . . . . .	18
3.6 Descrição de um Cenário . . . . .	19
4 O Problema e sua Modelagem . . . . .	22
<b>2 Análise e Discretização da Equação Elíptica</b>	<b>25</b>
1 Formulação variacional . . . . .	25
2 Adimensionalização da equação . . . . .	28
3 Existência de solução . . . . .	29
4 Discretização do problema . . . . .	39
5 Métodos Iterativos de Resolução: estudo da convergência . . . . .	44



<b>3</b>	<b>Resultados, Análise e Discussão</b>	<b>48</b>
1	Testes de validação do código . . . . .	48
2	Aplicações . . . . .	54
2.1	Simulação em dois ambientes adjacentes . . . . .	55
2.2	Ajustando o coeficiente de dispersão . . . . .	59
3	Análise dos resultados . . . . .	65
4	Conclusões . . . . .	66
	<b>Considerações finais</b>	<b>67</b>
	<b>Bibliografia</b>	<b>69</b>
<b>A</b>	<b>Conceitos, definições e teoremas</b>	<b>73</b>
1	Conceitos em análise funcional . . . . .	73
1.1	Definições . . . . .	73
1.2	Teoremas . . . . .	75
1.3	Lemas . . . . .	77
1.4	Desigualdades . . . . .	78
<b>B</b>	<b>Formato de arquivos para o programa</b>	<b>79</b>
1	Arquivo de dados de entrada . . . . .	79
<b>C</b>	<b>Códigos fontes</b>	<b>84</b>
1	Códigos para o MATLAB® . . . . .	84
1.1	Código principal . . . . .	84
1.2	Montagem das matrizes de contribuição local $m^e$ e $\sigma^e$ . . . . .	90
1.3	“Assembly” das matrizes $\mathbf{M}$ e $\mathbf{\Sigma}$ . . . . .	91
1.4	Acrescentando fontes pontuais (“populações nodais”) . . . . .	91
1.5	Acrescentando condições de contorno pontuais . . . . .	92
1.6	Rotinas de construção da matriz $\mathbf{N}$ . . . . .	92
1.7	Adaptando a matriz $\mathbf{N}$ às condições de contorno pontuais . . . . .	93
1.8	Rotinas para Método de Newton . . . . .	93
1.9	Dados de amostragem . . . . .	94

1.10	Montagem das matrizes de contribuição local $n^e$ . . . . .	94
1.11	“Assembly” da matriz $\mathbf{N}$ . . . . .	95
1.12	Rotinas para integração sobre fronteiras – montagem de $\mathbf{M}$ . . . . .	95
1.13	Rotinas para integração sobre elementos quadrados – montagem de $\mathbf{M}$	96
1.14	Rotinas para integração sobre elementos triangulares – montagem de $\mathbf{M}$	97
1.15	Rotinas para integração sobre fronteiras – montagem de $\mathbf{N}$ . . . . .	98
1.16	Rotinas para integração sobre elementos quadrados – montagem de $\mathbf{N}$ .	99
1.17	Rotinas para integração sobre elementos triangulares – montagem de $\mathbf{N}$	100
1.18	Funções base unidimensionais . . . . .	101
1.19	Funções base bidimensionais sobre elementos quadrados . . . . .	102
1.20	Funções base bidimensionais sobre elementos triangulares . . . . .	103
1.21	Construção da Matriz Jacobiana . . . . .	105

# Lista de Figuras

1.1	características da área geográfica . . . . .	20
3.1	solução analítica (malha com 24 triângulos) . . . . .	52
3.2	solução aproximada (malha com 24 triângulos) . . . . .	52
3.3	solução analítica (malha com 228 triângulos) . . . . .	52
3.4	solução aproximada (malha com 228 triângulos) . . . . .	52
3.5	solução analítica ( $\mathbf{u}_{analit}$ ) vs aproximada ( $\mathbf{u}_{approx}$ ), com elementos de 1 <sup>a</sup> ordem .	53
3.6	solução analítica ( $\mathbf{u}_{analit}$ ) vs aproximada ( $\mathbf{u}_{approx}$ ), com elementos de 2 <sup>a</sup> ordem .	53
3.7	solução aproximada com simetria radial . . . . .	54
3.8	simulação de uma invasão em dois ambientes (meios) adjacentes . . . . .	55
3.9	meios adjacentes: cenário 1 . . . . .	56
3.10	meios adjacentes: cenário 2 . . . . .	57
3.11	meios adjacentes: cenário 3 . . . . .	58
3.12	cenário para simulações . . . . .	61
3.13	modelo de malha discretizando o domínio . . . . .	62
3.14	colonização no estado estacionário . . . . .	64

# Lista de Tabelas

3.1	desvios relativos cometidos nas aproximações de uma solução analítica . . . . .	51
3.2	meios adjacentes: parâmetros para cenário 1 . . . . .	56
3.3	meios adjacentes: parâmetros para cenário 2 . . . . .	57
3.4	meios adjacentes: parâmetros para cenário 3 . . . . .	58
3.5	ajuste de coeficiente de dispersão em cenário de invasão . . . . .	63

# Introdução

Somente a partir de meados do século XX, na medida em que começa a mostrar coerência nos princípios e técnicas matemáticas utilizadas, passando a se definir como área específica do conhecimento, é que o estudo da Dinâmica de Populações tem evolução significativa. Até então, desenvolveu-se de forma esparsa e em decorrência de trabalhos publicados episodicamente ao longo dos últimos dois séculos, desde o seminal artigo de Leonhard Euler (Euler, 1970) em 1760.

Apresenta-se, portanto, como área científica bastante recente e que se supõe próspera em desenvolvimento, dada a crescente evolução na cooperação entre a Matemática e áreas correlatas a esses processos, como a Biologia, tendo como exemplo paradigmático o ocorrido entre a Matemática e a Física.

Hoje, a Dinâmica de Populações é largamente empregada na Biomatemática, representando uma formalização matemática, e por conseqüência uma síntese, de diversas teorias biológicas desde fenômenos de ocorrências à nível molecular, observáveis em Processos Físico-Químicos, passando pelo nível celular em Fisiologia, e chegando a processos epidemiológicos e sociobiológicos em organismos superiores, aqui incluindo-se evidentemente as sociedades humanas. Alinhava, do ponto-de-vista matemático, áreas de conhecimento que na sua origem biológica se mostravam por demais multifacetadas.

Neste amplo universo de que trata a dinâmica de populações destacamos a *Etologia*, ou ciência do comportamento, que tem no movimento espacial uma de suas vertentes de estudo. Os conceitos desenvolvidos para este tipo de movimento servem de protótipos ao estudo de deslocamentos virtuais em espaços de aspecto (do indivíduo) mais abstratos. Por exemplo, um movimento no espaço etário é análogo a um movimento de transporte causado por uma corrente de fundo com velocidade unitária. Representa, então, poderosa ferramenta na análise e compreensão de diversos fenômenos biológicos, argumentos que por si só justificam

seu estudo.

O desenvolvimento desta ciência também se deu apenas nas últimas décadas e graças a trabalhos desenvolvidos por Lorenz (1974), Tinbergen (1974), e Dukas (1998) para animais superiores, entre outros.

Em especial, na análise desses processos de deslocamento espacial no âmbito da Dinâmica Populacional, faz-se uso, em muitas situações, de equações ou de sistemas de equações de evolução a derivadas parciais. Este procedimento normalmente representa um ganho significativo na modelagem de tais fenômenos. São muitos os trabalhos produzidos abordando situações evolutivas em fenômenos com variações espaciais a partir de equações diferenciais. Todos seguem uma linha historicamente iniciada por Skellam (1951), Inglaterra, que justificou o uso de equações diferenciais parabólicas na descrição de fenômenos de dispersão populacional ao estudar a dispersão do carvalho inglês após a era glacial. Kareiva, Okubo, além dos brasileiros: Diniz, Lacaz, Pregnoatto, Sossae, Cantão, são pesquisadores e autores de trabalhos nessa área, dentre outros.

Uma outra abordagem bastante útil, na análise de fenômenos que envolvem aspectos dispersivos e migratórios de populações, é a de se usar a EDP elíptica (a equação estacionária de dispersão e de migração), embora introduzindo dificuldades analíticas significativas, especialmente quando se considera dentre os fenômenos modelados aqueles ligados à migração e aos movimentos, e principalmente numéricas quando da introdução de dinâmicas vitais populacionais não-lineares.

Em 1983, Kareiva (1983) fez uso original de uma equação estacionária aplicando-a para avaliar os parâmetros de uma dispersão populacional efetiva. Em outras palavras, propôs uma técnica, a de experimentos de campo de marca-recaptura, para obter o coeficiente  $D$  de dispersão<sup>1</sup> para dada espécie, ajustando os dados desta dispersão à referida equação.

Este tipo de estudo pode ter grande significância, por exemplo, em infestações danosas a plantações por insetos herbívoros. Nestes tipos de infestação, um dos fatores que determinam a colonização da plantação por estes insetos é a distribuição espacial da vegetação, a qual, dentro de certos limites, pode ser modificada, resultando numa alteração no comportamento de procura de alimento pelo inseto, pela alteração de seus movimentos, denominados **movimentos triviais**. Desta forma, mudanças no movimento trivial desses insetos, dificultando sua fixação, poderia resultar num menor grau de invasão, como o que se observa

---

<sup>1</sup>Medida tanto da transitabilidade do meio como da capacidade motil dos indivíduos.

em policulturas. Características como alternância entre áreas cultivadas e não cultivadas, ou alternâncias no cultivo de espécies diferentes, são interferências possíveis e que provocam alterações no movimento trivial, reduzindo os efeitos da invasão. Tais características podem ser estabelecidas a priori, e dentro dos limites aceitáveis, tendo como informação unicamente o valor do coeficiente de dispersão do inseto invasor associado ao tipo de plantação invadida.

Sendo assim, e seguindo as idéias de Kareiva, sugerimos uma forma de se efetuar o ajuste do coeficiente de dispersão em uma equação elíptica, que necessita, no entanto, de um bom “resolvedor” dessa equação aplicada a domínios com certa irregularidade nas formas e complexidade nas condições definidas sobre suas fronteiras.

Visando a busca pelo “resolvedor”, este trabalho apresenta uma análise e aproximação de fenômenos modelados por este último tipo de equação diferencial parcial, fenômenos que, como já citado, possuem suas características de dispersão-migração variando apenas espacialmente.

A irregularidade natural dos domínios em que tais situações se dão, bem como as condições de contorno, levam de modo bastante natural à procura de aproximações numéricas de precisão minimamente razoáveis e, atualmente, a escolha natural é a de se trabalhar com o Método dos Elementos Finitos.

Estabelecido o modelo em sua formulação clássica, no qual constarão um termo de dispersão ou difusão, o termo da migração, um termo para indicar uma possível hostilidade do meio, além da dinâmica vital densidade dependente, passamos à sua formulação fraca, ou variacional. Aqui, para efeito de estabelecermos esta nova formulação do problema dentro do nível de generalizações proposto por este trabalho, consideramos nosso domínio subdividido em subdomínios (dois ou mais) delimitados por fronteiras suaves e de medida nula, sobre as quais, separadamente, aplicamos diferentes condições de contorno (Dirichlet, Neumann ou mista) através de funções também suaves. Cada subdomínio irá descrever um meio com características próprias de dispersão, migração, capacidade de suporte etc, representados por parâmetros constantes na equação que aproxima o fenômeno em estudo. Saltos nas citadas características ocorrerão nas fronteiras interdomínios.

Um resultado de existência da solução fraca foi obtido a partir do *Método de Galerkin* juntamente com um teorema de ponto fixo. Já a existência de solução positiva (como resultado apropriado ao estudo de dinâmicas populacionais) pode ser obtida pelo *Método de Pohozaev* para problemas com condições de fronteira de Dirichlet nulas. Tal método, além

da caracterização de soluções positivas, permite mostrar a multiplicidade das mesmas. É tema de estudo do grupo de Biomatemática da UNICAMP (de Santana, 2003), e vale citá-lo, uma vez que aponta analiticamente na direção de existência desses tipos de solução.

A discretização da formulação variacional, via *Método de Galerkin*, com as funções testes do *Método dos Elementos Finitos*, leva a um sistema não-linear de equações cuja solução pelo *Método de Newton*, a partir de uma boa aproximação inicial (exigência para garantia de convergência deste método) obtida usando-se uma técnica de máxima descida, descreve a solução a se aproximar.

A discretização do domínio (região geográfica) é feita com auxílio de um gerador de malhas (no presente trabalho usa-se o **Gmsh**: aplicativo disponível em rede)

Este procedimento foi transformado em algoritmo, programado no ambiente **Matlab** disponível no laboratório do IMECC da UNICAMP, com os resultados numéricos apresentados de modo visualmente adequado à compreensão das simulações realizadas, utilizando o ambiente gráfico do próprio **Matlab**.

São apresentados dois testes utilizados na validação do código, com o objetivo de apontar a direção da correta construção dos algoritmos, e uma simulação como resultado final de um processo de ajuste do coeficiente de dispersão.

Destaca-se assim, o desenvolvimento de um código de programação, que permite aproximações numéricas para vários cenários descritivos de movimentos espaciais provocados por comportamentos sociobiológicos, e a partir da discretização de domínios (regiões sobre as quais se dão tais movimentos) com configurações bastante gerais, retratando, com boa aproximação, o real habitat da espécie em estudo. Neste código podemos obter os refinamentos dos resultados numéricos tanto atuando sobre as dimensões da malha, dito refinamento- $h$ , como sobre o grau dos polinômios usados na construção das funções testes presentes na formulação variacional, o dito refinamento- $p$ .



# Capítulo 1

## Uma Descrição Macroscópica dos Modelos de Dinâmica Populacional

Neste capítulo apresentaremos os ingredientes necessários à construção de um modelo dinâmico, visando uma descrição passo a passo dos conceitos associados a cada um dos termos da equação elíptica objeto das análises e aproximações apresentadas neste texto. Convém lembrar que a equação elíptica, escopo deste trabalho, obtém-se de uma equação parabólica quando a derivada em relação ao tempo se anula, e a solução do problema se torna função exclusiva das variáveis espaciais (situação denominada de estado permanente). O termo macroscópico, no título deste capítulo, indica que se faz uma descrição coletiva do fenômeno, obtida a partir de observações de comportamentos individuais de cada um dos elementos constituintes da população. O exemplo-problema de Euler (*modelo populacional com distribuição por idade*) será utilizado, concomitantemente com seu modelo para dinâmica de fluidos, como veículo na apresentação de tais conceitos. Todo este capítulo, nas descrições que seguem, e, principalmente, no ordenamento lógico, teve influência de leitura de texto a partir de conferência proferida no Congresso Latino Americano de Biomatemática de 2001, pelo Prof. Wilson Castro Ferreira Jr.

### 1 Euler e a Navalha de Ockham

A Dinâmica populacional tem por objetivo a descrição do número de indivíduos de uma população. Esta variável, por descrever o estado (condição) da população a cada instante de tempo, é dita *estado do sistema* ou, mais comumente, *variável de estado*.

Iniciamos a construção da dinâmica considerando um modelo matemático contínuo para evolução temporal de uma dada população e uma função  $\rho : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ , onde  $\rho(t)$ , representando o número de indivíduos da população no instante  $t$ , permite descrever estados passados e prever estados futuros, uma vez conhecido um seu estado.

Na construção do modelo demográfico de Euler, consideremos inicialmente que o estado do sistema varie apenas por conta de nascimentos e mortes na população, sem suposição ainda de outros processos que o modifiquem, como os migratórios, por exemplo<sup>1</sup> (atingiremos a complexidade do nosso modelo, como um todo, paulatinamente ao longo desta exposição).

Malthus abordou este problema propondo uma dinâmica bastante simples (e igualmente robusta), baseando-se na hipótese biológica de que as taxas de nascimento e morte são proporcionais ao número de indivíduos no estado presente, ou seja:

$$\frac{\partial \rho}{\partial t} = \nu \rho$$

Tal modelo fornece uma excelente descrição (ou caricatura, como se refere Segel (1984a)) da evolução temporal da população. Mas, considerando que taxas de nascimentos e mortes são proporcionais ao número total de indivíduos, parte da hipótese biológica de que se mantêm homogêneos seus aspectos vitais característicos. Não se distingue ainda, por exemplo, a capacidade sexual reprodutiva ou pré-disposição à morte, nas diferentes faixas etárias.

Já Euler, no seu estudo sobre populações humanas, deparou-se com a necessidade de acrescentar maiores detalhes ao modelo matemático dinâmico, de forma a contemplar as variações de natalidade e mortalidade com a distribuição etária, aspectos heterogêneos neste tipo de população. Parafraseando Ferreira Jr (2001), o modelo de Malthus é evidentemente cego para estas e outras tantas características.

Surge então o conflito entre o que é desejável e o que é necessário acrescentar-se a um modelo matemático.

A inclusão de detalhes em excesso, próprio de um impulso reducionista, é indesejável tendo em vista a dimensão intratável que o modelo pode adquirir. Mas, em contrapartida, a falta de informações pode incapacitar o modelo quanto a descrição matemática do fenômeno.

---

<sup>1</sup>Observe que em processos Físicos-Químicos, os termos nascimentos e mortes têm conceitos generalizados, podendo ser interpretados como fenômenos reativos com, respectivamente, geração e consumo, ou acréscimo e retirada, de substâncias presentes ou necessárias ao processo.

Ao conflito imobilizador contrapõe-se o princípio filosófico-científico conhecido como Lei da Parcimônia ou Navalha de Ockham, devido a seu anunciador, William de Ockham, o qual afirma que “não se deve multiplicar entidades desnecessariamente”. Sua interpretação mais comum diz que frente a duas hipóteses igualmente satisfatórias, devemos escolher a que necessite de menos elementos. Deve-se, no entanto, atentar para o termo “satisfatória” que aponta não necessariamente para o argumento mais óbvio ou de fácil entendimento. Trata-se na verdade mais de um alerta que lei, ou princípio, que auxilia o matemático aplicado na escolha de hipóteses e instrumentação matemática as mais simples dentre as úteis e factíveis (o modelo deve ser construído a partir de considerações básicas obtidas de informações relevantes do fenômeno em estudo). Quando adequadamente aplicado, garante qualidade científica e eficiência ao modelo construído. Como nos lembra Ferreira Jr (2001): “a única descrição perfeita de um fenômeno natural é o próprio fenômeno”; ou enfatizando Lee Segel, o modelo matemático é a “caricaturização” do fenômeno, na medida que realça certos aspectos que se deseja observar em detrimento dos demais.

Assim, como exemplar aplicação de parcimônia, Euler incorpora a seu modelo apenas o que possui de informação biológica (taxas de natalidade e mortalidade), de forma a permitir que o estado do sistema passe a descrever a distribuição etária. Este modelo de Euler é formalmente idêntico ao de Malthus e escrito na forma

$$\frac{d\rho}{dt} = \Lambda\rho$$

com o operador  $\Lambda$  contendo todas as hipóteses biológicas necessárias disponíveis.

## 2 Conceitos de Espaço de Aspecto e Densidade de Distribuição

Euler foi quem primeiramente introduziu o conceito de espaço de aspecto. Representado no modelo populacional, com distribuição etária, pela reta real positiva ( $\mathbb{R}^+$ ), tem todos os indivíduos “virtualmente” localizados sobre este espaço segundo a medida do aspecto idade.

Com os indivíduos da população distribuídos no espaço de aspecto, Euler também introduz o conceito de *função densidade de distribuição* ( $\rho$ ) (ou *função de distribuição*, ou ainda *função densidade*, simplesmente), e já que não se desejava nem se necessitava de representações excessivamente detalhadas de informações sobre a população, definiu-a, como modelo matemático, por meio de integrais

$$\int_{\Omega} \rho(x) dx = \text{“número de indivíduos localizados na região } \Omega \text{ do espaço de aspecto”},$$

para a descrição do cenário obtido.

No modelo demográfico esta integral representa o número de indivíduos da população com idade em  $\Omega$ . Notemos como tal definição se adequa bem a este caso de distribuição por faixa etária, pois realmente não há relevância em localizar pontualmente os indivíduos sobre o espaço de aspecto idade. A definição pontual ou Dedekiniiana desta função tem apenas o propósito de permitir que as integrais possam ser calculadas.

Tem-se assim uma descrição dos estados do sistema por meio de funções apropriadamente integráveis (sem patologias) sobre espaços de aspectos regulares (também sem patologias).

Por sua vez  $\rho(x, t)$ ,  $t \geq 0$ , descreve a evolução temporal do sistema dinâmico, tal que,  $\rho_t(x)$  determina, a cada instante  $t$ , o estado do sistema.

Em analogia com o modelo malthusiano, representamos a dinâmica deste sistema como

$$\frac{\partial \rho}{\partial t} = Op_t(\rho),$$

onde operador  $Op_t$ , ainda não definido, representa a operação a ser efetuada sobre a função  $\rho_t(x)$ , com atualização infinitesimal que resulte numa redistribuição “virtual” dos indivíduos no espaço de aspecto, indicando que, para construirmos o modelo, será necessário desenvolver representações matemáticas para os processos biológicos que determinam a modificação temporal dos aspectos dos indivíduos.

Vários são os exemplos de espaços de aspecto que podemos citar, dentre os quais destacamos aquele onde os indivíduos possam estar caracterizados por determinada quantidade de

substância que possuam (por exemplo, quantidade de anticorpos em estudos epidemiológicos). Exemplo extraordinário é fornecido por Segel e Perelson (1988), onde os autores utilizam um espaço de aspecto de dimensão infinita na descrição de interações segundo a aparência das formas dos indivíduos (no caso específico, populações de células e proteínas do sistema imunológico), com aplicação a problemas de sociobiologia observados em populações dotadas de comportamento complexo, como a de seres humanos.

O espaço de aspecto abordado aqui é o mais comum dentre os conhecidos, mas protótipo dos demais, alguns, pode-se dizer, de características inusitadas. Trata-se do espaço físico onde o aspecto a ser descrito é a própria localização espacial dos indivíduos que compõem a população, e onde a medida a ser efetuada é a posição de cada indivíduo. Neste caso a distribuição dos indivíduos no espaço de aspecto não é apenas virtual, mas representa efetivamente sua posição espacial.

### **3 Uma descrição fenomenológica do Princípio de Conservação**

É argumento fundamental na descrição de modelos de Dinâmica Populacional. Constitui-se numa contabilidade de entradas e saídas de indivíduos em conjuntos de controle do espaço de aspecto, entendendo-se entrada e saída como uma modificação no aspecto do indivíduo.

Tal modificação deve ter interpretação generalizada, sendo entendida como uma redistribuição dos indivíduos no espaço de aspecto, ou seja, modificações em suas posições virtuais, portanto em seus aspectos, gerando modificações no estado do sistema. São dadas ou por um movimento contínuo dos indivíduos pelo espaço de aspecto, com a alteração na contabilidade ocorrendo pela passagem dos mesmos através da fronteira de cada conjunto de controle, ou pelo surgimento e desaparecimento abrupto de indivíduos no interior de cada um destes conjuntos. Por exemplo, nascimentos e mortes no caso do espaço de aspecto localização espacial, devem ser consideradas como surgimentos e desaparecimentos abruptos, enquanto que o movimento contínuo por este espaço é dado pelo deslocamento espacial efetivo, ou seja, a

modificação mesmo na posição do indivíduo material. Já no caso da distribuição etária, nascimentos ocorrem a partir de um movimento contínuo por uma das fronteiras enquanto morte é dada por retirada abrupta, e, por sua vez, o movimento contínuo pelo espaço de aspecto descreve o envelhecimento dos indivíduos que compõem a população.

Já vimos que são variadas as maneiras de se caracterizar um espaço de aspecto, e devemos fazê-lo de forma inequívoca para que possamos aplicar adequadamente o princípio de conservação.

Deparamo-nos aqui novamente com o problema de se transferir adequadamente informações que ocorrem no nível microscópico, do comportamento individual, para o nível de descrição do comportamento coletivo e macroscópico.

### 3.1 O conceito de Fluxo e Densidade de Fluxo: Relações Constitutivas

O conceito de fluxo foi introduzido por Euler a partir de seu modelo para a dinâmica de fluidos invíscidos, comumente encarado como modelo matemático para a dinâmica do meio contínuo. Sob nosso ponto de vista ele pode ser entendido como dinâmica de populações de partículas discretas continuamente distribuídas, segundo interpretação Lagrangeana.

Em seu modelo, Euler parte da hipótese de que não há contabilizações abruptas no número de partículas (partículas não são criadas ou consumidas de forma abrupta internamente aos conjuntos de controle  $\Omega$ ); ele também considerou a matéria distribuída no espaço físico segundo uma função  $\rho(x, t)$ , representativa da densidade de massa, e, finalmente, uma função vetorial  $\vec{v}(x, t)$  para descrever a velocidade de deslocamento das partículas (alteração no aspecto das partículas, numa conceituação mais geral). Assim, a taxa de mudança no número de indivíduos sobre um conjunto de controle será

$$\frac{d}{dt} \int_{\Omega} \rho(x, t) dx = \int_{\Omega} \frac{\partial}{\partial t} \rho(x, t) dx,$$

e devida unicamente à taxa de passagem de indivíduos pela fronteira  $\partial\Omega$  do conjunto. Derivada positiva significando aumento de indivíduos no interior de  $\Omega$ , leva à adoção de taxa positiva

de passagem para dentro do conjunto, e negativa para fora. À esta taxa de passagem denominamos “fluxo”.

Na sua descrição de fluxo através de fronteiras, Euler introduziu o conceito de *vetor de densidade superficial de fluxo* ( $\vec{J}$ ), definido pela integral de superfície

$$\int_{\partial\Omega} \vec{J} \cdot d\vec{s} = \begin{cases} \text{número “líquido” de indivíduos que atravessam} \\ \text{a fronteira } \partial\Omega \text{ por unidade de tempo: positivo} \\ \text{se para fora e negativo se para dentro,} \end{cases}$$

com  $d\vec{s}$  representando um elemento de superfície com normal exterior a  $\Omega$ .

Assim, escrevemos a hipótese de Euler para a conservação de massa

$$\frac{d}{dt} \int_{\Omega} \rho(x, t) dx = \int_{\Omega} \frac{\partial}{\partial t} \rho(x, t) dx = \int_{\partial\Omega} -\vec{J} \cdot d\vec{s} = - \int_{\Omega} \text{div } \vec{J} dx$$

com a última igualdade decorrendo da aplicação do Teorema de Green, e que no caso unidimensional será dado por

$$\frac{d}{dt} \int_{\Omega} \rho(x, t) dx = - \int_{\Omega} \frac{\partial}{\partial x} J dx,$$

e, então,

$$\int_{\Omega} \left( \frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x} J \right) dx = 0, \text{ em qualquer } \Omega.$$

Portanto, se considerarmos o integrando contínuo sobre todo o espaço de aspecto (regular<sup>2</sup>), obtemos a equação de conservação geral

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x} J dx = 0, \quad \text{ou} \quad \frac{\partial \rho}{\partial t} = - \frac{\partial}{\partial x} J \tag{1.3.1}$$

É interessante observar que, em sua formulação integral, o princípio de conservação abrange também saltos de descontinuidade nos valores do fluxo ao longo de fronteiras, não valendo evidentemente a equação diferencial acima. Situações biológicas onde se observa tal fenômeno são descritas por Fagan e Cantrell (1988) e por Bassanezi e Ferreira Jr. (1988).

---

<sup>2</sup>Espaço com fronteira suave.

Obtém-se desta forma duas funções incógnitas, a saber,  $\rho$  e  $J$ , e apenas uma equação escalar, a apresentada em (1.3.1). Para espaços  $n$ -dimensionais tais funções se traduzem em  $n + 1$  funções, já que o divergente vem dado por

$$\operatorname{div} \vec{J} = \sum_{i=1}^n \frac{\partial}{\partial x_i} J_i,$$

ou seja, um sistema indeterminado. A indeterminação é superada apresentando-se o fluxo como funcional do estado do sistema (ou operador atuando sobre o estado do sistema) por

$$\vec{J}(x, t) = (j(\rho)(x, t))_i \quad (i = 1, 2, \dots, n),$$

chamada de *relação constitutiva*, nome herdado da dinâmica do meio contínuo.

A relação constitutiva, referindo-se a um fluxo, descreve então a movimentação dos indivíduos no espaço de aspecto como função da distribuição dos mesmos.

### 3.2 Fluxo por Transporte

Ou *densidade de fluxo por transporte*, é a relação constitutiva produzida pelo exemplo protótipo do conceito de fluxo (por convecção ou advecção). O movimento é determinado por um campo de velocidades  $\vec{v}(x, t)$  que impele os indivíduos constituintes da população, independentemente de qualquer comportamento deliberado que venham a apresentar (como partículas suspensas em um rio). O fluxo por transporte é dado pela equação

$$\vec{J} = \rho \vec{v}$$

obtida de argumentação geométrica (Bassanezi e Ferreira Jr., 1988). Considerando-se este tipo de fluxo, o princípio de conservação fornece a clássica equação de conservação de massa

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho \vec{v})}{\partial x} = 0 \Rightarrow \frac{\partial \rho}{\partial t} + \vec{v} \cdot \nabla \rho + \rho \operatorname{div}(\vec{v}) = 0,$$

que no caso de campos solenoidais de velocidades, resulta em



$$\frac{\partial \rho}{\partial t} + \vec{v} \cdot \nabla \rho = 0$$

Para o modelo de distribuição em faixa etária temos um “transporte” de velocidade unitária ( $dx/dt = 1$ ) ocasionado por variação temporal (ficamos um ano mais velhos por ano), ou seja, um fluxo na idade. Sendo assim

$$\frac{\partial \rho}{\partial t} = -\frac{\partial \rho}{\partial x},$$

com  $x$  pertencendo ao espaço de aspecto idade-cronológica,  $\frac{\partial \rho}{\partial x}$  a taxa de maturação (em conjuntos de controle macroscópicos, representa o fluxo de passagem de uma faixa etária para outra imediatamente superior), e não considerando ainda, para facilidades de interpretação, retiradas por morte dos indivíduos da população, que podendo ser proporcionais à densidade ( $\rho$ ), com proporcionalidade dada por taxas como funções da idade ( $x$ ), podem ser representadas por um termo de tipo fonte  $f(x, t) = -\mu(x)\rho(x, t)$ .

O fluxo por transporte, por ser involuntário, independe de comportamentos individuais e serve de base conceitual para os fluxos difusivo e por taxa.

### 3.3 Fluxo Difusivo

Os fluxos difusivos têm estudo indispensável na formulação de modelos de dinâmica populacional onde o espaço de aspecto envolvido é o espaço físico, remontando sua história aos movimentos denominados brownianos (Einstein, 1956).

Tais movimentos se referem a aleatoriedade das trajetórias exibidas por partículas em suspensão em um líquido, e foram relacionados por Einstein ao modelo de difusão da físico-química, atribuído a W. Fick, usado na descrição de processos como sedimentação, coagulação etc.

São muitas as situações na natureza nas quais indivíduos representativos de uma dada população, ao se movimentarem, descrevem trajetórias que podem ser consideradas aleatórias (Kareiva e Shigesada, 1983).

Comportamento motil aparentemente inadequado, caso aplicado ao movimento de organismos que, no desenvolver suas tarefas, buscam eficiência no referente ao baixo consumo

de energia e de tempo, na verdade se mostra como estratégia bastante sensata, sendo na maior parte das vezes a menos dispendiosa em termos de tais parâmetros de eficiência. E não fica difícil de entender porque este tipo de movimento, também denominado **busca aleatória** (ou “random walk”, do inglês), se mostra o mais adequado, quando levamos em conta situações, diga-se de antemão nada raras, em que o indivíduo não dispõe de nenhuma informação a respeito da localização do seu objetivo, ou mesmo quando dispõe de informações parciais sobre ele. Nestes casos não é ilógico pensar que se mostraria no mínimo imprudente a insistência na busca em uma única direção e sentido.

Sendo assim, o movimento aleatório, contrastando com o movimento por transporte, pode ser considerado como um comportamento deliberado (Ferreira Jr, 2001) de busca, em condições de ausência ou baixo nível<sup>3</sup> de informações disponíveis (Wolfe et al., 2000).

Em certas publicações, como em artigos (Skellam, 1951) ou livros (Murray (1989) e Edelstein-Keshet (1988)), com os quais nos deparamos, faz-se uso de instrumental matemático estocástico para justificar a expressão para o funcional de fluxo difusivo. Mas há uma argumentação mais simples apresentada originalmente por Einstein (1956), e que se utiliza de conceitos da teoria molecular do calor, para fornecer um ponto de vista do qual o processo de difusão pode ser considerado relacionando-o ao movimento irregular de partículas num solvente. No referido texto, também é investigado o processo de difusão como dependente da distribuição da pressão osmótica e da mobilidade da substância dissolvida no solvente, utilizando-se conceitos da teoria cinético-molecular dos gases.

O fluxo difusivo, ou mais precisamente, a **densidade superficial de fluxo**, para o movimento aleatório é descrito, novamente num caso unidimensional, pela relação constitutiva

$$\vec{J}_{difus}(x, t) = -D \frac{\partial \rho}{\partial x} = -D \nabla \rho,$$

podendo ser entendido como uma medida da assimetria espacial dos indivíduos constituintes da população.

A expressão indica que o fluxo é causado pelo gradiente da densidade, sendo portanto mais intenso quando na direção normal à superfície fronteira do conjunto de controle. Sob o ponto de vista macroscópico, representa um processo de homogeneização nas desigualdades

---

<sup>3</sup>Entenda-se baixo nível como também presença de dificuldades no processamento.

da distribuição. Na dinâmica de populações podemos pensar na ocorrência de uma mescla dos indivíduos da população.

O coeficiente  $D$ , denominado “coeficiente de difusão”, mede duas propriedades distintas: transitabilidade (textura) do meio e capacidade motil dos indivíduos, aspectos tipicamente etológicos.

Observe que, a nível microscópico, o fluxo causado pelo movimento browniano é distinto do fluxo causado pelo transporte. No transporte todos os indivíduos se deslocam numa mesma direção, no aleatório, diferentemente, os indivíduos trocam constantemente de direção, com saldo líquido resultante na direção de menor densidade.

Realça-se a interpretação deste tipo de fluxo como sendo originado de movimento de (auto)transporte no nível do comportamento coletivo, se escrevermos a densidade de fluxo difusivo na forma

$$\vec{J}_{difus}(x, t) = \rho \left( -D \frac{\partial(\ln \rho)}{\partial x} \right) = \rho \vec{v}$$

Utilizando o princípio de conservação, a dinâmica com fluxo difusivo será descrita pela equação diferencial

$$\frac{\partial \rho}{\partial t} = \frac{\partial}{\partial x} \left( D \frac{\partial \rho}{\partial x} \right) = \text{div}(D \nabla \rho)$$

### 3.4 Fluxo por Taxia

Sentir e responder a estímulos do meio no qual se vive é característica inerente a todo ser vivo. Respostas a sugestões ambientais que envolvam movimentos intencionados levam o nome de **taxia**. Segundo Segel (1984b), todo organismo capaz de se locomover responde por taxia a pelo menos um estímulo. Sabe-se que alguns animais são atraídos por luz, temperaturas e altos níveis de certas substâncias químicas (por exemplo, feromônios<sup>4</sup> e nutrientes), ao passo que são repelidos por influências potencialmente perigosas tais como presença de toxinas, extremos de temperatura etc. Portanto, conforme especialidade do organismo e dinâmica envolvida, podemos citar, como tipos de taxia, a aerotaxia, a quimiotaxia, a fototaxia, a termotaxia, a fitotaxia (ocorrendo em dinâmicas planta-herbívoro), e outros.

---

<sup>4</sup>Substâncias químicas envolvidas nos processos de propagação de informações entre membros da espécie

Representa, numa conceituação mais ampla, um aperfeiçoamento da estratégia de busca quando combinado com o processo de difusão, onde a detecção do alvo é feita em etapas consecutivas que intercalam movimentos aleatórios com os de busca direcionada. Esta estratégia é utilizada por insetos no sensoramento de feromônios (Murray (1977) e Assis (2003)).

O deslocamento por *taxia*, em geral, diz respeito, então, ao comportamento motil de indivíduos que dispõem de informações limitadas sobre a localização de seus alvos.

Analisemos, por exemplo, a forma mais presente de sensoramento espacial na literatura especializada. Este tipo de sensoramento, mediado por sinais químicos, é devido à presença de um campo escalar representando a distribuição de certa substância química relacionada ao alvo perseguido (como exemplo, podemos citar a secreção de feromônios na demarcação territorial). A distribuição de moléculas de tal substância tem dinâmica própria, já que pode ser interpretada como constituinte de uma população.

Consideramos, neste caso, a distribuição do campo sinalizador químico representada pela função escalar  $c = c(x, t)$ . Seu campo gradiente determina linhas de corrente que, dadas pela equação diferencial

$$\frac{dx}{dt} = \nabla c(x, t),$$

são como trajetórias virtuais, endereçadas ao alvo, a serem detectadas pelos organismos por meio de prospecções (amostragens) locais. Na prospecção, o organismo realiza movimentos brownianos ele próprio, ou em algum órgão sensorial que possua, vasculhando e obtendo informações locais sobre a densidade da substância alvo em alguns pontos aleatórios, e uma vez de posse do maior gradiente, realiza movimento retílineo deliberado em direção, sentido e intensidade. O movimento é interrompido então para uma nova avaliação local, e assim por diante.

Aparentemente representa (Pennisi, 2000) processos biológicos de aquisição, armazenagem, e processamento de informação dos mais eficientes. Notemos que mesmo se dando o processo todo a nível microscópico, produz efeito macroscópico semelhante ao que se observaria caso as trajetórias do campo gradiente (linhas virtuais) agissem como um fluxo por transporte (em biologia este fenômeno recebe o nome de **taxia** ou **tropismo**). Fazendo uso dessa análise, e numa analogia com o fluxo por convecção, não fica difícil entender a modelagem possível

para esse fascinante e formidável problema biomatemático, que envolve mecânica dos fluidos e teorias de filtragem em diferentes escalas e no mesmo instante de tempo (Murray, 1977).

Supondo, então,  $c$  como uma função representando alguma fonte de atração para organismos, uma força atrativa deverá, então, empurrá-los na direção de regiões de maior interesse. A direção e magnitude do movimento, portanto, serão determinados pelo gradiente de  $c$ . Desta forma, o fluxo líquido, ou fluxo quimiotático, naquela direção, pode ter razoável representação dada por

$$\vec{J} = \rho \chi \nabla c$$

o fator  $\chi$  retrata a relação entre o gradiente na substância  $c$  e a velocidade de migração da população. Em outras palavras, o fluxo por quimiotaxia é assumido proporcional ao gradiente  $\partial c / \partial x$ .

Substituindo na equação de conservação teremos

$$\frac{\partial \rho}{\partial t} = -div(\rho \chi \nabla c)$$

resultando numa equação para atração por  $c$ .

Assim, conforme definido por Keller e Segel (1971), tem-se a *densidade de fluxo por taxia*, representado como

$$\vec{J}_{taxia}(x, t) = \chi \rho \nabla c = \rho(\chi \nabla c)$$

Podemos interpretar macroscopicamente o termo  $\chi \nabla c$  como um campo de velocidades de transporte, onde o coeficiente  $\chi$  representa a sensibilidade dos organismos aos gradientes do campo (Rodrigues, 1998).

Novamente citando Ferreira Jr (2001): “Sensibilidades com respeito a campos de sinais distribuídos, às vezes rudimentares sob o ponto de vista individual, caracterizam comportamentos sociais responsáveis pelo desempenho de tarefas coletivas de extrema complexidade”. Como exemplo, podemos citar fenômenos com formação de estruturas fisiológicas como cicatrizes, processos inflamatórios etc.

Cabe salientar, entretanto, que há outras formas de taxa onde o sinal é transportado por difusão ou convecção. O exemplo usado como modelo, se refere apenas a uma das maneiras de descrevê-la.

### 3.5 Densidade Superficial de Fonte

O último ingrediente para a formulação do princípio de conservação que iremos aplicar, tem sua origem na descrição de fenômenos envolvendo perdas ou ganhos de substâncias, devido a transformações moleculares oriundas de reações químicas, ou por meio de processos físicos como evaporação, sedimentação, flutuação etc.

Permitirá a modificação do estado (ou aspecto) do indivíduo por interação, ou seja, por influência de fatores externos. Representa, matematicamente, modificações abruptas internamente aos conjuntos de controle do espaço de aspecto (os conjuntos sobre os quais se processam as contabilizações de entradas e saídas de indivíduos).

Em analogia com a função densidade de distribuição, possui também definição por meio de integral linear, superficial, ou volumétrica, conforme a dimensão do espaço de aspecto, como segue

$$\int_{\Omega} F(x, t) = \begin{cases} \text{taxa "líquida" de indivíduos: incluídos no} \\ \text{conjunto de controle } \Omega, \text{ no instante } t, \\ \text{se positiva, e excluídos se negativa} \end{cases}$$

Agora, de posse de todos os ingredientes, temos o princípio de conservação

$$\frac{d}{dt} \int_{\Omega} \rho \, dx = \int_{\partial\Omega} -\vec{J} \cdot d\vec{s} + \int_{\Omega} F \, dx, \text{ para todo } \Omega.$$

Para o caso linear podemos reescrevê-lo na forma

$$\int_{\Omega} \frac{\partial}{\partial t} \rho(x, t) dx = \int_{\Omega} -\frac{\partial}{\partial x} J(x, t) + \int_{\Omega} F(x, t) dx,$$

de onde obtemos, na hipótese de continuidade dos integrandos, a igualdade

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x} J - F = 0.$$

Encarando a densidade superficial de fonte como uma operação funcional aplicada sobre  $\rho$ , ou seja  $F = f(\rho)$ , a equação de conservação pode ser representada por

$$Op_t(\rho) = -\frac{\partial}{\partial x}J + F$$

Note que se mostra de absoluta relevância a apresentação do princípio de conservação na forma acima, já que não são raros os casos de fontes densidade dependentes, lineares ou não.

Assim denotado, o princípio de conservação adquire grande simplicidade formal. No entanto descreve uma enorme variedade de modelos biomatemáticos complexos, e cuja especificação depende da caracterização do operador  $Op_t$ , que se apresenta fundamentado nas relações funcionais de densidade de fluxo ( $J = j(\rho)$ ), representando o movimento contínuo dos indivíduos sobre o espaço de aspecto, e densidade de fonte ( $F = f(\rho)$ ), representando a inclusão ou exclusão de indivíduos ao longo do espaço de aspecto.

Um problema com variação espaço-temporal como o apresentado acima, não descreve um sistema dinâmico completamente. Necessita ainda que se forneça suas condições inicial e de fronteira, para podermos viabilizar suas soluções.

Os fenômenos de que tratamos neste trabalho, dizem respeito a problemas apresentando variação espacial apenas, que juntamente com suas condições de contorno (ou fronteira) originam um sistema estático, que também viabiliza soluções. Aspectos relacionados às condições de contorno serão discutidos no decorrer do desenvolvimento do presente trabalho.

### 3.6 Descrição de um Cenário

Passamos agora a descrever uma situação a qual ilustra a aplicação do código produzido. Nela, serão fictícios o parâmetro distribuição de densidade e a região sobre os quais, respectivamente, confrontaremos e executaremos nossas simulações, já que não tivemos acesso a dados experimentais.

Sendo assim, optamos por determinar, ou escolher, as características de tal região de forma a apresentarem a maior generalidade possível de condições que pudessem ser reproduzidas em simulações pelo código. Ficamos evidentemente limitados a um ambiente, ou cenário, que fosse viável na análise dos resultados obtidos, como veremos na seção que trata especificamente de tais experimentos.

Este comportamento experimental, se por um lado não permitiu a comparação com resultados reais, o que naturalmente forneceria suporte comprobatório às conclusões apresentadas, garantiu a execução de testes em ambientes relativamente complexos.

São, naturalmente, muitas as possibilidades de cenários a serem simulados. Na figura que segue apresentamos um deles, com algumas condições já esboçadas.

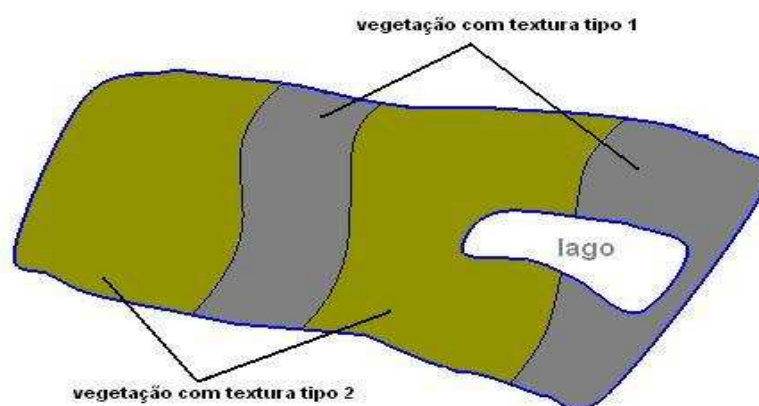


Figura 1.1: características da área geográfica

Faz-se necessário, antes de prosseguirmos, especificarmos melhor o termo cenário num modelo estacionário. Devemos entendê-lo não apenas como uma região ou área geográfica, onde se dará o fenômeno, composta de suas características intrínsecas como dimensão, textura de vegetação eventualmente presente (aspecto que influencia diretamente na capacidade motil da espécie), e presença de barreiras à esta mobilidade, como ocorrência de lagos, lagoas etc. Engloba, o termo, as condições nas fronteiras desta área geográfica, pois estas naturalmente sujeitam a população em estudo, também fontes externas resultantes, por exemplo, de interação humana, e dinâmicas densidade dependentes.

Então, como parte de um possível cenário, a figura procura representar uma área com contorno suave mas com certa irregularidade na forma, ou seja, uma estrutura a qual se aproxima de uma ocorrência real. Denominaremos esta área, ou região, por *domínio*, pois é termo também usado nas considerações analíticas do problema. No exemplo, este domínio se apresenta subdividido em *subdomínios* (o código implementado não limita o número desses subdomínios), intercalando o que podemos supor serem vegetações com diferentes texturas (dois tipos, na figura), habitats de uma mesma espécie animal.

Consideramos que as interfaces entre subdomínios, sobre as quais se dão as chamadas



*condições de compatibilidade*<sup>5</sup>, representam, tão somente, saltos em parâmetros como coeficiente de dispersão e a velocidade no fluxo migratório ou tático. Não atribuímos condições de contorno sobre essas fronteiras-interfaces. Mas o código possibilita que se introduza fontes pontuais<sup>6</sup>, definidas em qualquer parte do domínio, inclusive sobre essas interfaces.

O código também permite que se trabalhe com três diferentes tipos condições de fronteira nos problemas cuja solução aproxima: Dirichlet, Neumann, ou mistas (não há limite no número dessas condições a introduzir, e em suas possíveis combinações ou localizações). Dentre as condições de contorno interessantes a serem simuladas, podemos citar, por exemplo, Dirichlet nula,  $\rho = 0$ , simulando a presença de fronteiras muito distantes num habitat, a qual a espécie habitante não atinja, ou Neumann nula

$$\frac{\partial \rho}{\partial \eta} = 0,$$

simulando a presença de barreiras para a espécie, como regiões alagadas, ou Robin

$$\rho + h \frac{\partial \rho}{\partial \eta} = 0,$$

apresentada por Ludwig et al. (1979), e citada em de Santana (2003), como modelo para um meio hostil porém não letal à população.

Terminamos com uma descrição das dinâmicas vitais que implementamos. O código fornece resultados quando estão presentes três tipos de dinâmica, incluídas por entendermos serem freqüentes na literatura que trata de fenômenos de dispersão e migração, quais sejam: verhulstianas<sup>7</sup>, malthusianas e fontes constantes.

Concluimos este capítulo com a composição de um cenário completo, e conseqüentemente do sistema estático.

---

<sup>5</sup>Tais condições representam nada mais que a continuidade do fluxo entre subdomínios adjacentes.

<sup>6</sup>Tais fontes simulam bem fenômenos como os de inserção de espécies invasoras através de fontes controladas (fenômenos biológicos com interação humana).

<sup>7</sup>Pierre François Verhulst (1804-1849), matemático belga, introduziu a equação que contribui para a não linearidade da EDP em (1.4.1) como modelo de crescimento da população humana em 1838. Não pôde testar a exatidão de seu modelo em virtude dos dados censitários serem inadequados. Esta equação não despertou atenção até R. Pearl (1930) mostrar sua razoável concordância com dados experimentais de populações de *Drosophila melanogaster* (a mosca de frutas).

## 4 O Problema e sua Modelagem

Como já ressaltamos, as situações físicas a serem aqui consideradas, são modeladas por uma equação constitutiva e um princípio de conservação. A equação constitutiva, especificamente neste caso, é a formalização matemática de um modelo de difusão em meio *isotrópico*, em outras palavras, em meio onde a direção do fluxo é paralela à do gradiente da variável de estado (a densidade distribuída espacialmente). O princípio de conservação é representado por uma equação de conservação de massa, da forma descrita na seção anterior. Estas equações quando combinadas resultam numa formulação do problema em termos de densidades populacionais ( $\rho$ ).

Ampliando a generalidade do modelo quanto à possibilidade de descrever fenômenos que ocorram simultaneamente, acrescentamos um termo de transporte podendo representar indiferentemente fluxos por convecção ou taxa.

Para o balanceamento final da equação consideramos, adicionalmente, a presença de fontes externas<sup>8</sup> (ou interativas) e de dinâmicas vitais, incluindo aí os possíveis tipos de dinâmicas e fontes com as quais trabalharemos, representadas indistintamente por  $F = \rho f(\rho)$ , ou seja, como funcional da variável de estado.

Modelamos, ainda, nessa equação diversos fenômenos de decaimento aproximados em conjunto. Numa conceituação um pouco mais geral, trata-se tão somente de uma fonte (malthusiana, na verdade), mas que do ponto de vista biomatemático, pode ser conveniente considerar independente do termo fonte ( $F$ ), quando possuir interpretação distinta, representando, por exemplo, características de hostilidade do habitat.

Essas considerações nos leva a equação

$$-div[D(x, y) \nabla \rho(x, y)] + div(\mathbf{V}(x, y) \rho(x, y)) + b(x, y) \rho(x, y) = \rho(x, y) f(\rho(x, y)),$$

que modela o nosso objeto de estudo, qual seja, um fenômeno estacionário de dispersão populacional no plano, espacialmente dependente, e dinâmica vital para uma única espécie.

---

<sup>8</sup>Não faremos nenhum tipo de discussão matemática formal sobre fontes interativas em linha, ou pontuais, ou mesmo as definidas sobre todo o domínio por funções suaves, no entanto, na seção que trata da discretização do modelo, ficará evidente a forma computacional de as representar e tratar.

Ao considerarmos  $div(\mathbf{V}) = 0$ , ou seja, uma distribuição uniforme do campo de velocidades (equivalentemente, ausência de ação de fontes no interior do mesmo), resulta

$$-div(D \nabla \rho) + \mathbf{V} \cdot \nabla \rho + b \rho = \rho f(\rho),$$

uma equação elíptica onde o termo fonte ( $f(\rho)$ ), apresenta as características desejadas à descrição adequada de cada modelo. Ao introduzirmos o modelo de Verhulst, temos

$$-div(D \nabla \rho) + \mathbf{V} \cdot \nabla \rho + b \rho = \lambda \rho \left[ 1 - \left( \frac{\rho}{K} \right) \right] \quad (1.4.1)$$

Há uma descrição bastante detalhada da dinâmica Verhulstiana em Sossae (1995), e de outros tipos de dinâmicas vitais (ou fontes) neste mesmo texto, ou em Lacaz (1999) de forma bastante didática, e também em Pregnotatto (2002), algumas podendo apresentar características de não linearidade ainda maiores, como nos Modelos de Ayalla, Ehrenfeld e Gilpin (1973), indicado no já citado trabalho de Pregnotatto. A escolha pelo modelo de Verhulst, seguiu da decisão de, ao introduzirmos uma dinâmica vital com características não lineares, darmos a este trabalho complexidade adicional, observáveis tanto no aspecto dos estudos analíticos desenvolvidos, quanto das dificuldades numéricas apresentadas. A escolha de outros modelos não lineares talvez introduzisse fontes de grande especificidade, e assim fugindo ao escopo do presente trabalho, que dentre os múltiplos aspectos do problema Biomatemático que se pretendeu abordar, teve-se as características de intraespecificidade do modelo. Há também que se considerar sua simplicidade, grande aplicabilidade, e o fato de representar modelo precursor de muitos outros.

Na equação aparecem os seguintes coeficientes considerados pelo modelo e que são indicativos dos aspectos do fenômeno estudado:

1.  $D$ : coeficiente de dispersão populacional, medindo o espalhamento da população em unidade de área por unidade de tempo;
2.  $\mathbf{V}$ : campo vetorial surgindo devido a fatores externos como ventos predominantes na região, ou armadilhas de feromônios, ou podendo também ser considerado como uma característica migratória da espécie, medido em unidade de área por unidade de tempo;

3.  $b$ : como aspectos de mortalidade da dinâmica populacional da espécie já figuram no lado direito da equação, consideramos tal coeficiente para descrever a hostilidade do meio à população que se dispersa e migra sobre ele; do ponto de vista de uma situação física geral, podemos, em primeira aproximação, imaginar o termo associado a uma fonte com intensidade proporcional a  $\rho$ , medido em unidade inversa de tempo;
4.  $K$ : capacidade de suporte do meio ou nível de saturação ambiental, medido em unidade de densidade populacional;
5.  $\lambda$ : taxa específica intrínseca de crescimento, ou seja, a variação no tempo da densidade por unidade de densidade, na ausência de quaisquer fatores limitantes, medido em unidade inversa do tempo.

## Capítulo 2

# Análise e Discretização da Equação Elíptica

Introduzimos este capítulo com a formulação variacional do problema que naturalmente precede a demonstração de existência de solução, pois esta última a ela se refere. A demonstração de existência, por sua vez, demandou a inclusão de seção que trata de procedimentos que adimensionalizem a equação. O passo seguinte será a discretização do problema, que fornecerá o embasamento científico para a confecção do nosso código de programação e possibilitará as simulações numéricas. No final do capítulo fazemos uma apresentação do método de resolução do sistema não-linear de equações e, em seguida, uma breve discussão de sua estabilidade.

### 1 Formulação variacional

Em muitas aplicações a formulação variacional se constitui no princípio físico fundamental do problema, e a equação diferencial uma consequência. Exatamente como no caso de estudo de dinâmica de população, em que utilizamos o princípio de conservação já mesmo na descrição do modelo matemático. Portanto, não surpreende encontrarmos em tais aplicações, um forte movimento na direção de minimização de uma integral variacional sobre uma classe de funções, e que se constitui em procedimento inicial para a aplicação do Método de Elementos Finitos. Nesta formulação do problema, tem-se ainda a vantagem de trabalhar com espaços envolvidos menos restritivos quanto à diferenciabilidade da solução e dos coeficientes, que seria necessária à uma formulação clássica.

Consideraremos nosso problema sobre um domínio  $\Omega \subset \mathbb{R}^2$ , com fronteiras pelo menos lipschitzianas, subdividido em subdomínios  $\Omega_1$  e  $\Omega_2$ , separados por uma fronteira-interface  $\Gamma$  de medida nula, representando uma descontinuidade nas características do meio (cf. Seção 3.6, Capítulo 1).

Tomando uma função teste  $v$  suficientemente suave, e uma vez que as segundas derivadas de  $\rho$  não são necessariamente integráveis ao longo da interface  $\Gamma$ , iniciamos a obtenção da formulação variacional integrando separadamente a equação elíptica sobre  $\Omega_1$  e  $\Omega_2$ , como segue:

$$\begin{aligned} & \int_{\Omega_1} [-div(D_1 \nabla \rho) + \mathbf{V}_1 \cdot \nabla \rho + \mu_1 \rho - \rho f(\rho)] v \, d\mathbf{x} \\ & + \int_{\Omega_2} [-div(D_2 \nabla \rho) + \mathbf{V}_2 \cdot \nabla \rho + \mu_2 \rho - \rho f(\rho)] v \, d\mathbf{x} = 0, \end{aligned}$$

Sendo

$$\nabla \cdot (v D \nabla \rho) = div(v D \nabla \rho) = D \nabla \rho \cdot \nabla v + v div(D \nabla \rho),$$

implicando

$$v div(D \nabla \rho) = \nabla \cdot (v D \nabla \rho) - D \nabla \rho \cdot \nabla v,$$

reescrevemos as integrais acima, resultando em:

$$\begin{aligned} & \int_{\Omega_1} (D_1 \nabla \rho \cdot \nabla v + (\mathbf{V}_1 \cdot \nabla \rho) v + \mu_1 \rho v - \rho f(\rho) v) \, d\mathbf{x} \\ & + \int_{\Omega_2} (D_2 \nabla \rho \cdot \nabla v + (\mathbf{V}_2 \cdot \nabla \rho) v + \mu_2 \rho v - \rho f(\rho) v) \, d\mathbf{x} \\ & \quad - \int_{\Omega_1} \nabla \cdot (v D_1 \nabla \rho) \, d\mathbf{x} - \int_{\Omega_2} \nabla \cdot (v D_2 \nabla \rho) \, d\mathbf{x} = 0. \end{aligned}$$

Aplicando o Teorema da Divergência nas duas últimas integrais, resulta a igualdade

$$- \int_{\Omega_1} \nabla \cdot (v D_1 \nabla \rho) d\mathbf{x} - \int_{\Omega_2} \nabla \cdot (v D_2 \nabla \rho) d\mathbf{x} = - \int_{\partial\Omega_1} D_1 \frac{\partial \rho}{\partial \eta} v ds - \int_{\partial\Omega_2} D_2 \frac{\partial \rho}{\partial \eta} v ds.$$

Decompomos agora as duas últimas integrais sobre fronteiras em duas partes, obtendo

$$- \int_{\partial\Omega_1-\Gamma} D_1 \frac{\partial \rho}{\partial \eta} v ds - \int_{\partial\Omega_2-\Gamma} D_2 \frac{\partial \rho}{\partial \eta} v ds + \int_{\Gamma} \left( -D \frac{\partial \rho}{\partial \eta} \right)_1 v ds + \int_{\Gamma} \left( -D \frac{\partial \rho}{\partial \eta} \right)_2 v ds,$$

onde a notação  $\left( -D \frac{\partial \rho}{\partial \eta} \right)_i$ , indica que a componente normal do fluxo estará sendo avaliada sobre a região de fronteira  $\partial\Omega_i$ . Notando que as normais exteriores ( $\eta$ ) das regiões  $\Omega_1$  e  $\Omega_2$  são opostas sobre a fronteira-interface  $\Gamma$ , reescrevemos a última soma de integrais na forma

$$- \int_{\partial\Omega_1-\Gamma} D_1 \frac{\partial \rho}{\partial \eta} v ds - \int_{\partial\Omega_2-\Gamma} D_2 \frac{\partial \rho}{\partial \eta} v ds + \int_{\Gamma} \left( -D_1 \frac{\partial \rho}{\partial \eta} + D_2 \frac{\partial \rho}{\partial \eta} \right) v ds.$$

Ao último integrando impomos a condição de compatibilidade entre subdomínios, que, pelo princípio de conservação<sup>1</sup>, fornece

$$[[\sigma_\eta(s)]] = \sigma_\eta^1(s) - \sigma_\eta^2(s) = 0, \quad s \in \Gamma,$$

onde  $\sigma_\eta^i = -D_i \frac{\partial \rho}{\partial \eta}$ . Retornando e recombinao as integrais, resulta

$$\int_{\Omega} [D \nabla \rho \cdot \nabla v + (\mathbf{V} \cdot \nabla \rho) v + \mu \rho v - \rho f(\rho) v] d\mathbf{x} - \int_{\partial\Omega} D \frac{\partial \rho}{\partial \eta} v ds = 0. \quad (2.1.1)$$

Desta forma, fazendo

$$-\hat{\sigma}(s) + p(s)u(s) = \gamma(s),$$

uma condição mista dada, onde

---

<sup>1</sup>O princípio de conservação é uma afirmação envolvendo o fluxo, a qual requer que em todo subdomínio do problema o fluxo líquido de partículas, penetrando no mesmo, seja zero. Tem como consequência a inexistência de saltos no valor desse fluxo sobre as fronteiras dos referidos subdomínios.

$$\sigma_\eta \equiv \sigma(s) \cdot \eta(s) = \hat{\sigma}(s),$$

com  $s \in \partial\Omega_2$ , e  $\sigma(s)$  o fluxo que cruza a fronteira  $\Gamma$ . E uma vez que assumiremos  $\rho = \hat{\rho}$  sobre  $\partial\Omega_1$ , podemos transformar a equação em (2.1.1), e nosso problema se resumirá a encontrar uma função  $\rho$  tal que  $\rho = \hat{\rho}$  sobre  $\partial\Omega_1$ , e que obedece à formulação

$$\int_{\Omega} [D\nabla\rho \cdot \nabla v + (\mathbf{V} \cdot \nabla\rho) v + \mu \rho v - \rho f(\rho) v] d\mathbf{x} + \int_{\partial\Omega_2} p \rho v ds = \int_{\partial\Omega_2} \gamma v ds, \quad (2.1.2)$$

para toda  $v$ , função teste admissível<sup>2</sup>.

Note que, nesta formulação fraca, se tivermos em vez de uma condição mista, uma condição de Neumann sobre  $\partial\Omega_2$ , esta aparecerá em termos da integral do lado direito da igualdade acima, com  $\gamma = D \frac{\partial\rho}{\partial\eta}$ . Nestas condições a segunda integral do lado esquerdo da mesma igualdade se anula.

A condição de Dirichlet já aparece introduzida no problema advinda da definição da classe de funções admissíveis, pois escolhemos como função teste as funções  $v \in H^1(\Omega)$  tal que  $v = 0$  em  $\partial\Omega_1$ , e então a solução  $u$  também será uma função em  $H^1(\Omega)$  tal que  $\rho = \hat{\rho}$  em  $\partial\Omega_1$ .

Nosso problema variacional pode agora ser estabelecido concisamente da seguinte forma: encontrar a função  $\rho \in H^1(\Omega)$ , tal que  $\rho = \hat{\rho}$  sobre  $\partial\Omega_1$ , e onde a equação

$$\int_{\Omega} [D\nabla\rho \cdot \nabla v + (\mathbf{V} \cdot \nabla\rho) v + \mu \rho v - \rho f(\rho) v] d\mathbf{x} - \int_{\partial\Omega} D \frac{\partial\rho}{\partial\eta} v ds = 0,$$

vale para toda  $v \in H^1(\Omega)$ , tal que  $v = 0$  sobre  $\partial\Omega_1$ .

## 2 Adimensionalização da equação

Consideremos a equação construída no capítulo anterior,

---

<sup>2</sup>Funções, que neste problema, têm derivada de ordem um, ou menos, é quadrado integrável.



$$-div(D\nabla\rho) + div(\mathbf{V}\rho) + b\rho = \lambda\rho\left(1 - \frac{\rho}{K}\right), \quad \text{em } \Omega \subset \mathbb{R}^2.$$

Sendo  $div(\mathbf{V}\rho) = div(\mathbf{V})\rho + \mathbf{V}\cdot\nabla\rho$ , ao assumirmos  $div(\mathbf{V}) = 0$  (campo solenoidal) e  $D$  constante em  $\Omega$ , nós a reescrevemos como

$$-D\Delta\rho + \mathbf{V}\cdot\nabla\rho + b\rho = \lambda\rho\left(1 - \frac{\rho}{K}\right), \quad \text{em } \Omega \subset \mathbb{R}^2.$$

Adimensionalizando esta última equação pela capacidade de suporte, por possuir a mesma igual unidade de medida da variável de estado  $\rho$ , assim temos

$$-D\Delta(\rho/K) + \mathbf{V}\cdot\nabla(\rho/K) + b\frac{\rho}{K} = \lambda\frac{\rho}{K}\left(1 - \frac{\rho}{K}\right)$$

Como a unidade<sup>3</sup> do coeficiente de dispersão vem dada por  $[D] = L^2T^{-1}$ , impõe-se  $[\lambda] = T^{-1}$ . Assim,  $[\sqrt{D/\lambda}] = L^{-1}$  e, desta forma, prosseguindo na adimensionalização, obtemos

$$-\left(\frac{\partial^2(\rho/K)}{\partial(\bar{x}\sqrt{\lambda/D})^2} - \frac{\partial^2(\rho/K)}{\partial(\bar{y}\sqrt{\lambda/D})^2}\right) + \frac{1}{D}\mathbf{V}\cdot\left(\frac{\partial(\rho/K)}{\partial(\bar{x}\lambda/D)}, \frac{\partial(\rho/K)}{\partial(\bar{y}\lambda/D)}\right) + \frac{b}{\lambda}\frac{\rho}{K} = \frac{\rho}{K}\left(1 - \frac{\rho}{K}\right),$$

onde todos os agrupamentos de variáveis e parâmetros são adimensionais. Tomando agora  $u = \frac{\rho}{K}$ ,  $\mu = \frac{b}{\lambda} - 1$ ,  $x = \bar{x}\sqrt{\lambda/D}$ , e  $y = \bar{y}\sqrt{\lambda/D}$ , temos o problema na forma

$$-\Delta u + \frac{1}{D}\mathbf{V}\cdot\nabla u + \mu u = -u^2, \quad \text{em } \Omega \subset \mathbb{R}^2,$$

a ser usada na demonstração de existência que segue na próxima seção.

### 3 Existência de solução

O objetivo principal deste trabalho, como já enunciado, é a criação de uma linguagem algorítmica que se preste à verificação de parâmetros relativos a fenômenos de dinâmica populacional em estados permanentes, ou seja, fenômenos já sem variação no tempo. Mas,

---

<sup>3</sup>Nesta seção  $[.]$ ,  $L$  e  $T$ , representaram, respectivamente, a unidade de medida dos coeficientes e variáveis da equação, unidade de comprimento e unidade de tempo.

acreditando-se se garantir uma sequência lógica na aquisição de crença nos resultados a serem apresentados, discutiremos nesta seção uma existência de solução do problema variacional.

A demonstração que segue, com a condição de fronteira apresentada, deverá então contribuir para embasar a busca por aproximações numéricas. Tais aproximações referir-se-ão ao mesmo operador elíptico, mas em situações também outras de condições de fronteira, sejam estas condições de Neumann ou Dirichlet, consideradas combinadas (ditas mistas, aqui) ou separadamente, e agindo igualmente sobre toda a fronteira, ou de forma diferenciada sobre trechos com interseções vazias dessa fronteira.

A contribuição decorre do fato de que tal demonstração se trata, obviamente, de um indicador da possibilidade de ocorrência dessas soluções numéricas, as quais, vale salientar, possuirão sentido matemático pelo menos, e não necessariamente biológico.

Na busca pela existência de solução fraca do problema elíptico considerado, combinaremos a técnica do Método Variacional com um Teorema de Ponto-Fixo. Esta busca inicia-se reformulando variacionalmente o problema: achar  $u \in \mathcal{C}^2(\Omega)$  tal que

$$P_0 \left\{ \begin{array}{l} -\Delta u + \frac{1}{D}\mathbf{V} \cdot \nabla u + \mu u = -u^2 + f, \text{ em } \Omega \subset \mathbb{R}^2 \\ u|_{\partial\Omega} = 0, \end{array} \right.$$

com  $\partial\Omega$  fronteira suave de  $\Omega$ ,  $f \in L^2(\Omega)$  (fonte externa controlável), e a última igualdade ocorrendo no sentido do traço (cf. apêndice A-1.2.4).

A forma fraca do problema  $P_0$  torna-se: achar  $u \in H_0^1(\Omega)$  tal que

$$P_f \left\{ \begin{array}{l} (\nabla u, \nabla v)_{L^2(\Omega)} + \frac{1}{D}(\mathbf{V} \cdot \nabla u, v)_{L^2(\Omega)} + \mu(u, v)_{L^2(\Omega)} = -(u^2, v)_{L^2(\Omega)} + (f, v)_{L^2(\Omega)} \\ \forall v \in H_0^1(\Omega). \end{array} \right.$$

Observe que o termo  $(u^2, v)_{L^2(\Omega)} = \int u^2 v \, d\Omega$  está bem definido pois  $u \in H_0^1(\Omega) \hookrightarrow L^q(\Omega)$  (imersão contínua), para qualquer  $q$  finito (cf. apêndice A-1.2.5). Assim, especificamente teremos  $u \in H_0^1(\Omega) \hookrightarrow L^4(\Omega)$  e então  $u^2 \in L^2(\Omega)$ .

Agora, lembrando ser  $H_0^1(\Omega)$  um espaço de *Hilbert*<sup>4</sup> separável<sup>5</sup>, ele então admite *Base*

<sup>4</sup>Pois  $H_0^1(\Omega)$  é subespaço fechado do  $L^2(\Omega)$ .

<sup>5</sup>Para uma demonstração de que  $L^2(\Omega)$  é separável, usa-se o conjunto dos polinômios sobre  $\mathbb{R}^n$  tendo

de Schauder, isto é,  $H_0^1(\Omega) = \text{span}\{v_i\}_{i=1}^{+\infty}$ . Desta forma, definimos os subespaços  $H_n = \text{span}\{v_1, \dots, v_n\}$ , onde  $H_1 \subset H_2 \subset \dots \subset H_n \subset \dots \subset H_0^1(\Omega)$ , e para cada  $n$  temos os problemas aproximados: achar  $u_n \in H_n$  tal que

$$P_{f_a} \begin{cases} (\nabla u_n, \nabla \varphi)_{L^2(\Omega)} + \frac{1}{D}(\mathbf{V} \cdot \nabla u_n, \varphi)_{L^2(\Omega)} + \mu(u_n, \varphi)_{L^2(\Omega)} = -(u_n^2, \varphi)_{L^2(\Omega)} + (f, \varphi)_{L^2(\Omega)} \\ \forall \varphi \in H_n. \end{cases}$$

Como  $u_n = \sum_{i=1}^n a_i v_i$ , e o problema acima vale para todo  $\varphi = v_i$ ,  $i = 1, \dots, n$ , o mesmo será equivalente a: achar  $a = (a_1, \dots, a_n)^T \in \mathbb{R}^n$  tal que

$$Aa + \frac{1}{D}Ba + \mu Ca = -G(a) + \mathcal{F}. \quad (2.3.1)$$

Esta última sendo uma equação algébrica não linear onde

$$\begin{aligned} A &= [(\nabla v_i, \nabla v_j)_{L^2(\Omega)}]_{i,j}, \\ B &= [(\mathbf{V} \cdot \nabla v_i, v_j)_{L^2(\Omega)}]_{i,j}, \\ C &= [(v_i, v_j)_{L^2(\Omega)}]_{i,j}, \\ G(a) &= [(\left(\sum_{i=1}^n a_i v_i\right)^2, v_j)_{L^2(\Omega)}]_{i,j} = \sum_{i=1}^n \sum_{l=1}^n a_i a_l [(v_l v_i, v_j)_{L^2(\Omega)}]_j, \\ \mathcal{F} &= [(f, v_j)_{L^2(\Omega)}]_j, \end{aligned}$$

com  $i, j = 1, \dots, n$ .

Mostraremos que o  $P_{f_a}$  tem solução utilizando o *Teorema do Ponto-Fixo de Brower*<sup>6</sup> (cf. apêndice A-1.2.2). Para tanto, notemos que  $u_n$  será solução do  $P_{f_a}$  se for ponto-fixa da transformação

---

coeficientes complexos racionais.

<sup>6</sup>Por simplicidade, nas notações dos espaços  $L^p(\Omega)$  e  $H_1^0(\Omega)$ , omitiremos o domínio  $\Omega$  sobre o qual se encontram definidos.

$$\begin{aligned} T_n : H_n &\longrightarrow H_n \\ w &\longmapsto T_n w := z, \end{aligned}$$

onde  $z \in H_n$  é solução do  $P_{f_a}$  linearizado. Desta forma, linearizando a equação algébrica em (2.3.1), escrevemos

$$A[z] + \mu C[z] = (-1)\{G([w]) + \frac{1}{D}B[w] - \mathcal{F}\}, \quad (2.3.2)$$

onde  $[ \cdot ]$  representa o vetor dos coeficientes de  $z$  ou  $w$ .

$A$  e  $C$  são matrizes simétricas positivas definidas, e, portanto, inversíveis; então dado  $w$  existe um único  $[z]$  solução da equação (2.3.2), e isto garante que  $T_n$  está bem definida.

Devemos agora mostrar a continuidade de  $T_n$ . Para isto, tomemos  $\{w_k\}_{k=1}^{+\infty}$  com  $w_k \rightarrow w$ ,  $w, w_k \in H_n$ , qualquer  $k$  – sempre podemos tomar tal limite, uma vez que  $H_n$  é espaço normado (na norma  $\|\cdot\|_{H_0^1}$ ) de dimensão finita, e portanto completo nesta norma. Agora, do problema linearizado

$$(\nabla z, \nabla \varphi)_{L^2} + \mu(z, \varphi)_{L^2} = (-1) \left\{ (w^2, \varphi)_{L^2} + \frac{1}{D}(\mathbf{V} \cdot \nabla w, \varphi)_{L^2} - (f, \varphi)_{L^2} \right\}, \quad (2.3.3)$$

tomando-se  $T_n w := z$ , temos

$$\begin{aligned} (\nabla T_n w, \nabla \varphi)_{L^2} + \mu(T_n w, \varphi)_{L^2} \\ = (-1) \left\{ (w^2, \varphi)_{L^2} + \frac{1}{D}(\mathbf{V} \cdot \nabla w, \varphi)_{L^2} - (f, \varphi)_{L^2} \right\}, \end{aligned} \quad (2.3.4)$$

e uma vez que  $T_n w_k \in H_n$ , também temos

$$\begin{aligned} (\nabla T_n w_k, \nabla \varphi)_{L^2} + \mu(T_n w_k, \varphi)_{L^2} \\ = (-1) \left\{ (w_k^2, \varphi)_{L^2} + \frac{1}{D}(\mathbf{V} \cdot \nabla w_k, \varphi)_{L^2} - (f, \varphi)_{L^2} \right\}. \end{aligned} \quad (2.3.5)$$

Subtraindo-se (2.3.4) de (2.3.5),

$$\begin{aligned} (\nabla(T_n w_k - T_n w), \nabla \varphi)_{L^2} &+ \mu(T_n w_k - T_n w, \varphi)_{L^2} \\ &= (w^2 - w_k^2, \varphi)_{L^2} + \frac{1}{D}(\mathbf{V} \cdot \nabla(w - w_k), \varphi)_{L^2}, \end{aligned}$$

e fazendo-se  $\varphi = T_n w_k - T_n w$ , encontramos

$$\begin{aligned} \|\nabla(T_n w_k - T_n w)\|_{L^2}^2 &+ \mu\|T_n w_k - T_n w\|_{L^2}^2 \\ &= (w^2 - w_k^2, T_n w_k - T_n w)_{L^2} + \frac{1}{D}(\mathbf{V} \cdot \nabla(w - w_k), T_n w_k - T_n w)_{L^2}, \end{aligned}$$

e, então,

$$\begin{aligned} \|\nabla(T_n w_k - T_n w)\|_{L^2}^2 &+ \mu\|T_n w_k - T_n w\|_{L^2}^2 \\ &= (w^2 - w_k^2, T_n w_k - T_n w)_{L^2} + \frac{1}{D}(\mathbf{V} \cdot \nabla(w - w_k), T_n w_k - T_n w)_{L^2} \\ &\quad \text{(aplicando a desigualdade de Hölder e reagrupando)} \\ &\leq \left\{ \|w^2 - w_k^2\|_{L^2} + \frac{1}{D}\|\mathbf{V} \cdot \nabla(w - w_k)\|_{L^2} \right\} \|T_n w_k - T_n w\|_{L^2} \\ &\quad \text{(aplicando a desigualdade de Young com } \varepsilon = \frac{\mu}{2}\text{)}, \\ &\leq C_\varepsilon \left\{ \|w^2 - w_k^2\|_{L^2} + \frac{1}{D}\|\mathbf{V} \cdot \nabla(w - w_k)\|_{L^2} \right\} + \frac{\mu}{2}\|T_n w_k - T_n w\|_{L^2}^2 \end{aligned}$$

portanto

$$\begin{aligned} \|\nabla(T_n w_k - T_n w)\|_{L^2}^2 &\leq \|\nabla(T_n w_k - T_n w)\|_{L^2}^2 + \frac{\mu}{2}\|T_n w_k - T_n w\|_{L^2}^2 \\ &\leq C_\varepsilon \left\{ \|w^2 - w_k^2\|_{L^2} + \frac{1}{D}\|\mathbf{V} \cdot \nabla(w - w_k)\|_{L^2} \right\} \\ &= C_\varepsilon \left\{ \|w^2 - w_k^2\|_{L^2} + \frac{1}{D}\|\mathbf{V}\|_{L^\infty}^2 \|\nabla(w - w_k)\|_{L^2} \right\}, \end{aligned}$$

Desta forma, e pela desigualdade que obtemos a seguir

$$\begin{aligned}
\|w^2 - w_k^2\|_{L^2}^2 &= \int_{\Omega} |w^2 - w_k^2|^2 dx \\
&= \int_{\Omega} |(w - w_k)(w + w_k)|^2 dx \\
&= \int_{\Omega} |(w - w_k)^2| |(w + w_k)^2| dx \\
&\quad \text{(aplicando a desigualdade de Hölder)} \\
&\leq \|(w - w_k)^2\|_{L^2} \|(w + w_k)^2\|_{L^2} \\
&= \left( \int_{\Omega} |w - w_k|^4 dx \right)^{1/2} \left( \int_{\Omega} |w + w_k|^4 dx \right)^{1/2} \\
&= \|w - w_k\|_{L^4}^2 \|w + w_k\|_{L^4}^2 \\
&\leq C_1 \|w + w_k\|_{H_0^1}^2 C_2 \|w - w_k\|_{H_0^1}^2 \quad (\text{cf. apêndice A-1.2.5}) \\
&= C \|w + w - w + w_k\|_{H_0^1}^2 \|w - w_k\|_{H_0^1}^2 \\
&\leq C (\|2w\|_{H_0^1} + \|w - w_k\|_{H_0^1})^2 \|w - w_k\|_{H_0^1}^2,
\end{aligned}$$

temos a estimativa

$$C_{\varepsilon} \left\{ C (\|2w\|_{H_0^1} + \|w - w_k\|_{H_0^1})^2 \|w - w_k\|_{H_0^1}^2 + \frac{1}{D} \|\mathbf{V}\|_{L^\infty}^2 \|\nabla(w - w_k)\|_{L^2} \right\} \|\nabla(T_n w_k - T_n w)\|_{L^2}^2 \leq$$

Sendo assim, da desigualdade acima e da equivalência<sup>7</sup> entre as normas  $\|x\|_{H_0^1}$  e  $\|\nabla x\|_{L^2}$ , temos  $T_n w_k \rightarrow T_n w$  com  $w_k \rightarrow w$ . Portanto,  $T_n$  é de fato contínua.

Desta forma, para usarmos o Teorema do Ponto-fixo de Brower, basta mostrarmos que

$$T_n : B[R_0] \longrightarrow B[R_0],$$

para algum  $R_0$  adequado.

Para isto, busquemos estimativas adequadas para  $z$  em termos de  $w$ . Assim, voltando ao problema linearizado e colocando  $\varphi = z$  na equação em (2.3.3), ficamos com

---

<sup>7</sup>Segue facilmente da definição da norma em  $H_0^1$  e desigualdade no apêndice A-1.4.3.

$$\|\nabla z\|_{L^2}^2 + \mu \|z\|_{L^2}^2 = (-1) \left\{ (w^2, z)_{L^2} + \frac{1}{D} (\mathbf{V} \cdot \nabla w, z)_{L^2} - (f, z)_{L^2} \right\},$$

Agora, usando a igualdade  $\|w^2\|_{L^2}^2 = \int_{\Omega} |w^2|^2 d\Omega = \int_{\Omega} |w|^4 d\Omega = \|w\|_{L^4}^4$  e a imersão  $H_0^1(\Omega) \hookrightarrow L^4(\Omega)$ , para  $\Omega \subset \mathbb{R}^2$ , de onde tem-se  $\|w^2\|_{L^2}^2 \leq \tilde{C} \|w\|_{H_0^1}^4$ , e também da seqüência de desigualdades abaixo indicadas, a primeira decorrendo da desigualdade de Hölder e a segunda obtida a partir de Young com  $\varepsilon = \mu/4$ , temos

$$\begin{aligned} \|\nabla z\|_{L^2}^2 + \mu \|z\|_{L^2}^2 &\leq \|w^2\|_{L^2} \|z\|_{L^2} + \frac{1}{D} \|\mathbf{V}\|_{L^\infty}^2 \|\nabla w\|_{L^2} \|z\|_{L^2} + \|f\|_{L^2} \|z\|_{L^2} \\ &\leq C \left\{ \|w\|_{H_0^1}^4 + \frac{1}{D} \|\mathbf{V}\|_{L^\infty}^4 \|\nabla w\|_{L^2}^2 + \|f\|_{L^2}^2 \right\} + \frac{\mu}{2} \|z\|_{H_0^1}^2, \end{aligned}$$

ou seja

$$\|\nabla z\|_{L^2}^2 \leq \|\nabla z\|_{L^2}^2 + \frac{\mu}{2} \|z\|_{L^2}^2 \leq C \left\{ \|w\|_{H_0^1}^4 + \frac{1}{D} \|\mathbf{V}\|_{L^\infty}^4 \|\nabla w\|_{L^2}^2 + \|f\|_{L^2}^2 \right\}.$$

E por fim, como anteriormente mencionado, da equivalência entre as normas  $\|x\|_{H_0^1}$  e  $\|\nabla x\|_{L^2}$ , é possível concluirmos a estimativa

$$\|z\|_{H_0^1}^2 \leq C \left\{ \|w\|_{H_0^1}^4 + \frac{1}{D} \|\mathbf{V}\|_{L^\infty}^4 \|w\|_{H_0^1}^2 + \|f\|_{L^2}^2 \right\}.$$

Assim se  $w \in B[R]$ , queremos  $R$  tal que  $z \in B[R]$ . Para isto, devemos ter  $R > 0$ , tal que

$$\|z\|_{H_0^1}^2 \leq C \left\{ R^4 + \frac{1}{D} \|\mathbf{V}\|_{L^\infty}^4 R^2 + \|f\|_{L^2}^2 \right\}. \quad (2.3.6)$$

Basta agora que tomemos  $0 < R_0 < 1$ , tal que para qualquer  $R$ ,  $0 < R \leq R_0$ , tenhamos

$$C \left\{ R^4 + \frac{1}{D} \|\mathbf{V}\|_{L^\infty}^4 R^2 \right\} \leq \frac{1}{2} R^2.$$

Para tais  $R$  temos

$$C \left\{ R^4 + \frac{1}{D} \|\mathbf{V}\|_{L^\infty}^4 R^2 + \|f\|_{L^2}^2 \right\} \leq \frac{1}{2} R^2 + C \|f\|_{L^2}^2,$$

e se tomarmos  $f$  tal que  $C \|f\|_{L^2}^2 \leq \frac{1}{2} R^2$  teremos

$$\|z\|_{H_0^1}^2 \leq R_0^2,$$

e portanto

$$T_n : B[R_0] \longrightarrow B[R_0].$$

Aplicando o Teorema do Ponto Fixo de Brower, existe  $u_n \in B[R_0]$  solução do problema fraco aproximado  $(P_{f_a})$ . Além disso

$$\|u_n\|_{H_0^1} \leq R_0 \tag{2.3.7}$$

Para passarmos ao limite as equações que surgem na formulação do  $P_{f_a}$ , consideramos, a partir da desigualdade em (2.3.7), subsequências tais que:

1.  $u_{n_k} \rightharpoonup u$ , convergência fraca<sup>8</sup> em  $H_0^1(\Omega)$ , ( $\Rightarrow \nabla u_{n_k} \rightharpoonup \nabla u$ , pois, a derivada é um operador contínuo no sentido de distribuições);
2.  $u_{n_k} \rightarrow u$ , convergência forte<sup>9</sup> em  $L^2(\Omega)$ ;
3.  $u_{n_k} \rightarrow u$  q.t.p.<sup>10</sup>  $\Rightarrow (u_{n_k} - u)(u_{n_k} + u) \rightarrow 0$  q.t.p. (pois  $(u_{n_k} + u)$  é limitado)  $\Rightarrow u_{n_k}^2 - u^2 \rightarrow 0$  q.t.p.  $\Rightarrow u_{n_k}^2 \rightarrow u^2$  q.t.p., em  $\Omega$ .

---

<sup>8</sup>Uma vez que  $H_0^1$  é *Banach reflexivo* (cf. Df. em apêndice A-1.1.1) e então a condição em (2.3.7) define uma bola compacta na *topologia fraca* (cf. Df. em apêndice A-1.1.5), de acordo com um teorema consequência do *Teorema de Banach-Alaoglu-Bourbaki* (cf. apêndice A-1.2.3).

<sup>9</sup>Pois  $H_0^1(\Omega) \hookrightarrow L^2(\Omega)$ , e então, temos que toda seqüência limitada em  $H_0^1(\Omega)$  tem subsequência convergindo forte em  $L^2(\Omega)$ .

<sup>10</sup>Como implicação do ítem 2, e sendo definido q.t.p. como convergência pontual.



A partir do primeiro ítem acima, podemos ver que:

$$\begin{cases} (\nabla u_{n_k}, \nabla \varphi)_{L^2} \rightarrow (\nabla u, \nabla \varphi)_{L^2} \\ (u_{n_k}, \varphi)_{L^2} \rightarrow (u, \varphi)_{L^2} \\ (\mathbf{V} \cdot \nabla u_{n_k}, \varphi)_{L^2} \rightarrow (\mathbf{V} \cdot \nabla u, \varphi)_{L^2}, \text{ para qualquer } \varphi \in H_n, \end{cases}$$

faltando provar que

$$(u_{n_k}^2, \varphi)_{L^2} \rightarrow (u^2, \varphi)_{L^2}$$

Para isto, observemos que sendo  $\|u_{n_k}^2\|_{L^2}^2 = \|u_{n_k}\|_{L^4}^4$  e  $H_0^1(\Omega) \hookrightarrow L^4(\Omega)$  ( $\Omega \subset \mathbb{R}^2$ ), temos

$$\|u_{n_k}^2\|_{L^2}^2 = \|u_{n_k}\|_{L^4}^4 \leq C \|u_{n_k}\|_{H_0^1}^4 \leq CR_0^4,$$

podendo-se então, obter uma subsequência  $u_{n_k}^2 \rightharpoonup \xi$  em  $L^2(\Omega)$ <sup>11</sup>, e como também  $u_{n_k}^2 \rightarrow u^2$  *q.t.p.* em  $\Omega$  (limitado), então temos que  $u_{n_k}^2 \rightarrow u^2$ , para  $\xi = u^2$ , em  $L^2(\Omega)$ , conforme *Lema de Lions* (cf. apêndice A-1.3.1). E desta forma

$$(u_{n_k}^2, \varphi)_{L^2} \rightarrow (u^2, \varphi)_{L^2}, \text{ para qualquer } \varphi \in H_n.$$

Finalizando-se o problema de determinação de existência de solução para o  $P_{f_a}$ , sendo  $\{v_i\}_{i=1}^{+\infty}$  uma Base de Schawder de  $H_0^1(\Omega)$ , então  $\overline{\bigcup_k H_k} = H_0^1(\Omega)$ , e, portanto, para qualquer  $v \in H_0^1(\Omega)$ , existe  $v_k \rightarrow v$  em  $H_0^1(\Omega)$ ,  $v_k \in \bigcup_k H_k$ , e assim fazendo  $k \rightarrow \infty$  concluímos:

$$(\nabla u, \nabla v)_{L^2} + \frac{1}{D}(\mathbf{V} \cdot \nabla u, v)_{L^2} + \mu(u, v)_{L^2} = -(u^2, v)_{L^2} + (f, v)_{L^2}, \text{ para qualquer } v \in H_0^1(\Omega) \quad \square$$

É possível construirmos uma outra versão de demonstração de existência para o mesmo problema variacional, bastando que introduzamos, no problema clássico, a mudança de variável

---

<sup>11</sup>Novamente por Banach-Alaoglu-Bourbaki (cf. apêndice A-1.2.3)

$$\Psi = u e^{-(\alpha x + \beta y)},$$

proposta pelo Prof. Yuri Bozhkov, com  $\alpha$  e  $\beta$  constantes reais, o que o modifica fornecendo a equação

$$-\Delta \Psi + k\Psi = -\Psi^2 e^{(\alpha x + \beta y)} + f e^{-(\alpha x + \beta y)},$$

com  $k = \mu + \alpha^2 + \beta^2$ ,  $\alpha = \frac{V_1}{2D}$ ,  $\beta = \frac{V_2}{2D}$ ,  $V_1$  e  $V_2$  componentes de  $\mathbf{V}$ , e onde o termo de transporte não mais nela figura.

Após o que, obtemos um novo problema de contorno: achar  $\Psi$  tal que

$$P_0 \begin{cases} -\Delta \Psi + k\Psi = -\Psi^2 e^{(\alpha x + \beta y)} + f e^{-(\alpha x + \beta y)}, \text{ em } \Omega \subset \mathbb{R}^2 \\ \Psi|_{\partial\Omega} = 0, \end{cases}$$

muito semelhante ao primeiro problema inicial  $P_0$ , apresentado no início dessa seção. A forma fraca desta nova versão de  $P_0$  torna-se: achar  $\Psi \in H_0^1(\Omega)$  tal que

$$P_f \begin{cases} (\nabla \Psi, \nabla v)_{L^2(\Omega)} + k(\Psi, v)_{L^2(\Omega)} = -(\Psi^2 e^{(\alpha x + \beta y)}, v)_{L^2(\Omega)} + (f e^{-(\alpha x + \beta y)}, v)_{L^2(\Omega)} \\ \forall v \in H_0^1(\Omega), \end{cases}$$

e as estimativas necessárias à execução da demonstração são facilmente obtidas, apenas observando que, sendo a exponencial definida sobre  $\Omega$  limitado, existe  $M$  constante tal que  $e^{(\alpha x + \beta y)} \leq M$ ,  $\forall x, y \in \Omega$ .

Não desenvolveremos aqui esta outra demonstração, que possui, obviamente, como principal diferença em relação a apresentada, a eliminação das passagens relativas ao trato com termo de transporte.

Alguns comentários relevantes e interessantes podem ser feitos sobre as várias etapas deste procedimento de demonstração de existência.

Por exemplo, observemos a constante  $C$  na desigualdade em (2.3.6), que sendo independente da dimensão dos subespaços finitos  $H_n$  não tem influência no limitamento das soluções (restritas à bola unitária), nem permite convergência a zero ou a ocorrência de soluções ilimitadas quando tomamos  $n \rightarrow \infty$ . Ou que um estudo mais apurado do sinal no termo reativo ( $-u^2$ ) do problema  $P_0$ , poderia demonstrar uma influência do mesmo sobre suas soluções, talvez garantindo soluções “grandes”, não apenas as limitadas à bola  $R$  da desigualdade (2.3.6) conforme Boldrini (2004).

Um último comentário diz respeito a que mesmo sem se explorar o sinal no termo reativo não-linear, mas somente a ordem da não-linearidade, podemos concluir, novamente da desigualdade (2.3.6), que para pequenos valores de  $w$  estaremos obtendo pequenos valores para a solução do problema, posto que estimamos  $R$  em termos de  $w$ . Ainda mais, de acordo com o que se pode inferir da demonstração, a existência de soluções independe da ordem de seu expoente, já que sabemos estar  $H_0^1(\Omega) \hookrightarrow L^q(\Omega)$ , para  $2 \leq q < \infty$  finito.

## 4 Discretização do problema

O recurso que usamos para obter soluções numéricas para o problema é o dado pelo Método de Galerkin, no qual se utiliza uma seqüência de subespaços de dimensões finitas, aqui gerados pela família de polinômios de Lagrange, formando um sistema completo que se aproxima cada vez mais de  $H^1(\Omega)$ , ou pelo refinamento da malha que discretiza o domínio  $(\Omega)$  sobre o qual obtemos soluções aproximadas, ou pelo aumento do grau dos polinômios aproximantes. Tal escolha visa os procedimentos numéricos do Método de Elementos Finitos.

Não pretendemos reproduzir a teoria completa dos Elementos Finitos, a qual se encontra em leituras como Oden et al. (1981) ou Ciarlet (1987), mas não nos furtamos à apresentação de alguns procedimentos de cálculos realizados, e que tencionam mostrar uma certa versatilidade do código de programação desenvolvido.

Retornemos então ao problema descrito nas seções anteriores, em particular considerando a formulação variacional do problema de valor de contorno resumido no final da Seção 1, e tomemos  $H^1(\Omega)$  denotando a classe de funções admissíveis  $v$  definidas sobre todo o domínio  $\Omega$ . O problema é então encontrar  $u \in H^1(\Omega)$ , com  $u = \hat{u}$  como condição de Dirichlet sobre  $\partial\Omega_1$ , e tal que

$$\begin{aligned} \int_{\Omega} [D\nabla u \cdot \nabla v + (\mathbf{V} \cdot \nabla u)v + b u v - u f(u) v] d\mathbf{x} + \int_{\partial\Omega_2} p u v ds \\ = \int_{\partial\Omega_2} \gamma v ds, \end{aligned} \quad (2.4.1)$$

$v \in H^1(\Omega)$ ,  $v = 0$  sobre  $\partial\Omega_1$ .

Substituindo o domínio  $\Omega$  por  $\Omega_h$ , este último consistindo de uma coleção de elementos finitos  $E$  (quadrados e/ou triângulos) e  $N$  pontos nodais, define-se um subespaço  $H^h$   $N$ -dimensional de  $H^1(\Omega_h)$  pela construção de um conjunto apropriado de funções base  $\varphi_i$ ,  $i = 1, 2, \dots, N$ . O índice  $h$  relaciona-se com as dimensões máximas escolhidas na discretização do domínio espacial.

Sendo as funções base de  $H^h$  contínuas em cada elemento  $E$ , elas não podem modelar eventuais saltos nas propriedades do meio no interior de cada um destes elementos. Portanto, ao gerarmos a malha discretizadora dos subdomínios, devemos levar em conta a localização dos nós e fronteiras dos elementos, de forma a coincidirem com as interfaces nas quais tais saltos ocorrem.

Uma típica função teste será da forma

$$v_h(x, y) = \sum_{i=1}^N v_i \varphi_i(x, y). \quad (2.4.2)$$

Nossa aproximação da equação (2.4.1) consiste então em buscar uma função  $u_h$  em  $H^h$ , onde

$$u_h(x, y) = \sum_{j=1}^N u_j \varphi_j(x, y), \quad (2.4.3)$$

com  $u_j = \hat{u}_j$  em nós sobre  $\partial\Omega_{1h}$ , e

$$\begin{aligned} \int_{\Omega_h} [D\nabla u_h \cdot \nabla v_h + (\mathbf{V} \cdot \nabla u_h)v_h + b u_h v_h - u_h f(u_h)v_h] d\mathbf{x} + \int_{\partial\Omega_{2h}} p u_h v_h ds \\ = \int_{\partial\Omega_{2h}} \gamma v_h ds, \end{aligned} \quad (2.4.4)$$

para qualquer  $v_h \in H^h$ , tal que  $v_h = 0$  sobre  $\partial\Omega_{1h}$ . Aqui,  $\partial\Omega_{1h}$  e  $\partial\Omega_{2h}$  aproximam  $\partial\Omega_1$  e  $\partial\Omega_2$ , respectivamente e no sentido dado acima.

Assim, de (2.4.2), (2.4.3), (2.4.4), e considerando a bilinearidade do produto interno e linearidade do gradiente, temos após rearranjo adequado

$$\begin{aligned} \sum_{j=1}^N \left\{ \int_{\Omega_h} \left[ D\nabla\varphi_j \cdot \nabla\varphi_i + (\mathbf{V} \cdot \nabla\varphi_j)\varphi_i + b \varphi_j\varphi_i - \varphi_j f\left(\sum_{l=1}^N u_l\varphi_l\right) \varphi_i \right] d\mathbf{x} + \int_{\partial\Omega_{2h}} p \varphi_j\varphi_i ds \right\} u_j \\ = \int_{\partial\Omega_{2h}} \gamma \varphi_i ds, \end{aligned}$$

com  $i = 1, 2, \dots, N$ .

Uma vez que

$$u f(u) = \lambda u \left[ 1 - \left( \frac{u}{K} \right) \right],$$

podemos escrever

$$\begin{aligned} \sum_{j=1}^N \left\{ \int_{\Omega_h} [D\nabla\varphi_j \cdot \nabla\varphi_i + (\mathbf{V} \cdot \nabla\varphi_j)\varphi_i + (b - \lambda) \varphi_j\varphi_i] d\mathbf{x} + \int_{\partial\Omega_{2h}} p \varphi_j\varphi_i ds \right\} u_j \\ + \sum_{j=1}^N \left\{ \sum_{l=1}^N \left( \frac{\lambda}{K} \right) \left[ \int_{\Omega_h} \varphi_j\varphi_l \varphi_i d\mathbf{x} \right] u_l \right\} u_j \\ = \int_{\partial\Omega_{2h}} \gamma \varphi_i ds \end{aligned}$$

com  $i=1,2,\dots,N$ . (2.4.5)

Mas cada uma das integrais em (2.4.5) pode ser calculada como uma soma de contribuições fornecidas por cada elemento convexo e finito da malha. A compreensão de parte deste procedimento, em particular a forma como se adicionam todas as possibilidades de condições de contorno e tipos de fontes internas, permite entender a já citada versatilidade do código de programação escrito para se obter os resultados aproximados. Assim, denotando por  $\Omega_e$

cada um dos elementos da malha, e tomando  $u_h^e$  e  $v_h^e$  como restrições das aproximações de  $u_h$  e  $v_h$  sobre  $\Omega_e$ , e ainda  $\Psi_h^e$  como uma restrição local às funções de base  $\varphi_j$ , chegamos ao que denominamos de sistema de matrizes de contribuição local

$$\sum_{j=1}^{N_e} m_{ij}^e u_j^e + \sum_{j=1}^{N_e} n_{ij}^e(u_h^e) u_j^e = -\sigma_i^2 \quad , \quad i = 1, 2, \dots, N_e, \quad (2.4.6)$$

com  $N_e$  sendo o número de nós em  $\Omega_e$ ,  $n_{ij}^e(u_h^e)$  denotando ser cada elemento  $n_{ij}^e$  função de  $u_h^e$ , e onde

$$\begin{aligned} m_{ij}^e &= \int_{\Omega_e} \left[ D \left( \frac{\partial \Psi_i^e}{\partial x} \frac{\partial \Psi_j^e}{\partial x} + \frac{\partial \Psi_i^e}{\partial y} \frac{\partial \Psi_j^e}{\partial y} \right) + \left( V_1 \frac{\partial \Psi_j^e}{\partial x} + V_2 \frac{\partial \Psi_j^e}{\partial y} \right) \Psi_i^e + (b - \lambda) \Psi_i^e \Psi_j^e \right] d\mathbf{x} \\ n_{ij}^e &= \sum_{l=1}^{N_e} \left( \frac{\lambda}{K} \right) \left[ \int_{\Omega_e} \Psi_i^e \Psi_l^e \Psi_j^e d\mathbf{x} \right] u_l \\ \sigma_i^e &= \int_{\partial\Omega_e} \sigma_\eta \Psi_i^e ds \end{aligned}$$

com  $i = 1, 2, \dots, N$ . Aqui,  $m_{ij}^e$  e  $n_{ij}^e$  somados formarão os elementos da matriz de rigidez associada ao elemento  $\Omega_e$ , ou seja, matrizes que representam contribuições locais à matriz de rigidez global, enquanto que  $\sigma_i^e$  deverá compor o que chamamos de vetor de fluxo elementar, também uma contribuição local, obtido associando-se ao nó  $i$  de  $\Omega_e$  a última integral acima, lembrando que  $\sigma_\eta$  denota o fluxo ortogonal à fronteira  $\partial\Omega_e$ .

Formalmente, a matriz global do sistema de equações (2.4.5) é obtida somando-se o sistema (2.4.6) sobre todos os  $E$  elementos da malha. Neste procedimento a soma das integrais de contorno pode ser escrita na forma

$$\sum_{e=1}^E \Sigma_i^e = S_i^{(0)} + S_i^{(1)} + S_i^{(2)} \quad , \quad i = 1, 2, \dots, N_e$$

onde

$$\begin{aligned}
S_i^{(0)} &= \sum_{e=1}^E \int_{\partial\Omega_e - \partial\Omega_h} \sigma_\eta \varphi_i ds \\
S_i^{(1)} &= \int_{\partial\Omega_{1h}} \sigma_\eta \varphi_i ds \\
S_i^{(2)} &= \int_{\partial\Omega_{2h}} \sigma_\eta \varphi_i ds
\end{aligned} \tag{2.4.7}$$

Aqui  $\partial\Omega_e - \partial\Omega_h$  representa a porção de fronteira  $\partial\Omega_e$  de  $\Omega_e$  que não coincidem com  $\partial\Omega_h$ , i.e., as fronteiras interelementos.

As quantidades em (2.4.7) são interpretadas como segue:

1.  $S_i^{(0)}$ : definida somente em nós interiores aos domínios, apresentando a forma

$$S_i^{(0)} = \sum_{e=1}^E \int_{\partial\Omega_e - \partial\Omega_h} \sigma_\eta \varphi_i ds = \int_{\Gamma_1} [[\sigma_\eta]] \varphi_i ds + \dots + \int_{\Gamma_M} [[\sigma_\eta]] \varphi_i ds,$$

onde  $M$  é o número total de fronteiras interelementos tendo  $i$  como nó em comum e  $[[\sigma_\eta]]$  a compatibilidade de fluxo entre as fronteiras. Considerando-se a conservação de fluxo através de interfaces onde não se aplicam fontes pontuais ou fontes em linha, teremos  $[[\sigma_\eta]] = 0$ , mas também serão nulas as integrais no somatório acima, onde as  $\varphi_i$ 's dão contribuição nula. No caso de fontes pontuais a formulação variacional não se aplica estritamente, no entanto, podemos incluir seus efeitos da seguinte maneira: supomos que  $F = u f(u)$  tenha a forma

$$F(x, y) = \bar{F}(x, y) + \hat{F} \delta(x - x_i, y - y_i),$$

onde  $\bar{F}$  é a parte integrável e  $\hat{F} \delta(x - x_i, y - y_i)$  denota a fonte pontual de intensidade  $\hat{F}$  aplicada no ponto  $(x_i, y_i) \in \Omega_h$ , com a malha sendo convenientemente construída de forma a que os nós se localizem onde fontes pontuais atuam. Estes dois tipos de fontes estão previstas no código.

2.  $S_i^{(1)}$ : sendo  $\partial\Omega_{1h}$  a região onde se prescrevem as condições de Dirichlet, não é conhecido *a priori* o valor do fluxo nesta fronteira, no entanto, após todas as densidades  $u_1, u_2, \dots, u_N$

terem sido determinadas, obtemos uma aproximação do fluxo ortogonal a esta fronteira. No código esta condição representa a remoção das linhas e colunas associadas às posições dos nós da fronteira apresentando uma tal condição. Isto é equivalente a tomarmos  $v \in H_0^1(\Omega)$ , ou seja,  $v \equiv 0$  sobre  $\partial\Omega_1$ , o que acarreta  $v_i = 0$ , com o índice  $i$  percorrendo todos os nós sobre a fronteira  $\partial\Omega_{1h}$ .

3.  $S_i^{(2)}$ : sendo  $\partial\Omega_{2h}$  a região onde se prescrevem condições mistas, ou seja

$$\sigma_n(s) = p(s) u_h(s) - p(s) \hat{u}_h(s) = p(s) u_h(s) - \gamma(s)$$

Então, aproximadamente

$$S_i^{(2)} \approx \int_{\partial\Omega_{2h}} \left( p \sum_{j=1}^N u_j \varphi_j - \gamma \right) \varphi_i ds = \sum_{j=1}^N P_{ij} u_j - \gamma_i,$$

onde

$$\begin{aligned} \gamma_i &= \int_{\partial\Omega_{2h}} \gamma \varphi_i ds = \sum_{e=1}^E \int_{\partial\Omega_{2h}^e} \gamma \varphi_i ds = \sum_{e=1}^E \gamma_i^e \\ P_{ij} &= \int_{\partial\Omega_{2h}} p \varphi_i \varphi_j ds = \sum_{e=1}^E \int_{\partial\Omega_{2h}^e} p \varphi_i \varphi_j ds = \sum_{e=1}^E P_{ij}^e \end{aligned}$$

sendo  $\partial\Omega_{2h}^e$  a porção de  $\partial\Omega_e$  intersectando  $\partial\Omega_{2h}$ .

## 5 Métodos Iterativos de Resolução: estudo da convergência

A equação em (2.4.5) nos remete ao sistema não-linear

$$F(\mathbf{u}) = [\mathbf{M} + \mathbf{N}(\mathbf{u})] \cdot \mathbf{u} - \boldsymbol{\Sigma} = 0 \quad , \mathbf{u} = (u_1, u_2, \dots, u_N),$$



com  $\mathbf{N}(\mathbf{u})$  denotando a dependência em  $\mathbf{u}$  da matriz  $\mathbf{N}$ .

A solução deste sistema será obtida pelo Método de Newton, com convergência quadrática (cf. apêndice A-1.2.1), que no caso específico tem algumas de suas condições de convergência ( também cf. apêndice A-1.2.1) possíveis de serem verificadas *a priori*.

Como primeira dessas condições, garantimos  $F : \mathbb{R}^N \rightarrow \mathbb{R}^N$  como aplicação continuamente diferenciável num aberto convexo  $D \subset \mathbb{R}^N$ , pois, uma vez que cada um de seus elementos é dado por

$$\begin{aligned} F_i &= \sum_j m_{ij} u_j + \frac{\lambda}{K} \sum_j \left\{ \int_{\Omega_h} \left( \sum_l u_l \varphi_l \right) \varphi_j \varphi_i \, d\mathbf{x} \right\} u_j - \Sigma_i \\ &= \sum_j m_{ij} u_j + \frac{\lambda}{K} \int_{\Omega_h} \left( \sum_j \sum_l u_l u_j \varphi_l \varphi_j \right) \varphi_i \, d\mathbf{x} - \Sigma_i \end{aligned}$$

com  $i = 1, 2, \dots, N$ ,

e, então, lembrando que são polinômios as funções base  $\varphi_k$ ,  $k = 1, 2, \dots, N$ , temos

$$\begin{aligned} \frac{\partial F_i}{\partial u_\mu} &= m_{i\mu} + \frac{\lambda}{K} \int_{\Omega_h} \left[ \sum_j \left( \frac{\partial}{\partial u_\mu} \sum_l u_l \varphi_l \right) u_j \varphi_j + \sum_l \left( \frac{\partial}{\partial u_\mu} \sum_j u_j \varphi_j \right) u_l \varphi_l \right] \varphi_i \, d\mathbf{x} \\ &= m_{i\mu} + \frac{\lambda}{K} \int_{\Omega_h} \left( \sum_j u_j \varphi_j \varphi_\mu + \sum_l u_l \varphi_l \varphi_\mu \right) \varphi_i \, d\mathbf{x} \\ &= m_{i\mu} + 2 \frac{\lambda}{K} \int_{\Omega_h} \left( \sum_j u_j \varphi_j \right) \varphi_i \varphi_\mu \, d\mathbf{x} \\ &= m_{i\mu} + 2 n_{i\mu}(u_h) \end{aligned}$$

com  $i, \mu = 1, 2, \dots, N$ , (2.5.1)

ou seja, as derivadas primeiras são novamente elementos polinomiais, portanto, contínuas em  $D$ , e assim temos  $F \in \mathcal{C}^1(D)$ .

Como segunda condição, possível de verificação *a priori*, mostremos que a matriz Jacobiana de  $F$  é Lipschitz numa vizinhança  $N(\mathbf{u}, r)$ ,  $r > 0$ . Para tanto, basta observarmos que sendo esta matriz definida por

$$J_F(\zeta) = \left( \frac{\partial F_\mu(\zeta)}{\partial u_\nu} \right)_{\mu, \nu=1}^N,$$

tem seus elementos dados por (2.5.1), desta forma, com derivadas contínuas nesta vizinhança, fornecendo  $J_F \in \mathcal{C}^1(N(\mathbf{u}, r), r > 0)$ . Portanto, a matriz Hessiana

$$H_F(\zeta) = \left( \frac{\partial^2 F(\zeta)}{\partial u_\nu \partial u_\mu} \right)_{\mu, \nu=1}^N$$

existe e, então,

$$|J_F(x) - J_F(y)| \leq |H_F(\zeta)| |x - y|,$$

com  $\zeta$  ponto entre  $x$  e  $y$ , para todo  $x, y$  em  $N(\mathbf{u}, r)$ ,  $r > 0$ .

Uma deficiência do Método de Newton é que o mesmo necessita, além naturalmente das condições expostas e avaliadas acima, de uma boa aproximação inicial ( $\mathbf{u}^0$ ) para assegurar sua convergência no processo de iterações. Esta boa aproximação inicial será obtida usando-se uma técnica de máxima descida, ou “Steepest Descent” (Burden e Faires (1993) pg-551), que determina o mínimo local, mas com aproximação linear apenas.

A conexão entre a minimização de funções de  $\mathbb{R}^N$  em  $\mathbb{R}$  e a solução de um sistema de equações não lineares, dá-se pelo fato de um sistema da forma

$$\begin{aligned} f_1(u_1, u_2, \dots, u_N) &= 0 \\ f_2(u_1, u_2, \dots, u_N) &= 0 \\ &\vdots \\ f_N(u_1, u_2, \dots, u_N) &= 0, \end{aligned}$$

ter uma solução em  $\mathbf{u} = (u_1, u_2, \dots, u_N)^T$  precisamente quando a função  $g : \mathbb{R}^N \rightarrow \mathbb{R}$  definida por

$$g(u_1, u_2, \dots, u_N) = \sum_{i=1}^N [f_i(u_1, u_2, \dots, u_N)]^2,$$

tem um valor mínimo em zero.

Sendo a função  $g : \mathbb{R}^N \rightarrow \mathbb{R}$  polinomial, será  $G$ -diferenciável em  $\mathbf{u} \in \mathbb{R}^N$  (cf. Df. em apêndice A-1.1.6), portanto,

$$\lim_{\alpha \rightarrow 0} \left( \frac{1}{\alpha} \right) [g(\mathbf{u} - \alpha \nabla g(\mathbf{u})) - g(\mathbf{u})] + \|\nabla g(\mathbf{u})\|^2 = 0.$$

Segue que podemos escolher um  $\delta > 0$  tal que para todo  $\alpha \in (0, \delta)$  teremos

$$\left( \frac{1}{\alpha} \right) [g(\mathbf{u} - \alpha \nabla g(\mathbf{u})) - g(\mathbf{u})] + \|\nabla g(\mathbf{u})\|^2 < \|\nabla g(\mathbf{u})\|^2,$$

e desta forma, como, por definição,  $g(\mathbf{u}) \geq 0$ , para todo  $\mathbf{u} \in \mathbb{R}^N$ , a seqüência gerada pelo método será monótona limitada, ou seja,

$$0 \leq g(\mathbf{u}^{k+1}) \leq g(\mathbf{u}^k), \quad k = 0, 1, \dots,$$

portanto, convergente.

Notemos que o “Steepest Descent” sempre convergirá, independentemente da escolha feita para  $\mathbf{u}^0$ . No entanto, convém lembrar que dada a definição de  $g$ , a mesma possuirá termos polinomiais de primeiro a quarto grau, podendo o método, portanto, convergir para pontos que não o mínimo global. O que, obviamente, compromete a avaliação feita pelo método de Newton.

# Capítulo 3

## Resultados, Análise e Discussão

Neste capítulo apresentamos a simulação de dois cenários, um dos quais tendo por objetivo o de determinar o coeficiente de dispersão de um fenômeno com características específicas. Também são feitos alguns testes de validação do código, no sentido de adquirirmos confiança de que serão válidas as soluções aproximadas da equação elíptica, quando todos os termos de interesse nela figurarem. Aproveitamos os mesmos testes para efetuarmos comparações entre soluções obtidas a partir de aproximações com elementos finitos de primeira e segunda ordem, e entre diferentes refinamentos nas malhas de discretização dos domínios.

### 1 Testes de validação do código

Busca-se, efetuando estes testes, evitar a possibilidade de ocorrência de erros ao desenvolver o código de programação, dado que a presença de dinâmicas vitais não-lineares dificulta sobremaneira o processo de discretização da equação elíptica e, principalmente, sua implementação computacional. Convém frizarmos que testes comparativos com soluções analíticas dessa equação, quando nela constam todos os termos necessários ao desenvolvimento deste trabalho, incluindo a presença de dinâmica vital não-linear, não se apresentam, pois tais soluções não são ainda possíveis de se obter, não figurando, com certeza, em literatura matemática conhecida.

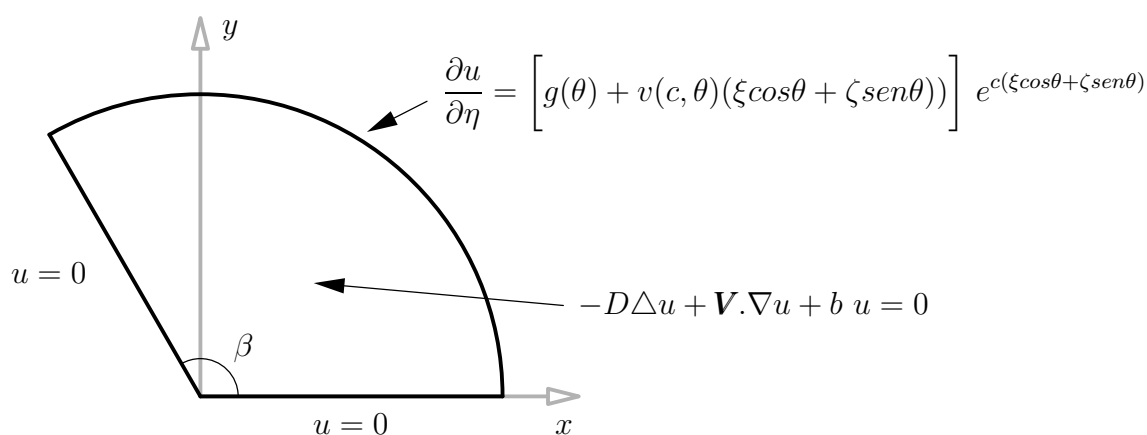
Desenvolvemos aqui um número razoável destes testes de validação, buscando abranger todos os termos da equação diferencial e todos os tipos de condições de contorno. Assim procedendo, avaliamos, quanto aos resultados obtidos, todas as rotinas do código relacionadas

aos cálculos matemáticos e procedimentos computacionais.

No que se refere à condição mista, por ser uma combinação linear entre condições de Dirichlet e Neumann, não a apresentamos diretamente nestes testes, pois sendo executadas separadamente as rotinas de inserção de condições de Dirichlet e o cálculo das integrais onde figuram as condições de Neumann<sup>1</sup>, isto permite que a condição mista seja avaliada de forma indireta.

Desta forma, nos limitamos a dois testes comparativos e de validação distintos, como seguem. Os resultados apresentados, ocorrendo sobre os domínios considerados, são canônicos e nos permitiram um número reduzido de soluções testes.

Para uma primeira avaliação, obtivemos a solução analítica do problema apresentado de forma completa na figura abaixo

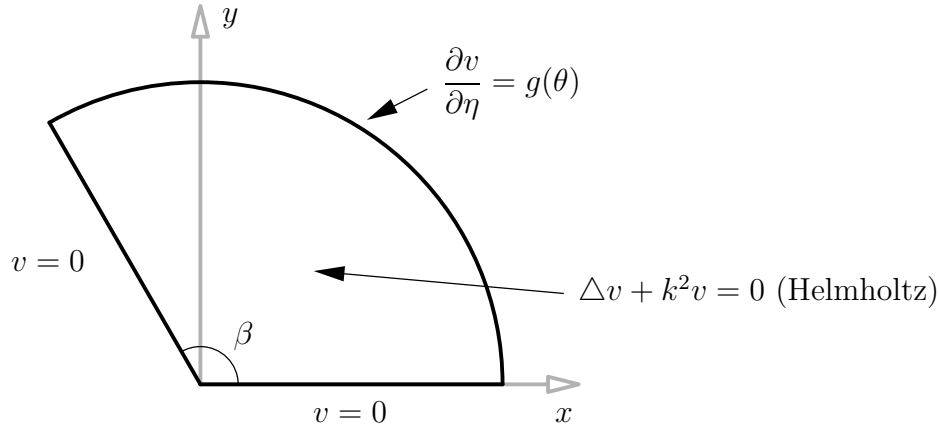


onde  $0 \leq \theta \leq \beta$ ,  $\xi$  e  $\zeta$  são constantes reais,  $c$  representa o raio do setor circular da figura acima, e  $v(c, \theta) = u(c, \theta)e^{-c(\xi \cos \theta + \zeta \sin \theta)}$ .

Com a mudança de variável  $v = u e^{-(\xi x + \zeta y)}$ , a equação diferencial do problema acima transforma-se numa *equação de Helmholtz* aplicada ao domínio com contorno e condições que seguem

---

<sup>1</sup>Condições de Neumann são avaliadas pelo código como uma fonte material de densidade linear distribuída sobre a fronteira, como o que ocorre com fontes, interpretadas na maneira convencional, para o equivalente em domínios unidimensionais. Este procedimento possibilita que se utilize as mesmas rotinas em aproximações de soluções tanto para problemas unidimensionais como bidimensionais



onde  $k^2 = -\frac{b}{D} - \xi^2 - \zeta^2$ ,  $\xi = \frac{V_1}{2D}$ ,  $\zeta = \frac{V_2}{2D}$ , e  $V_1$  e  $V_2$  componentes de  $\mathbf{V}$ . Sua solução em coordenadas polares é dada por

$$v(r, \theta) = \sum_{n=1}^{\infty} b_n J_{\left(\frac{n\pi}{\beta}\right)}(kr) \text{sen}\left(\frac{n\pi}{\beta}\theta\right),$$

onde

$$\left[ \frac{n\pi}{c\beta} J_{\left(\frac{n\pi}{\beta}\right)}(kc) - k^2 J_{\left(\frac{n\pi}{\beta}+1\right)}(kc) \right] b_n = \frac{2}{\beta} \int_0^{\beta} g(\theta) \text{sen}\left(\frac{n\pi}{\beta}\theta\right) d\theta,$$

com  $J_{\mu}$  uma *função de Bessel* de ordem  $\mu$ . Para  $g(\theta)$  escolhemos uma constante  $k_1$ , neste nosso teste, resultando

$$\frac{2}{\beta} \int_0^{\beta} g(\theta) \text{sen}\left(\frac{n\pi}{\beta}\theta\right) d\theta = \frac{4k_1}{n\pi}, \quad n = 1, 3, 5, \dots$$

Procedemos então à comparação entre as soluções aproximada e analítica deste problema. Como não há relevância em analisarmos os resultados gerados, mas tão somente compará-los, avaliando, assim, todas as rotinas executadas pelo código, escolhemos valores aleatórios, porém não nulos, para os parâmetros da equação. Utilizamos os seguintes valores:  $D = -1m^2s^{-1}$ ,  $V_1 = -3ms^{-1}$ ,  $V_2 = 2ms^{-1}$ , e  $b = 5s^{-1}$ ; e os desvios relativos, cometidos nas aproximações numéricas, resultando da fórmula

$$\mathcal{DR} = \frac{\|\mathbf{u}_{\text{aprox}} - \mathbf{u}_{\text{analit}}\|}{\|\mathbf{u}_{\text{analit}}\|}$$

Obtemos a tabela abaixo a partir de seis testes comparativos, oriundos de três refinamentos de malha. O último refinamento sendo significativamente maior que os dois primeiros, como se pode ver, e onde, para cada um dos refinamentos, nos utilizamos de duas distintas ordens de elementos triangulares no método de Elementos Finitos. Aparentemente, a partir de um certo nível de refinamento da malha discretizante, passamos a operar com matrizes de rigidez de elevados números de condição. Desta forma, prejudicando as respostas obtidas.

Tabela 3.1: desvios relativos cometidos nas aproximações de uma solução analítica

número de elementos	ordem dos elementos	$\mathcal{DR}$	$condest(\mathbf{M})$
24 triângulos	1 <sup>a</sup>	0.1730	112.7
	2 <sup>a</sup>	0.1335	585.3
228 triângulos	1 <sup>a</sup>	0.0639	1011.9
	2 <sup>a</sup>	0.0543	5441.2
600 triângulos	1 <sup>a</sup>	0.0148	2638.2
	2 <sup>a</sup>	0.0505	14908
861 triângulos	1 <sup>a</sup>	0.0055	3809
	2 <sup>a</sup>	0.0488	21455

Os resultados dos dois testes feitos com elementos finitos de segunda ordem, são apresentados, de forma comparativa, nas Figuras 3.1 e 3.2, e Figuras 3.3 e 3.4.

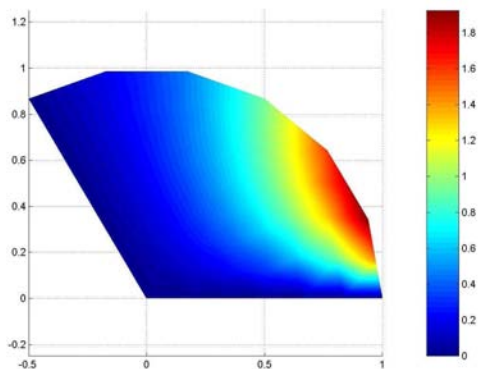


Figura 3.1: solução analítica (malha com 24 triângulos)

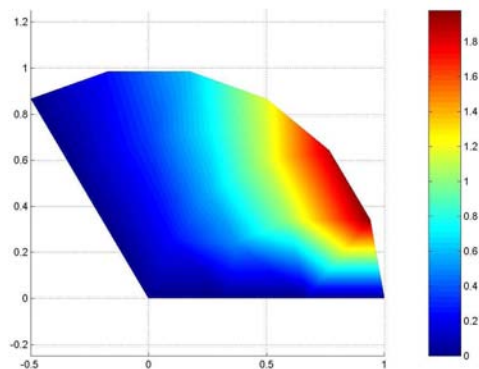


Figura 3.2: solução aproximada (malha com 24 triângulos)

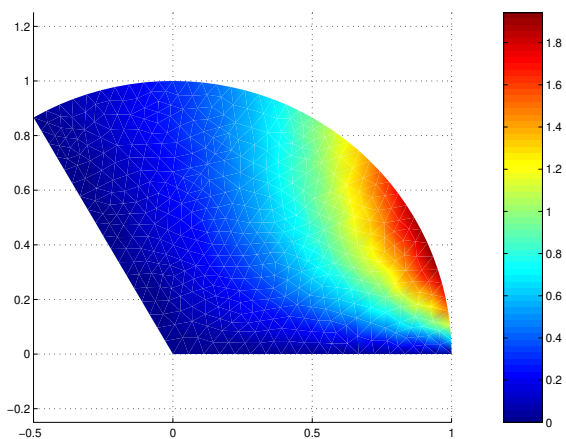


Figura 3.3: solução analítica (malha com 228 triângulos)

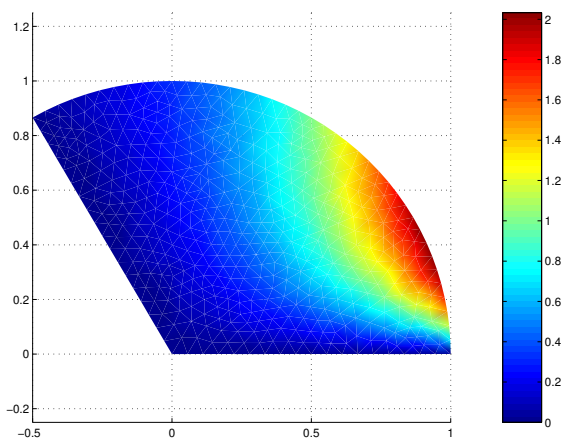


Figura 3.4: solução aproximada (malha com 228 triângulos)

Nas Figuras 3.5 e 3.6, fazemos um comparação mais “direta” entre as respostas obtidas no teste com 24 elementos triangulares de primeira e segunda ordem, respectivamente, inserindo num único gráfico todas as cotas de densidade populacional (resultados aproximados e analíticos), calculadas sobre cada um dos nós da malha de discretização do domínio. Os resultados são, obviamente, avaliados de forma pontual sobre estes nós, mas, para efeito de



melhor comparação, unimos os pontos discretos por segmentos, dando um aspecto visual falso de continuidade aos gráficos mas elucidativo quanto à qualidade das aproximações.

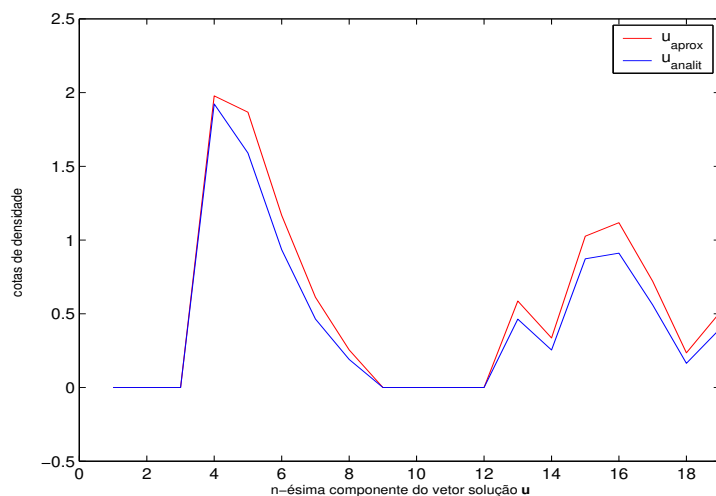


Figura 3.5: solução analítica ( $\mathbf{u}_{analit}$ ) vs aproximada ( $\mathbf{u}_{aprox}$ ), com elementos de 1ª ordem

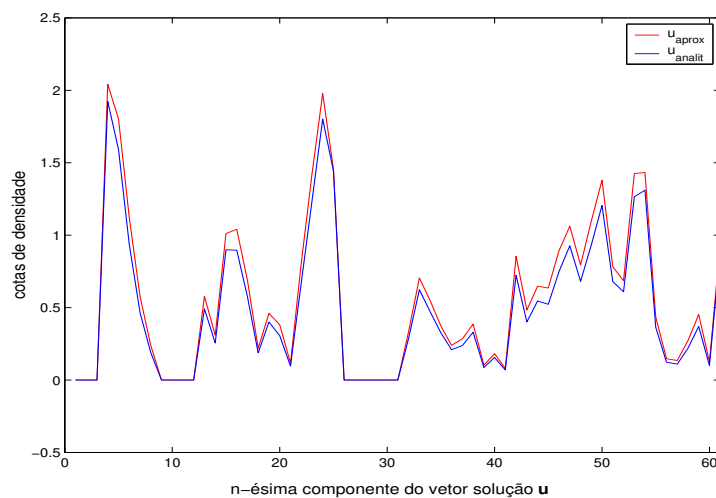


Figura 3.6: solução analítica ( $\mathbf{u}_{analit}$ ) vs aproximada ( $\mathbf{u}_{aprox}$ ), com elementos de 2ª ordem

O segundo teste de validação foi obtido com base nos resultados de simetria de Gidas et al. (1981) e de Lions (1981), os quais implicam numa solução  $u$  da equação diferencial

$$-\Delta u = \lambda u \left( 1 - \frac{1}{K} u \right),$$

ocorrendo num círculo com centro na origem, da forma  $u(x, y) = v(r)$ ,  $r = \sqrt{x^2 + y^2}$ , ou seja, radialmente simétrica.

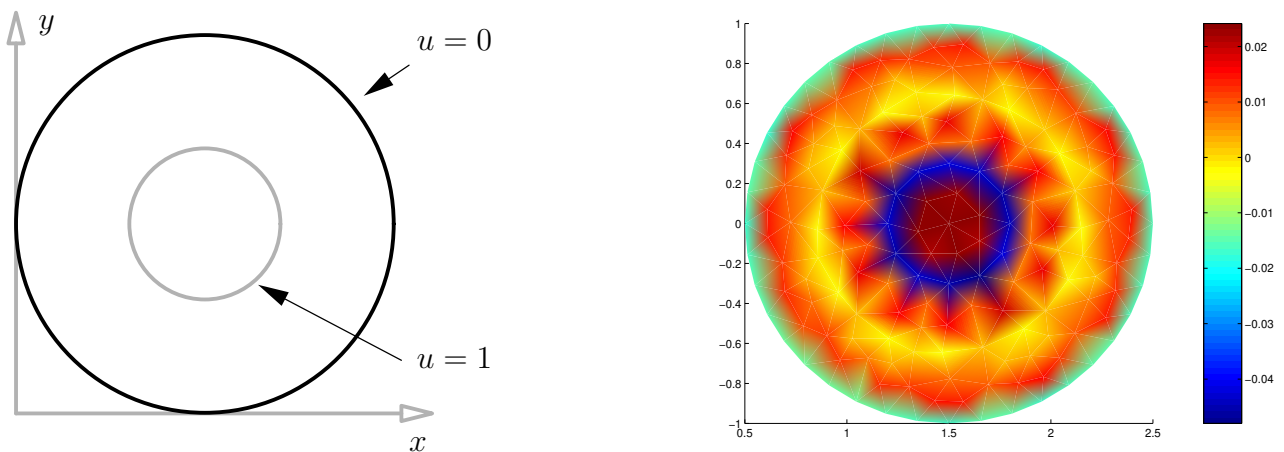


Figura 3.7: solução aproximada com simetria radial

A simetria (ver Figura 3.7) será tanto melhor quanto maior a ordem dos elementos finitos e o refinamento na discretização do domínio.

É certo que os resultados expostos neste capítulo nos garantem qualidade nas respostas fornecidas pelo programa, no que se refere à melhor precisão obtida com elementos finitos de maior grau, ou maior refinamento nas malhas. Mas também acreditamos serem elas confiáveis, por terem concordado (muito) bem com respostas analíticas e previsões de comportamento.

## 2 Aplicações

Realizamos simulações a partir de cenários hipotéticos, originados de fenômenos de colonização de certas regiões por espécie invasora. Na primeira subseção, as simulações têm

por objetivo demonstrar um comportamento previsto da população invasora, ao se incluir, na descrição de certos cenários nesses fenômenos, as chamadas fronteiras-interfaces. Já na segunda subseção, apresentamos um efetivo ajuste de parâmetro.

## 2.1 Simulação em dois ambientes adjacentes

Nas três simulações apresentadas a seguir, podemos pensar nos cenários envolvidos como que descrevendo fenômenos de colonização por espécie invasora à dois ambientes adjacentes e de características, eventualmente, distintas. A invasão se processa através do trecho de fronteira em contorno mais acentuado na figura. Modelamos a invasão pela fronteira utilizando-nos do fluxo ( $\sigma = -D\frac{\partial u}{\partial \eta}$ ), ao invés de uma condição de Neumann simplesmente, por entendermos ser tal procedimento o mais adequado na descrição de fenômenos naturais, e é exatamente o valor desse fluxo que fornecemos ao código na execução de todas as simulações. Sobre o restante dessa fronteira atribuímos a condição de Dirichlet nula.

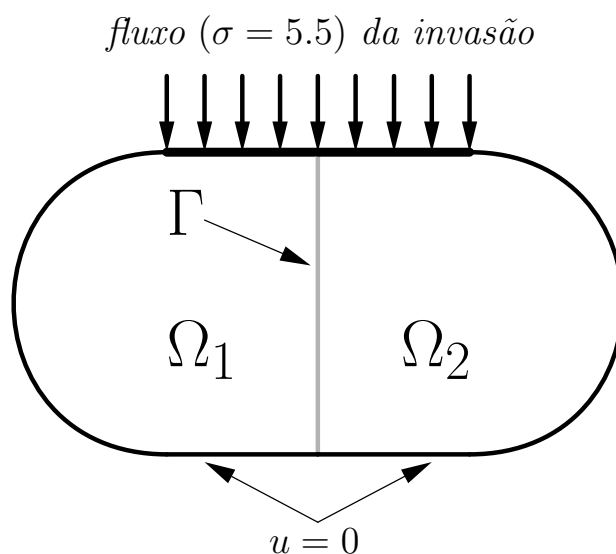


Figura 3.8: simulação de uma invasão em dois ambientes (meios) adjacentes

Mantivemos fixos os seguintes parâmetros:  $K_1 = K_2 = 50$ ,  $\lambda_1 = \lambda_2 = 0.7$ ,  $b_1 = b_2 = 0$ , e  $\sigma = 5.5$ . Com os seis primeiros parâmetros denotando, respectivamente, capacidades de suporte, taxas de crescimento intrínseco, e coeficientes de hostilidade nos ambientes adjacentes, na figura representados por  $\Omega_1$  e  $\Omega_2$ . A norma da diferença entre uma solução aproximada e

sua subsequente ( $\|\mathbf{u}^{k+1} - \mathbf{u}^k\|$ ), na seqüência de soluções geradas pelos métodos de resolução do sistema não-linear, foi imposta como sendo no máximo 0.0001. A solução usada na partida do Método de Máxima Descida foi  $(u_j^0 = 1)_{1 \leq j \leq N}$ , mas, obviamente, somente em índices de nós sobre os quais não se encontram definidas condições de Dirichlet.

Na primeira simulação os valores nos parâmetros da equação, e a malha gerada para a discretização do domínio, foram escolhidos de forma a fornecer uma solução simétrica entre os dois ambientes.

Os valores atribuídos aos parâmetros que restam livres são apresentados na Tabela 3.2, e a solução aproximada na Figura 3.9.

Tabela 3.2: meios adjacentes: parâmetros para cenário 1

parâmetro	domínio	
	$\Omega_1$	$\Omega_2$
$D$	4.0	4.0
$V$	(0,0)	(0,0)

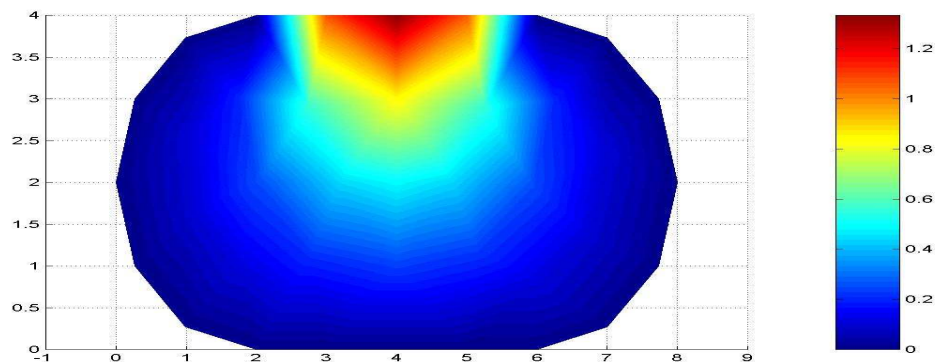


Figura 3.9: meios adjacentes: cenário 1

O fenômeno de invasão é evidente a partir de uma observação da figura, e o comportamento se apresenta mesmo como o esperado. A população invade a região colonizando-a,

mas se rarefaz à medida que se aproxima da fronteira com condição nula de Dirichlet, até desaparecer por completo sobre ela.

Na segunda simulação introduzimos o termo de migração, ou transporte, e diferentes coeficientes de dispersão sobre cada subdomínio.

Os valores atribuídos aos parâmetros que restam livres são apresentados na Tabela 3.3, e a solução aproximada na Figura 3.10.

Tabela 3.3: meios adjacentes: parâmetros para cenário 2

parâmetro	domínio	
	$\Omega_1$	$\Omega_2$
$D$	4.0	1.0
$V$	(0,-5.0)	(0,-5.0)

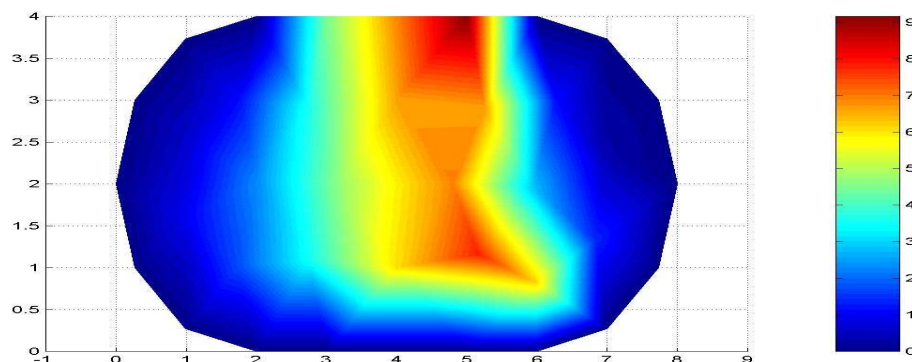


Figura 3.10: meios adjacentes: cenário 2

A solução se apresenta novamente conforme o intuitivamente previsto. Há um incremento na invasão em decorrência da introdução do termo de transporte, e, dado a diferença nas dispersões, e o fato de a fronteira-interface não impor restrições à passagem de indivíduos, pois sobre ela admitimos a compatibilidade de fluxo, um fluxo líquido origina-se através dessa

interface, observável da esquerda para a direita, tendo como consequência uma maior ocupação do ambiente à direita.

Na terceira e última simulação, a migração é colocada apenas no ambiente à direita, e voltamos com mesmo coeficiente de dispersão para os dois ambientes.

Os valores atribuídos aos parâmetros que restam livres são apresentados na Tabela 3.4, e a solução aproximada na Figura 3.11.

Tabela 3.4: meios adjacentes: parâmetros para cenário 3

parâmetro	domínio	
	$\Omega_1$	$\Omega_2$
$D$	4.0	4.0
$V$	(0,0)	(5.0,-5.0)

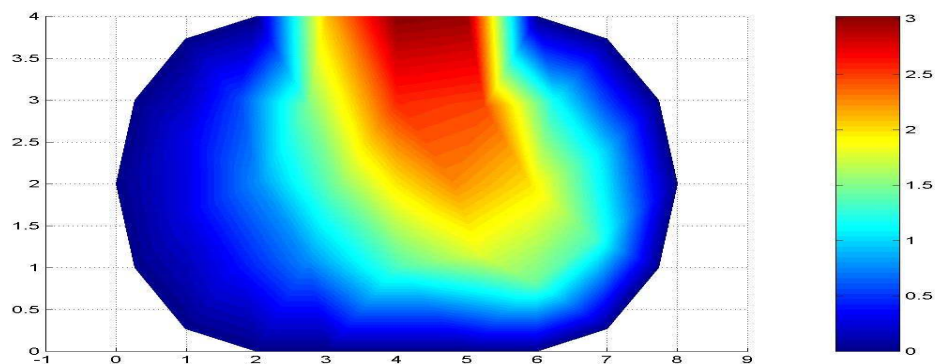


Figura 3.11: meios adjacentes: cenário 3

A solução aproximada é muito semelhante à obtida anteriormente, estando também de acordo com o previsto. A migração no ambiente à direita origina o mesmo fluxo através da interface entre os subdomínios, e temos assim uma maior povoamento desse ambiente.

## 2.2 Ajustando o coeficiente de dispersão

O código de programação desenvolvido, prestando-se a ajustar o parâmetro *coeficiente de dispersão* presente em uma equação elíptica não-linear<sup>2</sup> a partir de aproximações de sua solução, procura fornecer um valor ótimo à este coeficiente, independentemente de qual fenômeno, por esta equação modelado, esteja sob análise. Lembremos que, como já citado na parte introdutória deste trabalho, tais fenômenos podem abranger desde aqueles físico-químicos dos mais diversos, e com ocorrência a nível molecular, até os sociobiológicos de organismos superiores. Mas será apenas a partir destes últimos tipos de fenômenos citados, ou seja, os sociobiológicos observáveis, que construiremos um cenário tendo a solução no equilíbrio como objeto a ser aproximado.

Entenderemos como cenário, não apenas a região ou área geográfica, onde se dará o fenômeno, composta de suas características intrínsecas como dimensão, textura de vegetação eventualmente presente (aspecto que, como já anteriormente abordado, influencia diretamente na capacidade motil da espécie), e presença de barreiras à esta mobilidade, como ocorrência de lagos, lagoas etc, englobará também o termo as condições nas fronteiras desta área geográfica, pois que sujeitam a população em estudo. Acrescente-se ainda uma hostilidade do meio ambiente à população que se dispersa e migra sobre ele, como, por exemplo, existência de outra espécie predadora desta população.

O cenário, então, retrata um processo de invasão (já consumada) de uma população, que para efeito ilustrativo consideraremos como sendo de insetos herbívoros, em uma área agrícola de plantio a partir de uma área limítrofe, que também para efeito ilustrativo consideraremos como sendo de mata. Na verdade, é esta uma situação de ocorrência freqüente, dado que grandes extensões de plantio são comumente interrompidas, ou intercaladas, por áreas de preservação da vegetação original (nativa), e que estão sujeitas a povoamentos com tais características.

Prosseguindo na descrição de nosso cenário, consideramos que o fenômeno de invasão ocorre à vegetação que se instala na área de plantio, por a mesma possuir aspectos atrativos<sup>3</sup> para os insetos. Interrompendo talvez um período de atividade vital reduzida. Assim, observamos um fluxo de insetos através da fronteira na interface mata-plantação. Desta forma a área de mata pode ser entendida como se tratando apenas de um refúgio na ausência de área

---

<sup>2</sup>Com não linearidade aqui caracterizada por dinâmica verhulstiana.

<sup>3</sup>Químicos ou visuais, dependendo obviamente da especialidade da espécie invasora.

plantada, ou da mesma forma, refúgio antes da ocorrência do estado estacionário.

É relevante, para mostrarmos as qualidades e versatilidades inerentes ao código, que introduzamos uma certa complexidade referente às condições de contorno. Assim, incluímos no interior da área em estudo, um domínio no qual não haja possibilidade de sobrevivência para espécie a invasora. A existência de uma região de águas concentradas (ou corpos d'água), representadas por uma lagoa, por exemplo, é uma situação comum e que se presta bem a esta demonstração. Ainda assumindo que tais insetos não sobrevivam nesta porção de ambiente, isso pode ser modelado por um fluxo de passagem nulo na fronteira da interface entre a plantação e o que consideraremos doravante tratar-se de um lago.

Esta área, sobre a qual representaremos a invasão, terá ainda uma terceira fronteira, ou terceiro tipo de fronteira, além das originadas das interfaces plantação-mata e plantação-lago. Esta terceira fronteira, junto com as demais, completará todo o contorno da área de plantio, e ao longo da qual também devemos aplicar condições que, juntamente com as outras, nos permitirão a resolução do nosso modelo matemático. Imporemos uma condição de densidade populacional nula sobre este último trecho de fronteira. Isto feito, do ponto de vista de um processo de dispersão populacional, descreveremos perfeitamente uma fronteira distante da interface por onde se dá a invasão.

O levantamento e acesso às citadas informações que representam as condições necessárias à resolução do problema, quais sejam: fluxo de população na interface por onde se dá a invasão e na interface com a lagoa, valor de densidade populacional no resto de fronteira do domínio, campo vetorial originado por gradientes oriundos da presença de mecanismos atratores, ou oriundos de ventos predominantes, coeficiente representativo da hostilidade do meio (predação por outras espécies), capacidade de suporte do meio a sofrer invasão, relativo à espécie em estudo, e crescimento intrínseco da espécie, serão necessários ao processo de ajuste do coeficiente de dispersão.

Este coeficiente será ajustado a partir de determinação do desvio obtido na comparação direta entre a solução aproximada e alguns dados de densidade populacional, coletados em pontos distribuídos ao longo da plantação. Os pontos de coleta, para uma efetiva representação do comportamento invasor por toda a área de plantio, devem, naturalmente, estar bem distribuídos pela plantação, mas, certamente, em número limitado por considerações operacionais. Nas simulações que se seguem, consideraremos a existência de 16(dezesseis) destes pontos coletores, distribuídos como mostra a figura a seguir (Figura 3.12).



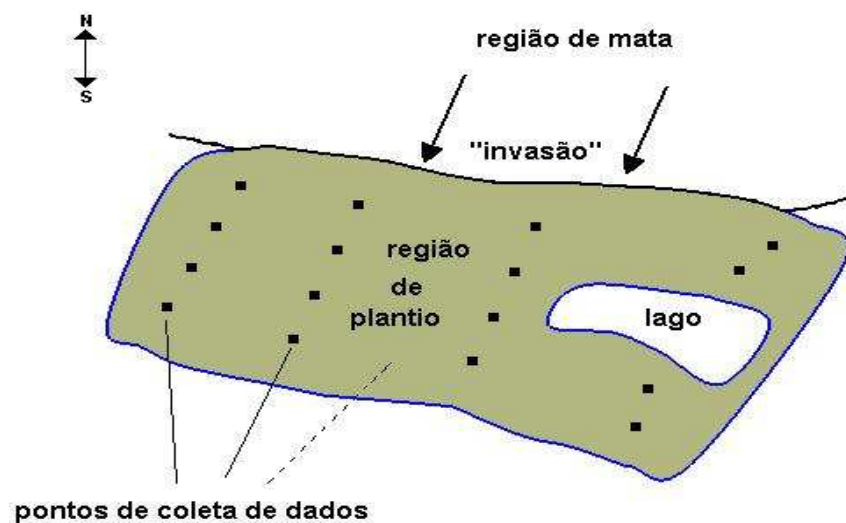


Figura 3.12: cenário para simulações

Um aspecto de fundamental importância diz respeito a procedimentos tomados quando da discretização do domínio – no presente caso a região de plantio. Deve-se sempre, na discretização, fazer concordar o total de pontos de informações obtidas em campo com nós (vértices dos triângulos) da malha de discretização, pois é sobre estes últimos que obtemos as soluções aproximadas. O exemplo de discretização abaixo apresentado, será usado nas simulações que se processarão na seqüência (Figura 3.13). Observemos, nesta discretização por triângulos, a concordância de posição dos vértices de alguns dos triângulos, representados por pontos escuros, com os pontos de coleta mostrados na figura anterior.

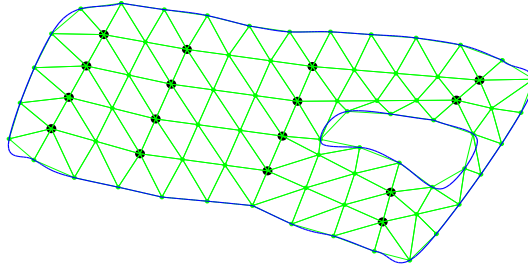


Figura 3.13: modelo de malha discretizando o domínio

Uma minimização do desvio entre dados coletados e seus correspondentes valores aproximados pelo código, a partir de variações no coeficiente de dispersão, talvez necessitasse da aplicação de técnicas matemáticas que recorressem a métodos em teoria de controle e de otimização. A partir de tais métodos, o valor deste desvio mínimo seria obtido pela resolução das equações abaixo

$$J(u, D) = \int_{\Omega} |u - u_d|^2 d\mathbf{x} + \int_{\Omega} |D|^2 d\mathbf{x}$$

$$(P) \begin{cases} \min J(u, D) \\ M(u, D) = 0 \end{cases}$$

$$M(u, D) = (u_1, u_2) \Leftrightarrow \begin{cases} -\Delta u + \frac{1}{D} \mathbf{V} \cdot \nabla u + \mu u + u^2 - f = \varphi_1 \\ u|_{\partial\Omega} = \varphi_2, \end{cases}$$

ou seja, um problema de controle ótimo em que queremos obter o melhor  $D$  tal que  $u$  fique próximo a uma função dada  $u_d$  (densidade nos pontos de amostragem).

Muito embora possa ser a forma acima descrita a mais indicada, ao observarmos a dependência linear do operador elíptico em relação ao coeficiente de dispersão, fica fácil a determinação de uma estratégia, ou metodologia operacional para seu ajuste. A proposta de tal metodologia, representativa de um “Shooting Method”, se fixará em duas ações: a primeira, resumindo-se a acompanharmos o desvio entre os dados de amostragem de densidade populacional ( $u_{am}$ ), em pontos especificados onde for de interesse a coleta de tais dados, e seus respectivos valores aproximados ( $u_{aprox}$ ), a partir da fórmula abaixo

$$\mathcal{DR} = \frac{|\mathbf{u}_{aprox} - \mathbf{u}_{am}|}{|\mathbf{u}_{am}|},$$

uma segunda ação, limitando-se ao acompanhamento dos valores aproximados obtidos sobre os referidos pontos, determinando o total dos que se encontram acima e dos que se encontram abaixo em relação aos dados coletados em campo. Esta última ação orienta a direção, pelo menos inicialmente, se crescente ou decrescente, de ajuste no coeficiente de dispersão.

A tabela que apresentamos a seguir, diz respeito a ajustes de  $D$  a partir de valores assumidos para os demais parâmetros da equação (elencados nesta mesma tabela).

Tabela 3.5: ajuste de coeficiente de dispersão em cenário de invasão

$D$	$\mathcal{DR}$	total de $u_{aprox}$ em relação aos $u_{am}$		
		abaixo	igual	acima
3.0	34.94%	3	0	13
3.2	33.32%	2	0	14
3.5	32.94%	2	0	14
4.0	35.13%	1	0	15

valores numéricos dos parâmetros na invasão

- taxa crescimento intrínseco  $\lambda = 0.07$
- capacidade suporte  $K = 50$
- direção de campo gradiente  $V = (-5, -15)$
- hostilidade do meio  $b = 0$
- fluxo na fronteira interface das regiões  $\sigma_n = -2.5$
- fluxo na fronteira do lago  $\sigma_n = 0$
- fronteira no infinito  $u = 0$

Como se pode observar, pela seqüência decrescente nos desvios percentuais obtidos, que houve, incontestavelmente, um ajuste para melhor em  $D$ , não obstante a ordem de grandeza destes desvios ter se mostrado relativamente alta.

Segue uma figura relativa à simulação descrita acima, com coeficiente de dispersão para o qual ocorre o menor desvio, ou seja,  $D = 3.5$ .

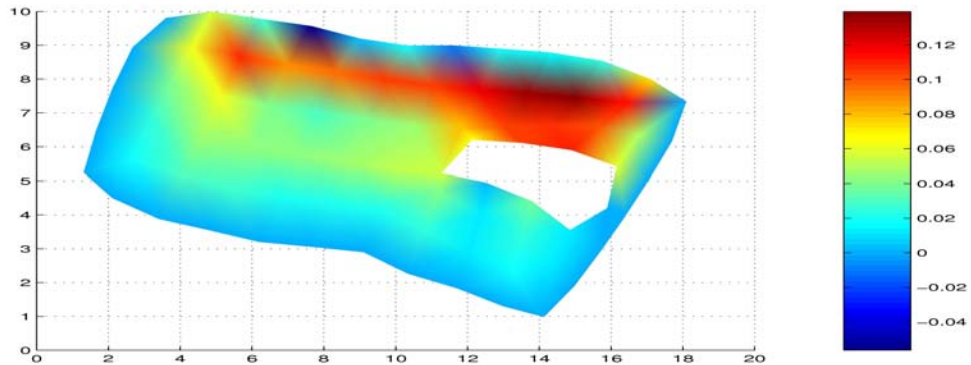


Figura 3.14: colonização no estado estacionário

Na figura acima intensidades de vermelho se referem à uma maior densidade populacional, e intensidades de cores próximas ao azul descrevem valores em torno de zero.

### 3 Análise dos resultados

Acreditamos que os erros que ocorrem com elevadas ordens de grandeza, não dizem respeito apenas a eventuais limitações do procedimento de ajuste proposto, reconhecidas existirem. Devemos sempre considerar os limites dos dados utilizados, ou mesmo do próprio modelo usado, pois, é da natureza da modelagem perguntarmos até que ponto o mesmo estaria adequado a um fenômeno de dispersão e migração que associado com um mecanismo de crescimento populacional.

Consideramos necessário um enfoque holístico na análise dos fenômenos naturais, levando-se em conta que são inúmeros os elementos a se considerarem na explicação de processos complexos como os são os da ecologia. Mas é oportuno lembrarmos do Princípio de Okham, que nos alerta sobre a intratabilidade de modelos por demais detalhados.

Além do que, e apesar das observações, a Figura 3.14 obtida está bem próxima do que nos parece afirmar a intuição<sup>4</sup> sobre o comportamento de um fenômeno de invasão com as características (parâmetros) assumidas, por apresentar-se com o aspecto mesmo de uma “onda” invasora. Embora sendo impróprio o uso do termo onda, o que nos remete a problemas com variação no tempo, representados matematicamente por equações diferenciais parabólicas e não elípticas, ele descreve bem o comportamento da distribuição populacional após uma invasão ter atingido o equilíbrio (como o esperado a priori). Qual seja, o de uma alta concentração de insetos nas proximidades da fronteira por onde se processa a invasão, mas ainda mais concentrado na parcela de área localizada entre a interface mata-plantação e a lagoa, pois é de se esperar que a presença da lagoa imponha uma dificuldade adicional ao processo de dispersão dessa população. Situação oposta ocorre na fronteira distante, ou seja, baixo povoamento por insetos.

Concordando, assim, o intuitivamente previsto com o posteriormente observado, é correto supormos a simulação como mais um teste de validação do código de programação, e

---

<sup>4</sup>Em filosofia, intuição é o “poder de adquirir conhecimento que não poderia ser obtido por inferência ou observação, por meio da razão ou da experiência”. Se são os axiomas objetos próprios da intuição, devemos aceitar a intuição como complementar à razão e não oposta à ela.

admitirmos como razoável a metodologia de ajuste do parâmetro coeficiente de dispersão.

## 4 Conclusões

Dos testes de validação e simulações, ficam evidentes as dificuldades de se lidar com equações do tipo elíptica quando desejamos obter soluções de problemas de contorno. Diferentemente do que ocorre com as equações parabólicas, aliás, de aplicação bem mais usual na descrição de uma dinâmica populacional, aqui nem todas as condições de contorno, implementadas computacionalmente, nos garantirão soluções positivas e portanto com significado e utilidade em tais fenômenos populacionais.

Também em determinadas situações, surgindo em verdade com razoável frequência, as aproximações obtidas podem se apresentar como não recuperadoras das soluções dos problemas. Uma confirmação analítica deste fato, também verificado experimentalmente em simulações efetuadas, mas não apresentadas, vem dos problemas de Neumann, que possuem todos, como propriedade característica, soluções determinadas a menos de uma constante positiva, ou seja, parâmetros surgem livres em suas soluções, impossibilitados que se acham de serem determinados, representando um grande prejuízo nas respostas aproximadas.

Embora sendo muitas as dificuldades, operacionais e matemáticas, de se trabalhar com uma equação elíptica com presença de dinâmica vital não linear, a mesma se prestou bem a objetivos de ajuste (determinação) de parâmetros presentes em fenômenos que governa, como a dinâmica populacional em estado estacionário que aqui se estudou.

# Considerações finais

Neste momento, cabe uma reflexão do que foi desenvolvido, do ponto de vista da contribuição para o conhecimento de um modo geral, nos capítulos precedentes ao longo desta dissertação.

Não se trata, e nem foi o objetivo central, ou mesmo não se teve a presunção, de ter criado aqui um trabalho original, dado que as linhas escritas que se seguiram representam meramente, e acima de tudo, a intenção deste autor em demonstrar sua habilidade em dar os primeiros passos no campo da perscrutação científica.

Mesmo evidentemente limitando-se a uma revisão bibliográfica de autores consagrados, nacionais ou não, e por vezes apresentando técnicas originais como, por exemplo, a de compatibilização entre subdomínios usada na formulação variacional, e apresentada por Diniz (Diniz, 2003), que simplificou e possibilitou a demonstração de existência de solução do problema fraco, há de se identificar certa originalidade, ou pelo menos dificuldade associada, no capítulo que trata desta demonstração de existência. Em especial, a descrição de aspectos relevantes nesta mesma demonstração poderá talvez compor futuros trabalhos de outros autores, ou mesmo deste.

Temos consciência da limitação do trabalho aqui desenvolvido. Neste sentido, novos desafios apontam tanto para a melhoria da interatividade do programa apresentado quanto dos algoritmos implementados. No primeiro destes aspectos, caberia desenvolvermos solucionadores (“solvers”) externos que podendo ser interligados ao aplicativo **Gmsh**, permitiria facilmente utilizar-se destes processos computacionais externos e, após coleta de dados, simular resultados dentro dos módulos de pós-processamento e interface gráfica do próprio **Gmsh**, mas para tanto deveríamos recorrer a solucionadores desenvolvidos em linguagem C, ou C++, ou mesmo híbrida C/C++. Quem sabe até possibilitaríamos seu uso, num estilo “caixa preta”,

por profissionais da biologia, menos hábeis ou preocupados com procedimentos computacionais. Com certeza uma portentosa tarefa para um iniciante em linguagens de programação, que exigiria conhecimentos de programação orientada a objetos, abandonada no decorrer deste trabalho e deixada como opção para implementação futura. No último aspecto, elevar-se a ordem dos elementos finitos utilizados, ou mesmo recorrer ao uso de funções base fornecidas pelos polinômios de Hermite, o que conferiria maior suavidade a elas, sempre com o objetivo de construir-se mais precisos e eficientes métodos de aproximação.

Talvez o uso da modelagem *fuzzy* em parâmetros, como para o termo advectivo que, podendo estar associado a ventos ocorrendo no ambiente ecológico, traz embutido algum grau de incerteza, possa também representar uma possibilidade para trabalhos futuros, sendo provavelmente um outro objetivo deste mesmo autor, em uma nova e adicional linha de investigação.



# Bibliografia

- Adams, R. A. (1975). *Sobolev Spaces*. Academic Press.
- Assis, R. A. (2003). Modelos em estratégias de forrageamento de formigas. Dissertação de Mestrado, IMECC – UNICAMP, Campinas/SP.
- Bassanezi, R. C. e Ferreira Jr., W. C. (1988). *Equações Diferenciais com Aplicações*. Ed. Harbra, S. Paulo.
- Boldrini, J. L. (2004). Comunicação pessoal.
- Brézis, H. (1984). *Análisis Funcional: Teoría y Aplicaciones*. Versión Española. Alianza Editorial, S. A., Madrid.
- Burden, L. B. e Faires, J. D. (1993). *Numerical Analysis*.
- Ciarlet, P. G. (1987). *The finite element for elliptic problems*. North-Holland, Amsterdam.
- de Santana, L. A. R. (2003). *Algumas Considerações Teóricas sobre a Persistência de Espécies em Meios Heterogêneos*. Tese de Doutorado, Imecc-Unicamp, Campinas/SP.
- Dennis JR, J. E. e Schnabel, R. D. (1983). *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Printice-Hall, Englewood Cliffs.
- Diniz, G. L. (2003). *Dispersão de poluentes num sistema ar-água: modelagem aproximação e aplicações*. Tese de Doutorado, IMECC – UNICAMP, Campinas/SP.
- Dukas, R. (1998). *Cognitive Ecology: The Evolutionary Ecology of Information Processing and Decision Making*. Univ. Chicago Press.

- Edelstein-Keshet, L. (1988). *Mathematical Models in Biology*. Random-House, N. York.
- Einstein, A. (1956). *Investigations on the Theory of the Brownian Movement – artigo – Elementary Theory of the Brownian Motion*. Dover.
- Euler, L. (1970). A general investigation into the mortality and multiplication of the human species,. *Acad. Royale des Sci. et Belles Lettres, Bruxellas , trad. Th. Pop Biol.*, 1:307–314.
- Evans, L. C. (1998). *Partial Differential Equations*, volume 19. American Mathematical Society, Providence, Rhode Island.
- Fagan, W. F. e Cantrell, R. S. (1988). How habitat edges change species interations. *American Naturalist*, 153(2):165–182.
- Ferreira Jr, W. C. (2001). Comportamento individual e dispersão social. Em *Conferência proferida no Congresso Latino Americano de Biomatemática*. Unicamp.
- Gidas, B., Ni, W., e Nirenberg, L. (1981). Symetry and related properties via the maximum principle. *Comm. Math. Phys.*, 68:209–243.
- Gilbarg, D. e Trudinger, N. S. (1983). *Elliptic Partial Differential Equations of Second Order*. Springer-Verlag.
- Kareiva, P. M. (1983). Local moviment in herbivorous insects: applying a passive diffusion model to mark-recapture field experiments. *Oecologia*, 57:322–327.
- Kareiva, P. M. e Shigesada, N. (1983). Analyzing insect movement as a correlated random walk. *Oecologia*, 56:234–238.
- Keller, E. F. e Segel, L. A. (1971). Models for chemotaxis. *J. Math. Biol.*, 30:225–234.
- Lacaz, T. M. V. S. (1999). *Análise de problemas populacionais intraespecíficos com difusão densidade-dependente*. Tese de Doutorado, IMECC-Unicamp, Campinas/SP.
- Lions, J. L. (1969). *Quelques Méthodes de Résolution des Problèmes aux Limits non Linéaires*. Dunod.
- Lions, P. L. (1981). Two geometrical properties of solutions of semilinear problems. *Appl. Anal.*, 12:267–272.

- Lorenz, K. (1974). *On Agression*. Oxford Univ. Press.
- Ludwig, D., Aronson, D. G., e Weinberger, H. F. (1979). Spatial patterning of the spruce budworm. *J. Math. Biology.*, 8:217–258.
- Murray, J. D. (1977). *Lectures on Nonlinear-differential-equations Models in Biology*. Claredon Press, Oxford.
- Murray, J. D. (1989). *Mathematical Biology*. Springer, N. York.
- Oden, J. T., Becker, E. B., e Carey, G. F. (1981). *Finite Elements: an Introduction*, volume 1. Prentice Hall.
- Pennisi, E. (2000). In nature, animals that stop and start win the race. *Proceedings of a Microprogram held, August 26-September 12/1986*, 288:83–85.
- Pregolato, S. A. (2002). *Mal-das-cadeiras em Capivaras: Estudo, Modelagem e Simulação de um caso*. Tese de Doutorado, FEEC-Unicamp, Campinas/SP.
- Rodrigues, L. A. D. (1998). *O Acaso e a Decisão: Modelos Matemáticos para a Dispersão Social*. Tese de Doutorado, Imecc-Unicamp, Campinas/SP.
- Segel, L. (1984a). *Modeling Dynamic Phenomena in Molecular Biology*. Cambridge Univ. Press.
- Segel, L. A. (1984b). Taxes in cellular ecology. *Lect. Notes Biomath*, 54:407–424.
- Segel, L. A. e Perelson, A. S. (1988). *Computation in Shape Space: A New Approach to Immune Network Theory*. Addison-Wesley. Theoretical Immunology II.
- Skellam, J. G. (1951). Random dispersal in theoretical populations. *Biometrika*, 38:196–218.
- Sossae, R. C. (1995). Dinâmica populacional densidade-dependente em processos de dispersão e migração. Dissertação de Mestrado, IMECC – UNICAMP, Campinas/SP.
- Tinbergen, N. (1974). *The Study of Instinct*. Oxford Univ. Press.
- Wolfe, J. A., Alvarez, G. A., e Horowitz, T. S. (2000). Attention is fast but volition is slow: A random scan is a quicker way to find items in a display than a systematic search. *Nature*, página 691.



# Apêndice A

## Conceitos, definições e teoremas

Neste apêndice, são apresentados alguns conceitos em Análise Funcional, que por serem úteis ao entendimento do texto, principalmente na parte que trata da demonstração de existência da formulação fraca, mas não muito freqüente na literatura, achamos por bem que figurassem no presente trabalho. Compõem-se de definições, lemas e teoremas, com demonstrações, as quais, quando necessário, fazemos referência aos livros que as contêm.

### 1 Conceitos em análise funcional

#### 1.1 Definições

**Definição 1.1.1 (Espaço Reflexivo).** Seja  $E$  um espaço de Banach, e seja  $J$  a injeção canônica de  $E$  em  $E''$  (bidual de  $E$ : espaço dos funcionais lineares limitados de  $E'$ ). Diz-se que  $E$  é *reflexivo* se  $J(E) = E''$ , ou seja, quando  $E$  é reflexivo se identificam implicitamente  $E$  e  $E''$  com a ajuda do isomorfismo  $J$ .

**Definição 1.1.2 (Base Hilbertiana).** Chama-se base hilbertiana a toda sucessão  $(e_n)$  de

elementos de  $H$  tais que:

- (i)  $|e_n| = 1 \forall n, (e_m, e_n) = 0 \forall m, n, \text{ com } m \neq n,$
- (ii) o espaço vetorial gerado pelos  $(e_n)$  é denso em  $H$ .

Além do que, todo  $u \in H$  se escreve na forma

$$u = \sum_{n=1}^{\infty} (u, e_n) e_n$$

**Definição 1.1.3 (Operador Compacto).** Tome  $X$  e  $Y$  espaços de Banach reais. Um operador linear limitado

$$K : X \longrightarrow Y$$

é dito compacto deste que para cada seqüência limitada  $\{u_k\}_{k=1}^{\infty} \subset X$ , a seqüência  $\{Ku_k\}_{k=1}^{\infty}$  é precompacta em  $Y$ ; isto é, existe uma subseqüência  $\{u_{k_j}\}_{j=1}^{\infty}$  tal que  $\{Ku_{k_j}\}_{j=1}^{\infty}$  converge em  $Y$ .

**OBS:** Tomando um  $H$  denotando um espaço de Hilbert real. É fácil ver que se um operador linear  $K : H \longrightarrow H$  é compacto e  $u_k \rightarrow u$ , então  $Ku_k \rightarrow Ku$ .

**Definição 1.1.4 (Imersão Compacta).** Tome  $X$  e  $Y$  espaços de Banach,  $X \subset Y$ . Diz-se que  $X$  está imerso compactamente em  $Y$ , e escreve-se

$$X \hookrightarrow\hookrightarrow Y$$

desde que

(i)  $\|x\|_Y \leq C\|x\|_X$  ( $x \in X$ ) para alguma constante  $C$  e

(ii) cada seqüência limitada em  $X$  é precompacta em  $Y$

**Definição 1.1.5 (Topologia Fraca).** A topologia fraca de um espaço normado  $X$  representa uma topologia enfraquecida sobre  $X$  onde ainda restam contínuas cada  $x' \in X'$ .

**Definição 1.1.6 (Gateaux-derivada).** Uma aplicação  $F : D \subset \mathbb{R}^n \rightarrow \mathbb{R}^m$  é *Gateaux-* (ou *G-*) *diferenciável* num ponto interior  $x$  de  $D$  se existir um operador linear  $A \in L(\mathbb{R}^n, \mathbb{R}^m)$  tal que, para cada  $h \in \mathbb{R}^n$ ,

$$\lim_{t \rightarrow 0} \left( \frac{1}{t} \right) \|F(x + th) - Fx - tAh\| = 0$$

## 1.2 Teoremas

**Teorema 1.2.1 (Convergência Local do Método de Newton).** Tome  $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$  continuamente diferenciável num conjunto aberto e convexo  $D \subset \mathbb{R}^n$ . Assuma que existe  $\mathbf{x}^* \in \mathbb{R}^n$ , e  $r, \beta > 0$ , tal que  $N(\mathbf{x}^*, r) \subset D$ ,  $F(\mathbf{x}^*) = \mathbf{0}$ ,  $J(\mathbf{x}^*)^{-1}$  existe com  $\|J(\mathbf{x}^*)^{-1}\| \leq \beta$ , e  $J \in Lip_\gamma(N(\mathbf{x}^*, r))$ . Então existe  $\epsilon > 0$  tal que para todo  $\mathbf{x}_0 \in N(\mathbf{x}^*, \epsilon)$  a seqüência  $\{\mathbf{x}_k\}$  gerada por

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - J(\mathbf{x}^{(k)})^{-1}F(\mathbf{x}^{(k)}), \quad k = 0, 1, \dots$$

está bem definida, converge para  $\mathbf{x}^*$ , e obedece

$$\|\mathbf{x}^{(k+1)} - \mathbf{x}^*\| \leq \beta\gamma\|\mathbf{x}^{(k)} - \mathbf{x}^*\|^2, \quad k = 0, 1, \dots$$

**prova:** ver Dennis JR e Schnabel (1983) pg-90

**Teorema 1.2.2 (Ponto-Fixo de Brower).** *Toda aplicação contínua da bola unitária fechada de  $\mathbb{R}^n$  em si mesma tem ponto-fixa.*

**prova:** ver Evans (1998) pg-441

**Teorema 1.2.3 (Banach-Alaoglu-Bourbaki).** *Seja  $E$  um espaço de Banach. Então  $E$  é reflexivo se, e só se,  $B_E = \{x \in E; \|x\| \leq 1\}$  é compacto na topologia  $\sigma(E, E')$ .*

**prova:** ver Brézis (1984) pg-42

**Teorema 1.2.4 (Traço).** *Tome  $\Omega$  um subconjunto limitado do  $\mathbb{R}^N$  com fronteira  $\partial\Omega$  Lipschitz. Então existe um único operador linear limitado*

$$\begin{aligned}\gamma_0 : W^{1,p}(\Omega) &\longrightarrow H^{1/2}(\partial\Omega) \\ u &\longmapsto \gamma_0(u)\end{aligned}$$

*tal que:*

(i)  $\gamma_0(u) = u|_{\partial\Omega}$  se  $u \in W^{1,p}(\Omega) \cap C(\bar{\Omega})$  e

(ii)  $\|\gamma_0(u)\|_{H^{1/2}(\partial\Omega)} \leq C\|u\|_{W^{1,p}(\Omega)}$ , para todo  $u \in W^{1,p}(\Omega)$ , com a constante  $C$  dependente apenas de  $p$  e  $\Omega$ .

*Também vale a recíproca, isto é, existe um único operador linear limitado*

$$\begin{aligned}l_0 : H^{1/2}(\partial\Omega) &\longrightarrow W^{1,p}(\Omega) \\ g &\longmapsto l_0(g) = h \in H^1(\Omega)\end{aligned}$$



tal que:

(i)  $\|h\|_{H^1(\Omega)} \leq C\|g\|_{H^{1/2}(\partial\Omega)}$ , para todo  $g \in H^{1/2}(\partial\Omega)$ , com a constante  $C$  dependente apenas de  $\Omega$ .

**prova:** ver Evans (1998) pg-258

**Teorema 1.2.5 (Teorema de Imersão de Sobolev).** Tome  $\Omega$  um domínio no  $\mathbb{R}^n$  e  $\Omega^k$  um  $k$ -dimensional domínio obtido por intersecção de  $\Omega$  com um  $k$ -dimensional plano em  $\mathbb{R}^n$ ,  $1 \leq k \leq n$  (portanto  $\Omega^n \equiv \Omega$ ). Tome  $p$  satisfazendo  $1 \leq p < \infty$ . Se  $\Omega$  tem a propriedade do Cone, então existe a imersão

$$W^{m,p}(\Omega) \hookrightarrow L^q(\Omega), \quad p \leq q < \infty$$

Todas as conclusões acima são válidas para domínios arbitrários desde que os  $W$ -espaços suportando imersões sejam substituídos pelos correspondentes  $W_0$ -espaços. Então, em particular  $H_1^0(\Omega) \hookrightarrow L^q(\Omega)$ ,  $2 \leq q < \infty$ . Diz-se de outro modo que existe uma constante  $C$  (dependente somente de  $\Omega$ ) tal que  $\|u\|_L^q \leq C\|u\|_{H_1^0}$ ,  $\forall u \in H_1^0(\Omega)$ .

**prova:** ver Adams (1975) pg-98

### 1.3 Lemas

**Lema 1.3.1 (Lions).** Seja  $\theta \subset \mathbb{R}^n$  limitado, com  $\|g_n\|_{L^p(\theta)} \leq C$ ,  $1 < p < +\infty$ , e  $g_n \rightarrow g$  q.t.p. em  $\theta$ , então existe  $\{g_n\}_{n=1}^{+\infty}$  tal que  $g_n \rightharpoonup g$  (fracamente) em  $L^p(\theta)$ .

**prova:** ver Lions (1969)

## 1.4 Desigualdades

**Desigualdade 1.4.1 (Young).** Tome  $1 < p, q < \infty$ ,  $\frac{1}{p} + \frac{1}{q} = 1$ . Então

$$ab \leq \frac{a^p}{p} + \frac{b^q}{q} \quad (a, b > 0)$$

**prova:** ver Evans (1998) pg-622

**Desigualdade 1.4.2 (Young com  $\varepsilon$ ).**

$$ab \leq \varepsilon a^p + C(\varepsilon)b^q \quad (a, b > 0, \varepsilon > 0) \quad \text{para} \quad C(\varepsilon) = (\varepsilon p)^{-q/p} q^{-1}$$

**prova:** ver Evans (1998) pg-622

**Desigualdade 1.4.3 (Poincaré).** Para  $u \in W_0^{1,p}(\Omega)$ ,  $1 \leq p < \infty$ , existe  $C = C(p, \Omega)$  tal que

$$\|u\|_{L^p(\Omega)} \leq C \|\nabla u\|_{L^p(\Omega)}$$

**prova:** ver Gilbarg e Trudinger (1983) pg-164

# Apêndice B

## Formato de arquivos para o programa

Este apêndice descreve o formato do arquivo com dados de entrada para o programa, que não podem ser modificados pelo usuário. No presente trabalho, arquivos neste formato foram obtidos com auxílio do aplicativo **Gmsh** disponível em rede, pelo menos no que se refere à obtenção do vetor de conectividade<sup>1</sup> e coordenadas dos nós da malha.

### 1 Arquivo de dados de entrada

Este arquivo, com extensão ‘.txt’, deve ser construído dividido em várias seções como segue:

*taxa-cresc*

*cap-suporte*

*número-nós*

*x-coord y-coord*

*. . .*

*número-elementos*

*tipo-elem ind-subdomínio ind-regra-integ num-nó lista-nó*

---

<sup>1</sup>vetor que fornece a lista dos nós de forma seqüencial (conectada), representado na descrição do *arquivo de dados de entrada* na alocação *lista-nó*

. . .

*número-subdomínios*

*coef-dif V<sub>x</sub>-coord V<sub>y</sub>-coord hostile fonte*

. . .

*número-populações-pontuais*

*ind-nó valor-pop*

. . .

*número-condições-contorno-pontuais*

*num-nó tipo-cond dirichlet neumann*

. . .

*número-regras-integração*

*ordem coord-ptos pesos*

. . .

onde:

*taxa-cresc*

é o valor de ponto-flutuante para a taxa de crescimento na fonte verhulstiana;

*cap-suporte*

é o valor de ponto-flutuante para a capacidade de suporte da população na fonte verhulstiana;

*número-nós*

é o número de nós presentes na malha de discretização do domínio;

*x-coord y-coord*

são os valores de ponto-flutuante para as coordenadas de cada nó;

*número-elementos*

é o número de elementos (segmentos, triângulos e quadrados) na malha;

*tipo-elem*

define o tipo de geometria de cada elemento

0 se segmento

1 se quadrado

2 se triângulo;

*ind-subdomínio*

sendo a contagem iniciada do zero, é o número do subdomínio ao qual o elemento pertence (para efeitos computacionais, entenda-se como subdomínio também os diferentes segmentos de fronteira sobre os quais definimos as condições de contorno, e linhas internas sobre as quais aplicamos densidades lineares de fluxo);

*ind-regra-integ*

atribui a cada elemento a ordem da regra de integração a partir de uma identificação deste elemento com o número da linha menos um na *matriz regra de integração*;

*num-nó*

é o número de nós presentes em cada elemento;

*lista-nó*

é a lista de nós presentes em cada elemento, definindo a conectividade entre os elementos;

*número-subdomínios*

é o número de subdomínios que dividem o domínio representativo do espaço geográfico mais os segmentos de fronteira e, eventualmente, mais os segmentos onde aplicamos fontes lineares de densidades populacionais;

*coef-dif*

é o valor de ponto-flutuante para coeficiente de difusão;

*V<sub>x</sub>-coord*

é o valor de ponto-flutuante para coordenada  $x$  do campo de velocidades;

*V<sub>y</sub>-coord*

é o valor de ponto-flutuante para coordenada  $y$  do campo de velocidades;

*hostil*

é o valor de ponto-flutuante para a hostilidade do meio, representando também o coeficiente associado ao valor de densidade populacional na condição mista, sobre a fronteira. Deverá conter o valor de *taxa-cresc* quando *tipo-elem* for zero;

*fonte*

é o valor de ponto-flutuante para a fonte externa, representando também fontes lineares ou o coeficiente associado à derivada direcional na condição mista sobre a fronteira;

*número-populações-pontuais*

é o número de fontes pontuais de população;

*ind-nó*

é o índice do nó sobre o qual atribuímos uma fonte pontual;

*valor-pop*

é o valor de ponto-flutuante para a fonte pontual;

*número-condições-contorno-pontuais*

é a soma do número de condições de contorno de Dirichlet, Neumann e mista que se deseja atribuir pontualmente;

*num-nó*

é o número do nó ao qual se atribui as condições pontuais;

*tipo-cond*

é o tipo de condição de contorno

0 se Dirichlet

1 se mista

2 se Neumann;

*dirichlet*

é o valor de ponto-flutuante para a condição de Dirichlet, ou para o coeficiente associado ao valor de densidade populacional na condição mista sobre a fronteira;

*neumann*

é o valor de ponto-flutuante para a condição de Neumann, ou para o coeficiente associado à derivada direcional na condição mista sobre a fronteira;

*número-regras-integração*

é o número de regras de Gauss-Legendre utilizadas para aproximações das integrais, define o número de linhas da matriz que contém todos os nós e respectivos pesos de cada fórmula;

*ordem*

é a ordem da fórmula de quadratura gaussiana;

*coord-ptos*

são as coordenadas dos zeros dos polinômios de Legendre;

*pesos*

são os pesos da fórmula.

# Apêndice C

## Códigos fontes

Neste apêndice, é apresentado o código de programa utilizado na implementação do método numérico escolhido para as aproximações da solução do problema. Tentamos, dentro do possível, apresentá-lo de forma que possibilitasse a compreensão, pelo menos parcialmente, do encadeamento das chamadas de rotinas e respectivas subrotinas.

### 1 Códigos para o MATLAB<sup>®</sup>

O código a seguir, foi implementados em ambiente Windows - no Matlab<sup>®</sup> versão 5.3.0.10183 Release 11.1.

#### 1.1 Código principal

```
1 %-----  
2 %      Resolução de equação elíptica não linear por Elementos Finitos  
3 %      para casos Uni e Bi-Dimensional, com implementação de método de  
4 %      máxima descida na determinação de chute inicial com aproximação  
5 %      linear, e método de Newton na determinação de solução aproximada  
6 %      com precisão quadrática.  
7 %-----  
8 %  
9 clear all  
10  
11 format
```



```

12 Filename = input('Nome do arquivo.txt com dados de entrada: ','s');
13 global num_knot                % Variáveis Globais
14 global coord_knot
15 global type_el
16 global mat_el
17 global index_rule_I_el
18 global num_knot_el
19 global disp_knot
20 global coef_mat
21 global num_knot_cdc
22 global knot_cdc
23 global type_cdc
24 global M
25 global N
26 global R
27 global JF
28 global Sigma
29 global matrix
30 global lambda
31 global num_el
32 global K
33
34 maxiter = 1.e5;
35 c = input('valor do chute = ');
36 eps = input('erro na aproximação = ');
37 figura = input('plotar sobre figura número = ');
38 [fid,message] = fopen(Filename, 'r'); % Abrindo arquivo.txt c/ dados p/ leitura
39 lambda = fscanf(fid,'%g',1);        % velocidade de crescimento intrínscica
40 K = fscanf(fid,'%g',1);              % capacidade ambiental de sustentação da
41
42 num_knot = fscanf(fid,'%d',1);       % número de nós da malha
43 coord_knot = fscanf(fid,'%g',[2 num_knot]); % coordenadas dos nós da malha
44 coord_knot = coord_knot';
45 num_el = fscanf(fid,'%d',1);        % número de elementos
46 pos = 0;
47 for line = 1 : num_el
48     row = 4;
49     vec = fscanf(fid,'%d',row);
50     type_el(line) = vec(1);          % elemento (segmento, triângulo, quadrado)
51     mat_el(line) = vec(2);           % índice do ambiente
52     index_rule_I_el(line) = vec(3);  % índice da regra de integração
53     num_knot_el(line) = vec(4);      % N° de nós,
54     row = vec(4);
55     vec = fscanf(fid,'%d',row);
56     pos = pos + row;
57 end

```

```

58 disp_knot = disp_knot;
59 num_mat = fscanf(fid,'%d',1);          % número de ambientes e CDC
60 coef_mat = fscanf(fid,'%f',[5 num_mat]); % coeficientes da equação
61
62 coef_mat = coef_mat';
63 num_nodal_load = fscanf(fid,'%d',1); % número de populações nodais
64 if num_nodal_load ~= 0
65     for line = 1 : num_nodal_load
66         vec = fscanf(fid,'%d %g',[2 1]);
67         disp_load(line) = vec(1);      % índice do respectivo nó da malha
68         val_load(line) = vec(2);      % valor da densidade de população nodal
69     end
70 end
71 num_knot_cdc = fscanf(fid,'%d',1);
72 if num_knot_cdc ~= 0
73     for line = 1 : num_knot_cdc;
74         row = 4;
75         vec = fscanf(fid,'%d %d %g %g',[row 1]);
76         knot_cdc(line) = vec(1);
77         type_cdc (line) = vec(2);      % tipo de CDC (Dirichlet, etc)
78         cdc(1 : 2,line) = vec(3 : 4); % condições de contorno pontuais
79     end
80     knot_cdc = knot_cdc;
81     cdc = cdc';
82 end
83 num_rule_I = fscanf(fid,'%d',1);      % número de regras de integração
84 matrix = zeros(num_rule_I,15);
85 for line = 1 : num_rule_I
86     scalar = fscanf(fid,'%d',1);      % ordem da fórmula de quadratura (Gaussiana)
87     matrix(line,1) = scalar;
88     scalar = 2 * scalar + scalar;
89     vec = fscanf(fid,'%f',scalar);
90     matrix(line,2 : (scalar + 1)) = vec'; % nós (zeros de Legendre) e pesos
91                                           % da fórmula
92 end
93 fclose(fid);                          % Fechamento do arquivo
94 M = sparse(num_knot,num_knot);
95 Sigma = sparse(num_knot,1);
96 posi = 0;                              % Construindo M (matriz linear no sistema) e
97                                           % Sigma a partir de contribuições locais de
98                                           % matrizes me e sigma_e. Nesta etapa o código
99                                           % já acrescenta todos os tipos de CDC
100 for e = 1 : num_el
101     [me,sigma_e,disp_knot_el,posi] =...
102         local_matrix_M(type_el(e),mat_el(e),num_knot_el(e),index_rule_I_el(e),posi);
103     [M,Sigma] = global_matrix_M(me,sigma_e,disp_knot_el,num_knot_el(e));

```

```

104 end
105 if num_nodal_load ~= 0           % Acrescentando populações nodais ao vetor Sigma
106     Sigma = load_vec(num_nodal_load,disp_load,val_load);
107 end
108 if num_knot_cdc ~= 0           % Acrescentando CDC pontuais ao vetor Sigma
109     [M,Sigma] = cdc_point(num_knot_cdc,knot_cdc,type_cdc,cdc);
110 end
111 epsilon = 1;                   % o "Steepest Descent"
112 uh0(1,1 : num_knot) = c;
113 iterammd = 0;
114 index = 0;
115 if num_knot_cdc ~= 0
116     for i = 1 : num_knot_cdc % Inserindo cdc de Dirichlet no vetor "chute" inicial
117         uh0(knot_cdc(i) + 1) = cdc(i,1);
118     end
119 end
120 fall = 0;
121 while (epsilon > eps) & (fall == 0)
122     iterammd = iterammd + 1;
123     N = sparse(num_knot,num_knot);
124     N = matrix_N(uh0);          % (matriz não-linear no sistema)
125     if num_knot_cdc ~= 0       % Adaptando N às CDC pontuais
126         N = cdc_point_N(num_knot_cdc,knot_cdc,type_cdc);
127     end
128     JF = M + 2 * (lambda / K) * N; % matriz Jacobiana do sistema
129     R = M + (lambda / K) * N;
130     F = R * uh0' - Sigma;
131     g1 = sum(F .* F);
132     vecg(iterammd) = g1;
133     for j = 1 : num_knot
134         grad_g(j) = 2 * sum(F .* JF(:,j));
135     end
136     Grad(iterammd,1:2) = [grad_g(8) grad_g(8)];
137     norm_grad_g = norm(grad_g,2);
138     if norm_grad_g == 0
139         fall = 1;
140         g = g1;
141         uh = uh0;
142     else
143         grad_g = grad_g / norm_grad_g;
144         alpha3 = 1;
145         z3 = uh0 - alpha3 * grad_g;
146         N = sparse(num_knot,num_knot);
147         N = matrix_N(z3);
148         if num_knot_cdc ~= 0
149             N = cdc_point_N(num_knot_cdc,knot_cdc,type_cdc);

```

```

150     end
151     R = M + (lambda / K) * N;
152     F = R * z3' - Sigma;
153     g3 = sum(F .* F);
154     while (g3 >= g1) & (fall == 0)
155         if alpha3 < eps / (2^128)
156             fall = 1;
157             g = g1;
158             uh = uh0;
159         else
160             alpha3 = alpha3 / 2;
161             z3 = uh0 - alpha3 * grad_g;
162             N = sparse(num_knot,num_knot);
163             N = matrix_N(z3);
164             if num_knot_cdc ~= 0
165                 N = cdc_point_N(num_knot_cdc,knot_cdc,type_cdc);
166             end
167             R = M + (lambda / K) * N;
168             F = R * z3' - Sigma;
169             g3 = sum(F .* F);
170         end
171     end
172     if fall == 0
173         alpha2 = alpha3 / 2;
174         z2 = uh0 - alpha2 * grad_g;
175         N = sparse(num_knot,num_knot);
176         N = matrix_N(z2);
177         if num_knot_cdc ~= 0
178             N = cdc_point_N(num_knot_cdc,knot_cdc,type_cdc);
179         end
180         R = M + (lambda / K) * N;
181         F = R * z2' - Sigma;
182         g2 = sum(F .* F);
183         h1 = (g2 - g1) / alpha2;
184         h2 = (g3 - g2) / (alpha3 - alpha2);
185         h3 = (h2 - h1) / alpha3;
186         alpha0 = 0.5 * (alpha2 - (h1 / h3));
187         z0 = uh0 - alpha0 * grad_g;
188         N = sparse(num_knot,num_knot);
189         N = matrix_N(z0);
190         if num_knot_cdc ~= 0
191             N = cdc_point_N(num_knot_cdc,knot_cdc,type_cdc);
192         end
193         R = M + (lambda / K) * N;
194         F = R * z0' - Sigma;
195         g0 = sum(F .* F);

```

```

196         minimum = min([g0 g3]);
197         if minimum == g0
198             uh = z0;
199             g = g0;
200         elseif minimum == g3
201             uh = z3;
202             g = g3;
203         end
204         UH(iterammd,1:2) = [uh(8) uh(8)];
205     end
206 end
207 epsilon = norm(g - g1,2);
208 uh0 = uh;
209 end
210 if fall == 0 % aproximação por Método de Newton
211     [uh0,iteramN] = Newton(epsilon,eps,g1,uh0);
212 end
213 maxuh = max(uh);
214 maxuh0 = max(uh0);
215 [u_real,mod_u_real,maxi] = data4(num_knot); % Comparação com dados de campo
216 pos = 0; row = 0;
217 for line = 1 : num_el % reordenação de índices para uso em comando
218     % trisurf
219     col = num_knot_el(line);
220     if (type_el(line) == 2)
221         if (col == 3)
222             row = row + 1;
223             index_knot(row,1:3) = zeros(1,3);
224             index_knot(row,1:3) = disp_knot((pos + 1) : (3 + pos));
225             pos = pos + col;
226         else
227             if (col == 6)
228                 row = row + 1;
229                 index_knot(row,1:3) = zeros(1,3);
230                 index_knot(row,1) = disp_knot(1 + pos);
231                 index_knot(row,2) = disp_knot(2 + pos);
232                 index_knot(row,3) = disp_knot(4 + pos);
233                 row = row + 1;
234                 index_knot(row,1:3) = zeros(1,3);
235                 index_knot(row,1) = disp_knot(2 + pos);
236                 index_knot(row,2) = disp_knot(5 + pos);
237                 index_knot(row,3) = disp_knot(4 + pos);
238                 row = row + 1;
239                 index_knot(row,1:3) = zeros(1,3);
240                 index_knot(row,1) = disp_knot(2 + pos);
241                 index_knot(row,2) = disp_knot(3 + pos);

```

```

242         index_knot(row,3) = disp_knot(5 + pos);
243         row = row + 1;
244         index_knot(row,1:3) = zeros(1,3);
245         index_knot(row,1) = disp_knot(4 + pos);
246         index_knot(row,2) = disp_knot(5 + pos);
247         index_knot(row,3) = disp_knot(6 + pos);
248         pos = pos + col;
249     end
250 end
251 else
252     pos = pos + col;
253 end
254 end
255 figure(figura);                % plotando resultados
256 trisurf(index_knot+1,coord_knot(:,1),coord_knot(:,2),uh'),colorbar;
257 shading interp;
258 colormap('jet');
259 view(0,90)
260 distance = 0; ind = 0; down = 0; flat = 0; up = 0;
261 for line = 1 : num_knot
262     if u_real(line) ~= 0
263         ind = ind + 1;
264         vind(ind) = line;
265         urC(ind) = u_real(line);
266         uhOC(ind) = uh0(line);        % contagem de dados amostrados
267         if urC(ind) > uhOC(ind)      % acima dos aproximados
268             down = down + 1;
269         elseif urC(ind) == uhOC(ind) % iguais aos aproximados
270             flat = flat + 1;
271         else
272             up = up + 1;              % abaixo dos aproximados
273         end
274         distance = distance + (uhOC(ind) - urC(ind)) * (uhOC(ind) - urC(ind));
275     end
276 end
277 distance = sqrt(distance);          % desvio

```

## 1.2 Montagem das matrizes de contribuição local $m^e$ e $\sigma^e$

```

1 function [me,sigma_e,disp_knot_el,posi]=...
2         local_matrix_M(type_el,type_mat,Ne,indxrI,posi)
3
4 global coef_mat
5 global disp_knot
6 global coord_knot
7     type_mat = type_mat + 1;

```

```

8   alpha = coef_mat(type_mat,1); % coeficiente de dispersao
9   V1 = coef_mat(type_mat,2);    % componente na direcao x do campo de vetores
10  V2 = coef_mat(type_mat,3);    % componente na direcao Y do campo de vetores
11  b = coef_mat(type_mat,4);     % medida de hostilidade do meio
12  gamma = coef_mat(type_mat,5); % fluxo em de fronteira
13  disp_knot_el = disp_knot((posi + 1) : (posi + Ne));
14  for index = 1 : Ne
15      i = disp_knot_el(index) + 1;
16      coord_knot_el(index,:) = coord_knot(i,:);
17  end
18  switch type_el
19  case 0 % rotinas de integração sobre fronteira
20      [me,sigma_e] = integ_seg(coord_knot_el,Ne,alpha,V1,V2,b,gamma,indexrI);
21      posi = posi + Ne;
22  case 1 % rotinas de integração sobre elementos quadrados
23      [me,sigma_e] = integ_sq(coord_knot_el,Ne,alpha,V1,V2,b,gamma,indexrI);
24      posi = posi + Ne;
25  case 2 % rotinas de integração sobre elementos triangulares
26      [me,sigma_e] = integ_tri(coord_knot_el,Ne,alpha,V1,V2,b,gamma,indexrI);
27      posi = posi + Ne;
28  end

```

### 1.3 “Assembly” das matrizes $M$ e $\Sigma$

```

1  function [M,Sigma]=global_matrix_M(me,sigma_e,disp_knot_el,posf)
2      % "assembly" da matriz de rigidez
3      % (parte linear do sistema)
4      global M          % Variáveis Globais
5      global Sigma
6      for i = 1 : posf
7          disp_i = disp_knot_el(i) + 1;
8          Sigma(disp_i) = Sigma(disp_i) + sigma_e(i);
9          for j = 1 : posf
10             disp_j = disp_knot_el(j) + 1;
11             M(disp_i,disp_j) = M(disp_i,disp_j) + me(i,j);
12         end
13     end

```

### 1.4 Acrescentando fontes pontuais (“populações nodais”)

```

1  function Sigma = load_vec(num_nodal_load,disp_load,val_load)
2
3  global Sigma
4  if num_nodal_load ~= 0
5      for i = 1 : num_nodal_load

```

```

6     displ = disp_load(i) + 1;
7     val = val_load(i);
8     Sigma(displ) = Sigma(displ) + val;
9     end
10  else
11     Sigma = Sigma;
12  end
13

```

## 1.5 Acrescentando condições de contorno pontuais

```

1  function [M,Sigma] = cdc_point(num_knot_cdc,knot_cdc,type_cdc,cdc);
2
3  global M
4  global Sigma          % Variáveis Globais
5  for k = 1 : num_knot_cdc
6     u0 = cdc(k,1); p0 = cdc(k,2);
7     switch type_cdc(k)
8     case 0              % condição Dirichlet
9         M(knot_cdc(k) + 1,:) = 0;
10        M(:,knot_cdc(k) + 1) = 0;
11        M(knot_cdc(k) + 1,knot_cdc(k) + 1) = 1;
12        Sigma(knot_cdc(k) + 1) = u0;
13    case 1              % condição mista
14        M(knot_cdc(k) + 1,knot_cdc(k) + 1) = M(knot_cdc(k) + 1,knot_cdc(k) + 1) + p0;
15        Sigma(knot_cdc(k) + 1) = Sigma(knot_cdc(k) + 1) + p0 * u0;
16    case 2              % condição Neumann
17        Sigma(knot_cdc(k) + 1) = Sigma(knot_cdc(k) + 1) + p0 * u0;
18    end
19  end

```

## 1.6 Rotinas de construção da matriz N

```

1  function N = matrix_N(vec)
2
3  global index_rule_I_el % Construindo N (matriz não-linear no sistema)
4  global num_knot_el     % a partir de contribuições locais de matrizes ne
5  global type_el
6  global num_el
7  posi = 0;
8  for e = 1 : num_el     % contribuições locais
9     [ne,disp_knot_el,posi] = ...
10     local_matrix_N(vec,type_el(e),num_knot_el(e),index_rule_I_el(e),posi);
11     N = global_matrix_N(ne,disp_knot_el,num_knot_el(e));
12  end

```



## 1.7 Adaptando a matriz N às condições de contorno pontuais

```
1 function N = cdc_point_N(num_knot_cdc,knot_cdc,type_cdc);
2
3     global N                % Variáveis Globais
4     for k = 1 : num_knot_cdc
5         switch type_cdc(k)
6             case 0          % condição Dirichlet
7                 N(knot_cdc(k) + 1,:)= 0;
8                 N(:,knot_cdc(k) + 1)= 0;
9             end
10    end
```

## 1.8 Rotinas para Método de Newton

```
1 function [uh0,iter2] = Newton(epsilon,eps,g1,u0);
2
3     global num_knot_cdc
4     global knot_cdc
5     global type_cdc
6     global num_knot
7     global M
8     global Sigma
9     global lambda
10    global K
11
12    iter2 = 0; eps = eps * eps; uh0 = uh0';
13    while (epsilon > eps)
14        iter2 = iter2 + 1;
15        N = sparse(num_knot,num_knot);
16        N = matrix_N(uh0);
17        if num_knot_cdc ~= 0          % Adaptando N às CDC pontuais
18            N = cdc_point_N(num_knot_cdc,knot_cdc,type_cdc);
19        end
20        JF = M + 2 * (lambda / K) * N;
21        R = M + (lambda / K) * N;
22        F = R * uh0 - Sigma;
23        [L,U,P] = lu(JF);          % resolvendo o sistema linear a cada iteração
24        F = (-1) * P * F;
25        v = L \ F;
26        s = U \ v;
27        uh = uh0 + s;
28        epsilon = norm(uh - uh0,2);
29        uh0 = uh;
30    end
```

## 1.9 Dados de amostragem

```
1 function [u_real,mod_u_real,mini] = fenom(num_knot)
2
3 u_real = zeros(1,num_knot);
4 u_real(11) = 0.0130;
5 u_real(12) = 0.0235;
6 u_real(13) = 0.0330;
7 u_real(14) = 0.0528;
8 u_real(15) = 0.0129;
9 u_real(16) = 0.0231;
10 u_real(17) = 0.0433;
11 u_real(18) = 0.0829;
12 u_real(19) = 0.0085;
13 u_real(20) = 0.0128;
14 u_real(21) = 0.0688;
15 u_real(22) = 0.0920;
16 u_real(23) = 0.0055;
17 u_real(24) = 0.0125;
18 u_real(25) = 0.0738;
19 u_real(26) = 0.0955;
20
21 u_real = u_real - 0.00;
22 ind = 0; v = zeros(1,16);
23 for line = 1 : num_knot
24     if u_real ~= 0
25         ind = ind + 1;
26         v(ind) = u_real(line);
27     end
28 end
29 mini = min(v);
30 mod_u_real = sqrt(sum(u_real .* u_real));
```

## 1.10 Montagem das matrizes de contribuição local $n^e$

```
1 function [ne,disp_knot_el,posi]=local_matrix_N(uh0,type_el,Ne,indexrI,posi)
2
3 global coef_mat
4 global disp_knot
5 global coord_knot
6 disp_knot_el = disp_knot((posi + 1) : (posi + Ne));
7 for index = 1 : Ne
8     i = disp_knot_el(index) + 1;
9     uh0_e(index) = uh0(i);
10    coord_knot_el(index,:) = coord_knot(i,:);
11 end
```

```

12     switch type_el
13     case 0 % segmento em fronteira
14         ne = integ_seg_N(type_el,uh0_e,coord_knot_el,Ne,indexrI);
15         posi = posi + Ne;
16     case 1 % quadrado
17         ne = integ_sq_N(uh0_e,coord_knot_el,Ne,indexrI);
18         posi = posi + Ne;
19     case 2 % triangulo
20         ne = integ_tri_N(uh0_e,coord_knot_el,Ne,indexrI);
21         posi = posi + Ne;
22     end

```

### 1.11 “Assembly” da matriz N

```

1 function N=global_matrix_N(ne,disp_knot_el,posf)
2         % "assembly" da matriz de rigidez
3         % (parte linear do sistema)
4     global N % Variáveis Globais
5     for i = 1 : posf
6         dispi = disp_knot_el(i) + 1;
7         for j = 1 : posf
8             dispj = disp_knot_el(j) + 1;
9             N(dispi,dispj) = N(dispi,dispj) + ne(i,j);
10        end
11    end

```

### 1.12 Rotinas para integração sobre fronteiras – montagem de M

```

1 function [me,sigma_e] = integ_seg(coord_knot_el,Ne,alpha,V1,V2,b,gamma,indexrI);
2         % construção de matriz de rigidez e vetor de carga elementeres
3     global matrix % Variáveis Globais
4     global lambda
5     N1 = matrix(indexrI + 1,1);
6     z_Legendre = zeros(N1,2);
7     w1 = zeros(1,N1);
8     pos = 1;
9     for k = 1 : N1 % zeros de Legendre
10        z_Legendre(k,:) = matrix((indexrI + 1),(pos + 1) : (pos + 2));
11        pos = pos + 2;
12    end
13        % pesos na integração de Gauss
14    w1 = matrix(indexrI + 1,(pos + 1) : (2 * N1 + N1 + 1));
15    me = zeros(Ne,Ne); sigma_e = zeros(Ne,1);
16    qsi = linspace(-1,1,Ne);
17    for n = 1 : N1

```

```

18     [psi,dpsi] = shape1D(qsi,z_Legendre(n,1),Ne);
19     psi = psi';
20     dpsi = dpsi';
21     dxdqsi = 0; dydqsi = 0;
22     for i = 1 : Ne           % Jacobiano da transformação
23         dxdqsi = dxdqsi + coord_knot_el(i,1) * dpsi(i);
24         dydqsi = dydqsi + coord_knot_el(i,2) * dpsi(i);
25     end
26     dsdqsi = sqrt(dxdqsi * dxdqsi + dydqsi * dydqsi);
27     factor = dsdqsi * wl(n);
28     sigma_e = sigma_e + psi * factor;
29     for i = 1 : Ne
30         for j = 1 : Ne
31             psii = psi(i);
32             psij = psi(j);
33             dpsii = dpsi(i);
34             dpsij = dpsi(j);
35             me(i,j) = me(i,j) + factor *...
36                 (alpha * dpsii * dpsij * (1 / dsdqsi) * (1 / dsdqsi) +...
37                 V1 * dpsij * psii * (1 / dsdqsi) + (b - lambda) * psii * psij);
38         end
39     end
40     end
41     sigma_e = gamma * sigma_e;

```

### 1.13 Rotinas para integração sobre elementos quadrados – montagem de M

```

1 function [me,sigma_e] = integ_sq(coord_knot_el,Ne,alpha,V1,V2,b,gamma,indxrl);
2
3     global matrix
4     global lambda
5     N1 = matrix(indxrl + 1,1);
6     z_Legendre = zeros(N1,2);
7     wl = zeros(1,N1);
8     pos = 1;
9     for k = 1 : N1
10        z_Legendre(k,:) = matrix((indxrl + 1),(pos + 1) : (pos + 2));
11        pos = pos + 2;
12    end
13    wl = matrix(indxrl + 1,(pos + 1) : (2 * N1 + N1 + 1));
14    me = zeros(Ne,Ne); sigma_e = zeros(Ne,1);
15    for n = 1 : N1
16        [PSI_hat,dPSI_dqsi_hat,dPSI_deta_hat] =...
17        shape2D_sq(z_Legendre(n,1),z_Legendre(n,2),Ne);

```

```

18     PSI_hat = PSI_hat';
19     dPSI_dqsi_hat = dPSI_dqsi_hat';
20     dPSI_deta_hat = dPSI_deta_hat';
21         % Jacobiano da transformação
22     J = Jacobian(dPSI_dqsi_hat,dPSI_deta_hat,coord_knot_el,Ne);
23     absdetJ = abs(det(J));
24     factor = absdetJ * wl(n);
25     sigma_e = sigma_e + PSI_hat * factor;
26     for i = 1 : Ne
27         for j = 1 : Ne
28             I = J' \ [dPSI_dqsi_hat(i) dPSI_deta_hat(i)]';
29             II = J' \ [dPSI_dqsi_hat(j) dPSI_deta_hat(j)]';
30             me(i,j) = me(i,j) + factor *...
31                 (alpha * (I(1) * II(1) + I(2) * II(2)) +...
32                 (V1 * II(1) + V2 * II(2)) * PSI_hat(i) +...
33                 (b - lambda) * PSI_hat(i) * PSI_hat(j));
34         end
35     end
36 end
37 sigma_e = gamma * sigma_e;

```

## 1.14 Rotinas para integração sobre elementos triangulares – montagem de M

```

1 function [me,sigma_e] = integ_tri(coord_knot_el,Ne,alpha,V1,V2,b,gamma,indexrI);
2     global matrix
3     global lambda
4     N1 = matrix(indexrI + 1,1);
5     z_Legendre = zeros(N1,2);
6     wl = zeros(1,N1);
7     pos = 1;
8     for k = 1 : N1
9         z_Legendre(k,:) = matrix((indexrI + 1),(pos + 1) : (pos + 2));
10        pos = pos + 2;
11    end
12    wl = matrix(indexrI + 1,(pos + 1) : (2 * N1 + N1 + 1));
13    me = zeros(Ne,Ne); sigma_e = zeros(Ne,1);
14    for n = 1 : N1
15        [PSI_hat,dPSI_dqsi_hat,dPSI_deta_hat] =...
16            shape2D_tri(z_Legendre(n,1),z_Legendre(n,2),Ne);
17        PSI_hat = PSI_hat';
18        dPSI_dqsi_hat = dPSI_dqsi_hat';
19        dPSI_deta_hat = dPSI_deta_hat';
20        % Jacobiano da transformação
21        J = Jacobian(dPSI_dqsi_hat,dPSI_deta_hat,coord_knot_el,Ne);

```

```

22     absdetJ = abs(det(J));
23     factor = absdetJ * wl(n);
24     sigma_e = sigma_e + PSI_hat * factor;
25     for i = 1 : Ne
26         for j = 1 : Ne
27             I = J' \ [dPSI_dqsi_hat(i) dPSI_deta_hat(i)]';
28             II = J' \ [dPSI_dqsi_hat(j) dPSI_deta_hat(j)]';
29             me(i,j) = me(i,j) + factor *...
30                 (alpha * (I(1) * II(1) + I(2) * II(2)) +...
31                 (V1 * II(1) + V2 * II(2)) * PSI_hat(i) +...
32                 (b - lambda) * PSI_hat(i) * PSI_hat(j));
33         end
34     end
35 end
36 sigma_e = gamma * sigma_e;

```

## 1.15 Rotinas para integração sobre fronteiras – montagem de N

```

1 function ne = integ_seg_N(type_el,uh0_e,coord_knot_el,Ne,indxrI);%uh0
2
3 global matrix
4 N1 = matrix(indxrI + 1,1);
5 z_Legendre = zeros(N1,2);
6 wl = zeros(1,N1);
7 pos = 1;
8 for k = 1 : N1
9     z_Legendre(k,:) = matrix((indxrI + 1),(pos + 1) : (pos + 2));
10    pos = pos + 2;
11 end
12 wl = matrix(indxrI + 1,(pos + 1) : (2 * N1 + N1 + 1));
13 ne = zeros(Ne,Ne);
14 qsi = linspace(-1,1,Ne);
15 if type_el ~= 0
16     for n = 1 : N1
17         [psi,dpsi] = shape1D(qsi,z_Legendre(n,1),Ne);
18         psi = psi';
19         dpsi = dpsi';
20         dxdqsi = 0; dydqsi = 0;
21         for i = 1 : Ne % Jacobiano da transformação
22             dxdqsi = dxdqsi + coord_knot_el(i,1) * dpsi(i);
23             dydqsi = dydqsi + coord_knot_el(i,2) * dpsi(i);
24         end
25         dsdqsi = sqrt (dxdqsi * dxdqsi + dydqsi * dydqsi);
26         factor = dsdqsi * wl(n);
27         s = 0;
28         for l = 1 : Ne

```

```

29         s = s + psi(1) * uh0_e(1);
30     end
31     for i = 1 : Ne
32         psii = psi(i);
33         for j = 1 : Ne
34             psij = psi(j);
35             ne(i,j) = ne(i,j) + factor * psij * s * psii;
36         end
37     end
38 end
39 end

```

## 1.16 Rotinas para integração sobre elementos quadrados – montagem de N

```

1 function ne = integ_sq_N(uh0_e, coord_knot_el, Ne, indexrI);
2
3 global matrix
4 Nl = matrix(indexrI + 1,1);
5 z_Legendre = zeros(Nl,2);
6 wl = zeros(1,Nl);
7 pos = 1;
8 for k = 1 : Nl
9     z_Legendre(k,:) = matrix((indexrI + 1),(pos + 1) : (pos + 2));
10    pos = pos + 2;
11 end
12 wl = matrix(indexrI + 1,(pos + 1) : (2 * Nl + Nl + 1));
13 ne = zeros(Ne,Ne);
14 for n = 1 : Nl
15     [PSI_hat,dPSI_dqsi_hat,dPSI_deta_hat] =...
16         shape2D_sq(z_Legendre(n,1),z_Legendre(n,2),Ne);
17     PSI_hat = PSI_hat';
18     dPSI_dqsi_hat = dPSI_dqsi_hat';
19     dPSI_deta_hat = dPSI_deta_hat';
20         % Jacobiano da transformação
21     J = Jacobian(dPSI_dqsi_hat,dPSI_deta_hat,coord_knot_el,Ne);
22     absdetJ = abs(det(J));
23     factor = absdetJ * wl(n);
24     s = 0;
25     for l = 1 : Ne
26         s = s + PSI_hat(l) * uh0_e(l);
27     end
28     for i = 1 : Ne
29         PSII = PSI_hat(i);
30         for j = 1 : Ne

```

```

31     PSIj = PSI_hat(j);
32     ne(i,j) = ne(i,j) + factor * (PSIj * s * PSII);
33     end
34     end
35 end
36

```

## 1.17 Rotinas para integração sobre elementos triangulares – montagem de N

```

1  function ne = integ_tri_N(uh0_e,coord_knot_el,Ne,indexrI);
2
3  global matrix
4  Nl = matrix(indexrI + 1,1);
5  z_Legendre = zeros(Nl,2);
6  wl = zeros(1,Nl);
7  pos = 1;
8  for k = 1 : Nl
9      z_Legendre(k,:) = matrix((indexrI + 1),(pos + 1) : (pos + 2));
10     pos = pos + 2;
11 end
12 wl = matrix(indexrI + 1,(pos + 1) : (2 * Nl + Nl + 1));
13 ne = zeros(Ne,Ne);
14 for n = 1 : Nl
15     [PSI_hat,dPSI_dqsi_hat,dPSI_deta_hat] = ...
16     shape2D_tri(z_Legendre(n,1),z_Legendre(n,2),Ne);
17     PSI_hat = PSI_hat';
18     dPSI_dqsi_hat = dPSI_dqsi_hat';
19     dPSI_deta_hat = dPSI_deta_hat';
20     % Jacobiano da transformação
21     J = Jacobian(dPSI_dqsi_hat,dPSI_deta_hat,coord_knot_el,Ne);
22     absdetJ = abs(det(J));
23     factor = absdetJ * wl(n);
24     s = 0;
25     for l = 1 : Ne
26         s = s + PSI_hat(l) * uh0_e(l);
27     end
28     for i = 1 : Ne
29         for j = 1 : Ne
30             ne(i,j) = ne(i,j) + factor * (PSI_hat(j) * s * PSI_hat(i));
31         end
32     end
33 end
34

```



## 1.18 Funções base unidimensionais

```
1 function [psi,dpsi] = shape1D(qsi,x,dim);
2
3 if dim == 2
4     psi(1) = (x - qsi(2)) / (qsi(1) - qsi(2));
5     dpsi(1) = 1 / (qsi(1) - qsi(2));
6     psi(2) = (x - qsi(1)) / (qsi(2) - qsi(1));
7     dpsi(2) = 1 / (qsi(2) - qsi(1));
8 else
9     qsi_i = qsi(2 : dim);
10    n = x - qsi_i;
11    qsii = qsi(1);
12    d = qsii - qsi_i;
13    qsi_i = n ./ d;
14    psi(1) = prod(qsi_i);
15    s = 0;
16    for j = 1 : (dim - 1)
17        aux = qsi_i(j);
18        qsi_i(j) = 1 / d(j);
19        dpsii = prod(qsi_i);
20        dpsii = s + dpsii;
21        s = dpsii;
22        qsi_i(j) = aux;
23    end
24    dpsi(1) = dpsii;
25    for i = 2 : (dim - 1)
26        qsi_i = [qsi(1 : (i - 1)) qsi((i + 1) : dim)];
27        n = x - qsi_i;
28        qsii = qsi(i);
29        d = qsii - qsi_i;
30        qsi_i = n ./ d;
31        psi(i) = prod(qsi_i);
32        s = 0;
33        for j = 1 : (dim - 1)
34            aux = qsi_i(j);
35            qsi_i(j) = 1 / d(j);
36            dpsii = prod(qsi_i);
37            dpsii = s + dpsii;
38            s = dpsii;
39            qsi_i(j) = aux;
40        end
41        dpsi(i) = dpsii;
42    end
43    qsi_i = qsi(1 : (dim - 1));
44    n = x - qsi_i;
45    qsii = qsi(dim);
```

```

46     d = qsi_i - qsi_i;
47     qsi_i = n ./ d;
48     psi(dim) = prod(qsi_i);
49     s = 0;
50     for j = 1 : (dim - 1)
51         aux = qsi_i(j);
52         qsi_i(j) = 1 / d(j);
53         dpsidim = prod(qsi_i);
54         dpsidim = s + dpsidim;
55         s = dpsidim;
56         qsi_i(j) = aux;
57     end
58     dpsidim = dpsidim;
59 end

```

## 1.19 Funções base bidimensionais sobre elementos quadrados

```

1 function [PSI_hat,dPSI_dqsi_hat,dPSI_deta_hat] = shape2D_sq(qsil,etal,Ne);
2
3 true = 0;
4 if Ne == 8
5     true = 1;
6     Ne = Ne + 1;
7 end
8 dim = sqrt(Ne);
9 qsi = linspace(-1,1,dim);
10 [psiqsi,dpsidqsi] = shape1D(qsi,qsil,dim);
11 if qsil == etal
12     psieta = psiqsi;
13     dpsideta = dpsidqsi;
14 else
15     [psieta,dpsideta] = shape1D(qsi,etal,dim);
16 end
17 k = 1;
18 for j = 1 : dim
19     for i = 1 : dim
20         PSI_hat(k) = psiqsi(i) * psieta(j);
21         dPSI_dqsi_hat(k) = dpsidqsi(i) * psieta(j);
22         dPSI_deta_hat(k) = psiqsi(i) * dpsideta(j);
23         k = k + 1;
24     end
25 end
26 if true == 1
27     c = [-1/4 1/2 -1/4 1/2 0 1/2 -1/4 1/2 -1/4];
28     for k = 1 : 4
29         PSI_hat(k) = PSI_hat(k) + c(k) * PSI_hat(5);

```

```

30     dPSI_dqsi_hat(k) = dPSI_dqsi_hat(k) + c(k) * dPSI_dqsi_hat(5);
31     dPSI_deta_hat(k) = dPSI_deta_hat(k) + c(k) * dPSI_deta_hat(5);
32 end
33 for k = 6 : 9
34     PSI_hat(k) = PSI_hat(k) + c(k) * PSI_hat(5);
35     dPSI_dqsi_hat(k) = dPSI_dqsi_hat(k) + c(k) * dPSI_dqsi_hat(5);
36     dPSI_deta_hat(k) = dPSI_deta_hat(k) + c(k) * dPSI_deta_hat(5);
37 end
38 PSI_hat = [PSI_hat(1 : 4) PSI_hat(6 : 9)];
39 dPSI_dqsi_hat = [dPSI_dqsi_hat(1 : 4) dPSI_dqsi_hat(6 : 9)];
40 dPSI_deta_hat = [dPSI_deta_hat(1 : 4) dPSI_deta_hat(6 : 9)];
41 end

```

## 1.20 Funções base bidimensionais sobre elementos triangulares

```

1 function [PSI_hat,dPSI_dqsi_hat,dPSI_deta_hat] = shape2D_tri(qsil,etal,Ne);
2
3     i = 2;
4     ne = 1;
5     while ne ~= Ne
6         dim = i;
7         ne = ne + i;
8         i = i + 1;
9     end
10    vec = linspace(0,1,dim);
11    m = 1;
12    for j = 1 : dim
13        vecj = vec(j);
14        for i = 1 : (dim - j + 1)
15            veci = vec(i);
16            if i ~= 1
17                vec_zeta = vec(1 : (i - 1));
18                n = qsil - vec_zeta;
19                d = veci - vec_zeta;
20                vec_zeta = n ./ d;
21                zeta1 = prod(vec_zeta);
22                s = 0;
23                for k = 1 : (i - 1)
24                    aux = vec_zeta(k);
25                    vec_zeta(k) = 1 / d(k);
26                    dpsidzeta1 = prod(vec_zeta);
27                    dpsidzeta1 = s + dpsidzeta1;
28                    s = dpsidzeta1;
29                    vec_zeta(k) = aux;
30                end
31            else

```

```

32         zeta1 = 1;
33         dpsidzeta1 = 0;
34     end
35     if j ~= 1
36         vec_zeta = vec(1 : (j - 1));
37         n = etal - vec_zeta;
38         d = vecj - vec_zeta;
39         vec_zeta = n ./ d;
40         zeta2 = prod(vec_zeta);
41         s = 0;
42         for k = 1 : (j - 1)
43             aux = vec_zeta(k);
44             vec_zeta(k) = 1 / d(k);
45             dpsidzeta2 = prod(vec_zeta);
46             dpsidzeta2 = s + dpsidzeta2;
47             s = dpsidzeta2;
48             vec_zeta(k) = aux;
49         end
50     else
51         zeta2 = 1;
52         dpsidzeta2 = 0;
53     end
54     if (i + j) <= dim
55         vec_zeta = vec((i + j) : dim);
56         n = qsil + etal - vec_zeta;
57         d = veci + vecj - vec_zeta;
58         vec_zeta = n ./ d;
59         zeta3 = prod(vec_zeta);
60         s = 0;
61         for k = (i + j) : dim
62             aux = vec_zeta(k - (i + j) + 1);
63             vec_zeta(k - (i + j) + 1) = 1 / d(k - (i + j) + 1);
64             dpsidzeta3 = prod(vec_zeta);
65             dpsidzeta3 = s + dpsidzeta3;
66             s = dpsidzeta3;
67             vec_zeta(k - (i + j) + 1) = aux;
68         end
69     else
70         zeta3 = 1;
71         dpsidzeta3 = 0;
72     end
73     PSI_hat(m) = zeta1 * zeta2 * zeta3;
74     dPSI_dqsi_hat(m) = (dpsidzeta1 * zeta3 + zeta1 * dpsidzeta3) * zeta2;
75     dPSI_deta_hat(m) = (dpsidzeta2 * zeta3 + zeta2 * dpsidzeta3) * zeta1;
76     m = m + 1;
77 end

```

78        end

## 1.21    Construção da Matriz Jacobiana

```
1 function J = Jacobian(dpsidqsi,dpsideta,coord_knot_el,Ne)
2                    % jacobiano da transformação
3     dxdqsi = 0; dxdeta = 0; dydqsi = 0; dydeta = 0;
4     for k = 1 : Ne
5         dxdqsi = dxdqsi + coord_knot_el(k,1) * dpsidqsi(k);
6         dxdeta = dxdeta + coord_knot_el(k,1) * dpsideta(k);
7         dydqsi = dydqsi + coord_knot_el(k,2) * dpsidqsi(k);
8         dydeta = dydeta + coord_knot_el(k,2) * dpsideta(k);
9     end
10     J = [dxdqsi dxdeta ; dydqsi dydeta];
```