

ANÁLISE VARIACIONAL E NUMÉRICA DA EQUAÇÃO DE POISSON EM
PROBLEMAS DE TRANSFERÊNCIA DE CALOR

Yuri Paes Gomes

Projeto de Graduação apresentado ao Curso de Engenharia Mecânica da Escola Politécnica, Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Engenheiro.

Orientadores: Gustavo Rabello dos Anjos
Alex Farah Pereira

Rio de Janeiro
Maio de 2025



UNIVERSIDADE FEDERAL DO RIO DE JANEIRO

Departamento de Engenharia Mecânica

DEM/POLI/UFRJ



ANÁLISE VARIACIONAL E NUMÉRICA DA EQUAÇÃO DE POISSON EM
PROBLEMAS DE TRANSFERÊNCIA DE CALOR

Yuri Paes Gomes

PROJETO FINAL SUBMETIDO AO CORPO DOCENTE DO DEPARTAMENTO DE ENGENHARIA MECÂNICA DA ESCOLA POLITÉCNICA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE ENGENHEIRO MECÂNICO.

Aprovada por:

Prof. Gustavo Rabello dos Anjos, PhD

Prof. Alex Farah Pereira, D.Sc.

Prof. Fernando Pereira Duda, D.Sc.

Prof. Luã Guedes Costa, D.Sc.

RIO DE JANEIRO, RJ – BRASIL

MAIO DE 2025

Paes Gomes, Yuri

Análise Variacional e Numérica da Equação de Poisson em Problemas de Transferência de Calor/ Yuri Paes Gomes. – Rio de Janeiro: UFRJ/Escola Politécnica, 2025.

X, 38 p.: il.; 29, 7cm.

Orientadores: Gustavo Rabello dos Anjos

Alex Farah Pereira

Projeto de Graduação – UFRJ/ Escola Politécnica/ Curso de Engenharia Mecânica, 2025.

Referências Bibliográficas: p. 29 – 30.

1. Análise Variacional.
2. Análise Numérica.
3. Transferência de Calor. I. Rabello dos Anjos, Gustavo *et al.* II. Universidade Federal do Rio de Janeiro, UFRJ, Curso de Engenharia Mecânica. III. Análise Variacional e Numérica da Equação de Poisson em Problemas de Transferência de Calor.

À minha mãe

Agradecimentos

Agradeço, primeiramente, à minha família (em especial, minha mãe, a qual sempre esteve disposta como um tremendo suporte emocional durante minha trajetória de vida), amigos e professores, que estiveram comigo e me ensinaram profundamente durante o curso.

Adicionalmente, agradeço ao Governo Federal e à sociedade brasileira pelo custeio deste caro ensino de alta qualidade da Universidade Federal do Rio de Janeiro, o qual não teria quaisquer condições de pagar por recursos próprios durante o período de curso. Dessa maneira, me sinto em total obrigação de fazer jus a esse enorme privilégio e, portanto, trabalharei para retribuir todo esse valor, multiplicado em inúmeras vezes, à UFRJ, ao Brasil e ao mundo.

Resumo do Projeto de Graduação apresentado à Escola Politécnica/UFRJ como parte dos requisitos necessários para a obtenção do grau de Engenheiro Mecânico

ANÁLISE VARIACIONAL E NUMÉRICA DA EQUAÇÃO DE POISSON EM
PROBLEMAS DE TRANSFERÊNCIA DE CALOR

Yuri Paes Gomes

Maio/2025

Orientadores: Gustavo Rabello dos Anjos

Alex Farah Pereira

Programa: Engenharia Mecânica

Durante o curso de Engenharia Mecânica, principalmente no contexto de Transferência de Calor, é introduzida a equação de Poisson, que é essencial para resoluções de problemas em regime permanente. No entanto, a ausência de um estudo matemático rigoroso para discutir a existência de uma solução única e de um método numérico de resolução compatível com os utilizados pela indústria retiram um aprendizado conceitual profundo sobre essa equação tão fundamental para diversos fenômenos físicos. Desse modo, apresenta-se, neste trabalho, um estudo da equação de Poisson pela ótica da Análise Variacional para provar a existência de uma única solução (utilizando o Teorema de Lax-Milgram) e a resolução do problema numericamente (pelo Método de Galerkin).

Abstract of Undergraduate Project presented to POLI/UFRJ as a partial fulfillment of the requirements for the degree of Mechanical Engineer

VARIATIONAL AND NUMERICAL ANALYSIS OF THE POISSON
EQUATION FOR HEAT TRANSFER PROBLEMS

Yuri Paes Gomes

May/2025

Advisors: Gustavo Rabello dos Anjos

Alex Farah Pereira

Department: Mechanical Engineering

During the Mechanical Engineering course, mainly in the context of Heat Transfer, it is introduced the Poisson Equation, which is essential to solve steady state problems. However, the lacking of a rigorous mathematical study for discussing the existence of a unique solution and the presentation of a numerical method that is compatible with the industry standard removes an important deep conceptual learning about this very important equation. In this sense, this work aims to study the Poisson Equation from the Variational Analysis perspective to prove that there exists an unique solution to the equation (using the Lax-Milgram Theorem) and to derive a numerical solution (using the Galerkin Method).

Sumário

Lista de Figuras	x
1 Introdução	1
1.1 Motivação	1
1.2 Objetivo	1
2 Revisão Bibliográfica	2
2.1 Equações Diferenciais Parciais	2
2.1.1 Definição Geral	2
2.1.2 A Equação de Poisson	3
2.2 Transferência de Calor	3
2.2.1 Condução	3
2.2.2 Convecção	4
2.2.3 Radiação	5
2.3 A Equação do Calor	5
2.4 Método dos Elementos Finitos	6
3 Análise Funcional	7
3.1 Espaços de Hilbert	7
3.2 Medida de Lebesgue	11
3.3 Espaços de Sobolev	12
3.4 Teorema de Lax-Milgram	12
3.5 Existência e Unicidade da Equação de Poisson	14
4 Estudo Numérico	17
4.1 O Método dos Elementos Finitos	17

4.1.1	O Método de Galerkin	17
4.1.2	Representação Gráfica do Método	19
4.1.3	Elementos e Funções Base	20
5	Resultados	23
5.1	Estudo de Caso	23
5.1.1	Solução Manufaturada	24
5.1.2	Solução Numérica	24
6	Conclusões e Trabalhos Futuros	28
	Referências Bibliográficas	29
A	Código Fonte	31

Lista de Figuras

4.1	Domínio contínuo $\Omega = [0, 1] \times [0, 1]$	19
4.2	Domínio $\Omega = [0, 1] \times [0, 1]$ discretizado em 16 elementos	19
5.1	Domínio $\Omega = [0, 1] \times [0, 1] - [0.45, 0.55] \times [0.45, 0.55]$	23
5.2	Resultado para 64 elementos	25
5.3	Resultado para 256 elementos	25
5.4	Resultado para 1024 elementos	26
5.5	Resultado para 4096 elementos	26
5.6	Resultado para 16384 elementos	27
5.7	Erro relativo em função do número de elementos, com eixo horizontal em escala logarítmica	27

Capítulo 1

Introdução

1.1 Motivação

Na formação do Engenheiro Mecânico, há uma grande lacuna matemática em relação à clareza sobre a existência de soluções únicas para as importantes equações diferenciais parciais que são apresentadas durante o curso de Engenharia Mecânica. Desse modo, este breve estudo expandirá a compreensão conceitual sobre a Equação de Poisson, que é extremamente essencial para vários problemas da engenharia mecânica, sobretudo nas áreas de Transferência de Calor e de Mecânica dos Sólidos.

1.2 Objetivo

O objetivo deste trabalho é provar uma existência única para a Equação de Poisson por métodos da Análise Variacional e mostrar a eficácia do Método de Elementos Finitos na resolução dos problemas em Transferência de Calor. Em especial, neste texto, a temática será o problema de transferência de calor em regime permanente, o qual será estudado analiticamente utilizando a construção do Espaço de Sobolev (a partir da ideia dos Espaços de Banach e de Hilbert, além da noção sobre Medida e Integração de Lebesgue) e do Teorema de Lax-Milgram, além de ser analisado numericamente pelo Método de Galerkin, o qual faz parte da família dos Métodos de Elementos Finitos, e que será facilmente derivado das ferramentas do estudo analítico apresentado.

Capítulo 2

Revisão Bibliográfica

2.1 Equações Diferenciais Parciais

2.1.1 Definição Geral

Segundo Lawrence Evans [1], uma equação diferencial parcial é uma equação envolvendo uma desconhecida função de duas ou mais variáveis e todas ou algumas de suas derivadas parciais. Desse modo, fixando um inteiro $k \geq 1$, D a notação para as derivadas parciais e sendo U um conjunto aberto do \mathbb{R}^n , pode-se fazer a seguinte definição matemática:

Definição 1. *Uma expressão da forma*

$$F(D^k u(x), D^{k-1} u(x), \dots, Du(x), u(x), x) = 0 \quad (x \in U) \quad (2.1)$$

é dita uma equação diferencial parcial de k -ésima ordem, em que

$$F : \mathbb{R}^{n^k} \times \mathbb{R}^{n^{k-1}} \times \dots \times \mathbb{R}^n \times \mathbb{R} \times U \longrightarrow \mathbb{R}$$

é dada e

$$u : U \longrightarrow \mathbb{R}$$

é a incógnita.

Nesse sentido, a equação diferencial parcial é resolvida se for encontrado todas as funções u que verifiquem a equação (2.1) e que satisfaçam certas condições auxiliares chamadas de condições de contorno em alguma parte Γ de ∂U .

2.1.2 A Equação de Poisson

Ainda segundo Lawrence Evans, dado um conjunto aberto $\Omega \subset \mathbb{R}^n$, define-se a equação de Poisson por:

$$-\Delta u = f \quad (2.2)$$

em que $f: \Omega \rightarrow \mathbb{R}^n$ é uma função a qual é chamada de termo fonte e Δ é o operador laplaciano, definido por:

$$\Delta u := \sum_{i=1}^n \frac{\partial^2 u}{\partial x_i^2} \quad (2.3)$$

Nesse sentido, $u: \Omega \subset \mathbb{R}^n \rightarrow \mathbb{R}$ é a função incógnita da equação.

2.2 Transferência de Calor

Segundo Frank Incropera e David Dewitt [2], a transferência de calor é a energia térmica em trânsito devido a uma diferença de temperaturas no espaço. Nesse contexto, a temperatura pode ser definida, segundo Moysés Nussenzveig[3], por uma grandeza física que mede a energia cinética média de cada grau de liberdade. Nesse sentido, é de grande importância quantificar o fluxo de calor para uma ampla gama de atividades da engenharia, para o dimensionamento correto de projetos.

Em relação ao campo da Transferência de Calor, essa pode ser dividida em 3 grandes fenômenos: a condução, a convecção e a radiação, os quais, na natureza, tendem a ocorrer de forma simultânea nos processos de transferência de calor. Porém, por questões de modelagem e de simplificação matemática, é possível desprezar algum dos fenômenos em relação ao(s) outro(s) para resolução do sistema.

A seguir, será revisado os conceitos de condução, convecção e radiação.

2.2.1 Condução

A condução é definida como uma forma de transferência de calor por meio da existência de um diferencial de energia cinética média entre corpos diferentes, ou seja

diferença de temperatura entre esses. Nesse contexto, seu sentido é observado empiricamente como um fluxo de calor \dot{q} entre de um ponto mais alta de temperatura para um ponto mais baixo de temperatura.

Nesse contexto, o Físico e Matemático Joseph Fourier [2] foi o primeiro a modelar esse fenômeno, indicando que o fluxo de calor é proporcional ao gradiente de Temperatura pelo produto entre a área A por qual passa o fluxo de calor e a condutividade térmica k que depende do material, como visto na equação (2.4).

$$\dot{q} = -kA\nabla T. \quad (2.4)$$

Note que o sinal negativo na equação é devido ao sentido da troca de calor (da temperatura mais alta para temperatura mais baixa).

2.2.2 Convecção

A convecção é o fenômeno de transferência de calor que ocorre quando existe uma diferença de temperatura entre um elemento parado e um fluido em movimento.

A modelagem para o fenômeno da Convecção é atribuída à Issac Newton[2]. Nesse sentido, diz-se que a transferência de calor entre os componentes é proporcional à diferença de temperatura entre o fluido T_f e superfície T pelo produto entre a área A na qual ocorre a troca de calor e uma constante de convecção h , como visto na equação (2.5).

$$\dot{q} = hA(T_f - T). \quad (2.5)$$

Nesse contexto, note que a equação está escrita de forma que, no caso do fluxo de calor trocado ser negativo, significa que o objeto está transferindo calor para o fluido.

2.2.3 Radiação

Por fim, a radiação é o fenômeno de transferência de calor na forma de ondas eletromagnéticas segundo o eletromagnetismo clássico ou pela transmissão de fótons discretos segundo a hipótese de Planck[2], não precisando de um meio massivo para se propagar, diferentemente das outras formas de transmissão de calor descritas acima.

A equação que modela esse tipo de transmissão de calor é a chamada Lei de Stefan-Boltzmann, descrita na equação (2.6), em que um corpo de temperatura T_1 e de emissividade ϵ com área de troca A troca uma quantidade de calor q com um corpo de temperatura T_2 , dado a uma proporcionalidade com a constante de Stefan σ .

$$\dot{q} = -\epsilon A \sigma (T_1^4 - T_2^4). \quad (2.6)$$

2.3 A Equação do Calor

Considere um sólido homogêneo e isotrópico de volume infinitesimal contínuo $dV = dx dy dz$ em um domínio Ω e com contorno Γ . Então, pelo balanço de energia, o acúmulo de energia em dV deve ser igual à divergência do fluxo de calor por condução somado a um termo fonte de geração de calor no volume Q_g . Desse modo, obtém-se a famosa *equação do calor*:

$$\rho c_p \frac{\partial T}{\partial t} = -k \nabla \cdot (\nabla T) + Q_g \quad (2.7)$$

$$\rho c_p \frac{\partial T}{\partial t} = -k \Delta T + Q_g \quad (2.8)$$

em que ρ representa massa específica do material e c_p o seu calor específico a volume constante.

Nesse contexto, note que, em regime permanente (isto é, $\frac{\partial T}{\partial t} = 0$), têm-se

$$-\Delta T = -Q_g/k \quad (2.9)$$

que é uma equação de Poisson, a qual será tratada neste texto.

2.4 Método dos Elementos Finitos

Dado que nem sempre é possível obter uma solução simples para problemas descritos por equações diferenciais parciais, a necessidade por métodos que aproximem a solução com a maior precisão e acurácia possível é fundamental. Dentre essa classe de métodos, chamados de métodos numéricos, destaca-se o Método de Elementos Finitos para resolução dos problemas.

Em termos operacionais, o Método dos Elementos Finitos, por sua vez, consiste em discretizar o domínio contínuo em múltiplos elementos interconectados e não sobrepostos; definir funções de base para representar a função a ser determinada e montar um sistema linear.

Esse operacional, por sua vez, é baseado na forma fraca das equações diferenciais, que é dada pela integração das equações sobre o domínio, criando os chamados *funcionais*.

Capítulo 3

Análise Funcional

Para que a equação (3.1) tenha valor em termos físicos e de engenharia, deve-se provar que existe um conjunto finito de soluções para a equação. Desse modo, será provada a existência de solução única para o problema (com condições de contorno de Robin, para generalizar a demonstração):

$$-\Delta T = f \quad \text{em } \Omega \quad (3.1)$$

$$\alpha T + \nabla T \cdot \mathbf{n} = g \quad (\text{com } \alpha > 0) \quad \text{em } \partial\Omega. \quad (3.2)$$

Para desenvolver a prova, a equação (3.1) será multiplicada por uma função teste $v \in H^1$. Dessa forma, para continuar a demonstração, é importante definir o espaço H^1 (Sobolev). Para isso, estudemos algumas resultados que serão úteis para o desenvolvimento do problema, como o Teorema de Lax-Milgram, cuja prova pode ser conferida em Brezis [4]. Porém, antes de enunciar esse teorema é preciso definir conceitos preliminares importantes baseados em Kreyszig e Bazán, Pereira, Fernandes [5][6]

3.1 Espaços de Hilbert

Antes de compreender o que é um espaço de Hilbert, um conceito extremamente importante para este estudo, é importante entender sobre espaços de Banach, os quais são definidos a seguir.

Definição 2. *Um Espaço de Banach é um espaço vetorial normado completo.*

Para compreender essa definição, é preciso entender o significado de um *espaço vetorial*, de um *espaço vetorial normado* e, por fim, de um *espaço vetorial normado completo*.

Desse modo, para começar, define-se um espaço vetorial como todo aquele que respeita os 8 axiomas a seguir para cada vetores \mathbf{u}, \mathbf{v} e \mathbf{w} e para a, b escalares.

1. **Associatividade** - $\mathbf{u} + (\mathbf{v} + \mathbf{w}) = (\mathbf{u} + \mathbf{v}) + \mathbf{w}$
2. **Comutatividade** - $\mathbf{u} + \mathbf{v} = \mathbf{v} + \mathbf{u}$
3. **Elemento Identidade Vetorial** - $\mathbf{v} + \mathbf{0} = \mathbf{v}$
4. **Elemento Inverso** - $\mathbf{v} + (-\mathbf{v}) = \mathbf{0}$
5. **Associatividade na Multiplicação entre Vetores e Escalares** - $a(b\mathbf{v}) = (ab)\mathbf{v}$
6. **Elemento Identidade Escalar** - $1\mathbf{v} = \mathbf{v}$.
7. **Propriedade Distributiva de um Escalar para Soma de Vetores** - $a(\mathbf{u} + \mathbf{v}) = a\mathbf{u} + a\mathbf{v}$
8. **Propriedade Distributiva de um Vetor para Soma de Escalares** - $\mathbf{u}(a + b) = a\mathbf{u} + b\mathbf{u}$.

Dado essa definição, é possível definir um *espaço vetorial normado*, como sendo aquele equipado com uma *norma*, a qual define-se abaixo.

Definição 3. *Seja X um espaço vetorial. Desse modo, uma norma em X é uma função $|\cdot| : X \rightarrow \mathbb{R}$ que satisfaz as seguintes propriedades.*

1. $|x| \geq 0 \quad \forall x \in X$
2. $|x| = 0 \iff x = 0$
3. $|\lambda x| = |\lambda||x| \quad \forall \lambda \in \mathbb{R} \quad \forall x \in X$
4. $|x + y| \leq |x| + |y| \quad \forall x, y \in X$.

Note que, com essa definição é possível que um espaço vetorial tenha várias normas. Por exemplo, tome o espaço $C[0, 1] = \{f[0, 1] \rightarrow \mathbb{R} \text{ contínuas}\}$. Desse modo, pode-se definir as duas normas abaixo.

- $|f| = \sup\{|f(x)| ; x \in [0, 1]\}$ é a norma do supremo da função em $C[0, 1]$.
- $|f| = \int_0^1 |f(x)| dx$ é a norma da função em $C[0, 1]$.

Além disso, é possível demonstrar que o espaço é métrico e, desse modo, define-se uma métrica chamada distância entre dois vetores, a qual é representada por $d(\mathbf{u}, \mathbf{v}) = |\mathbf{u} - \mathbf{v}|$ a qual induzida pela norma, por ser invariante a translação (isto é, $d(\mathbf{u}, \mathbf{v}) = d(\mathbf{u} + \mathbf{a}, \mathbf{v} + \mathbf{a})$ para quaisquer vetores $\mathbf{u}, \mathbf{v}, \mathbf{a}$) e absolutamente homogêneo (isto é, $d(b\mathbf{u}, b\mathbf{v}) = |b|d(\mathbf{u}, \mathbf{v})$, para cada b escalar e quaisquer vetores \mathbf{u}, \mathbf{v}).

Por fim, para determinar um *espaço vetorial normado completo*, é preciso definir a sequência de Cauchy.

Definição 4. *Uma sequência é dita de Cauchy se, para cada número real positivo ϵ , existe um inteiro positivo N , tal que para todos os números naturais $m, n > N$, têm-se $|\mathbf{x}_m - \mathbf{x}_n| < \epsilon$.*

Dado essa definição, diz-se que um espaço vetorial é normado *completo*, se, para cada sequência de Cauchy em um dado espaço métrico X , existe um $x \in X$, tal que $\lim_{n \rightarrow \infty} |\mathbf{x}_n - \mathbf{x}| = 0$.

Dessa maneira, note que, por exemplo, o espaço $P[0, 1]$ de todos polinômios definidos em $[0, 1]$ não é um espaço de Banach, pois, apesar de ser normado, uma vez que contém $C[0, 1]$, não é completo, dado que não é qualquer sequência de Cauchy nesse espaço que converge para um ponto desse.

Nesse sentido, sabe-se que qualquer função contínua pode ser aproximada por polinômios. Desse modo, utilizando a série de Taylor centrada em $x = 0$, têm-se que:

$$e^x = \sum_{n=0}^{+\infty} \frac{x^n}{n!} \quad \forall x \in [0, 1]. \quad (3.3)$$

Ora, apesar de $\sum_{n=0}^k \frac{x^n}{n!}$ ser uma sequência de Cauchy em $P[0, 1]$, essa converge para a função e^x em $x \in [0, 1]$, que não é polinômio.

Com esse conhecimento, pode-se definir o que é um espaço de Hilbert, como feito a seguir.

Definição 5. *O Espaço de Hilbert é um Espaço de Banach com norma induzida por produto interno.*

Dado essa definição, define-se o que é produto interno a seguir.

Definição 6. *Seja V um espaço vetorial sobre \mathbb{K} um subcorpo dos números complexos \mathbb{C} . Desse modo, para quaisquer vetores $u, v, w \in V$ e todos os escalares $k \in \mathbb{K}$, o produto interno como a função binária*

$$\langle \cdot, \cdot \rangle : V \times V \rightarrow \mathbb{K}$$

com as seguintes propriedades:

1. *Linearidade* - $\langle u + v, w \rangle = \langle u, w \rangle + \langle v, w \rangle$
2. *Homogeneidade* - $\langle ku, v \rangle = k\langle u, v \rangle$
3. *Positividade* - $\langle v, v \rangle \geq 0$ e $\langle v, v \rangle = 0 \iff v = 0$
4. *Simetria Hermitiana* - $\langle u, v \rangle = \overline{\langle v, u \rangle}$.

Em particular, se $\mathbb{K} = \mathbb{R}$, então a propriedade (4) transforma-se em $\langle u, v \rangle = \langle v, u \rangle$, $\forall u, v \in V$.

3.2 Medida de Lebesgue

Para provar a existência e unicidade de solução da Equação de Poisson pela Análise Variacional é necessário evoluir a noção de medida de integração aprendida no curso de engenharia para Medida em Lebesgue, pois a medida em Riemann impede, por exemplo, que o Teorema Fundamental do Cálculo seja aplicado para funções diferenciáveis para as quais suas derivadas não fossem integráveis em Riemann. Desse modo, define-se a medida de Lebesgue a seguir.

Definição 7. *Seja (X, Σ) e (Y, T) espaços mensuráveis, isto é, X e Y conjuntos equipados com os respectivos σ -álgebras Σ e T . Desse modo, a função $f : X \rightarrow Y$ é dita mensurável em Lebesgue se, para cada $E \in T$ têm-se*

$$f^{-1}(E) := \{x \in X \mid f(x) \in E\} \in \Sigma. \quad (3.4)$$

Para que a definição acima seja compreendida em totalidade, defini-se σ -álgebra a seguir.

Definição 8. *Seja X um conjunto e seja $P(X)$ o conjunto de todos subconjuntos de X (incluindo o conjunto vazio e o próprio conjunto X). Então, o subconjunto $\Sigma \subseteq P(X)$ é chamado de σ -álgebra se, e somente se, satisfaz as três propriedades a seguir:*

1. *X está em Σ é um conjunto universal (conjunto que contém todos objetos incluindo ele mesmo).*
2. *Σ é fechado para complementação, ou seja, se um conjunto C está em σ , então seu complemento também estará.*
3. *Σ é fechado para uniões contáveis, ou seja, se os conjuntos C_1, C_2, C_3, \dots estão em Σ , então a união dos conjuntos também estará.*

3.3 Espaços de Sobolev

Primeiramente, define-se $L^2(\Omega)$ como o conjunto de funções $u: \Omega \subset \mathbb{R}^n \rightarrow \mathbb{R}$ que tem os suas p-ésima potências integráveis em medida de Lebesgue, isto é,

$$L^2(\Omega) = \{u : \Omega \subset \mathbb{R}^n \rightarrow \mathbb{R}; \int_{\Omega} \|u\|^2 dV < \infty\}. \quad (3.5)$$

Define-se seu produto interno em $L^2(\Omega)$ como:

$$\langle u, v \rangle_{L^2} = \int_{\Omega} uv dV. \quad (3.6)$$

Desse modo, um espaço de Sobolev $H^1(\Omega)$ é um espaço das funções $u \in L^2(\Omega)$ para as quais existe a função teste $v \in L^2(\Omega)$ de forma que

$$\int_{\Omega} u \frac{\partial \phi}{\partial x_i} dV = - \int_{\Omega} v \phi dV, \quad \forall \phi \in C_c^\infty(\Omega). \quad (3.7)$$

Em que, por sua vez, o conjunto C_c^∞ é o conjunto das funções que são C^∞ as quais se anulam fora de um conjunto fechado e limitado (compacto) $K \subset \Omega$. Nesse contexto, defini-se $v = \frac{\partial u}{\partial x_i}$ uma *derivada fraca* de u em relação à x_i . Desse modo, pode-se sintetizar $H^1(\Omega)$ por

$$H^1(\Omega) = \{u \in L^2(\Omega); v \in L^2(\Omega); i = 1, \dots, n\}. \quad (3.8)$$

Pode-se munir esse espaço com o seguinte produto interno:

$$\langle u, v \rangle_{H^1} = \langle u, v \rangle_{L^2} + \sum_i^n \langle v_i, \frac{\partial v}{\partial x_i} \rangle_{L^2}. \quad (3.9)$$

3.4 Teorema de Lax-Milgram

Dado o resumo teórico das ferramentas anteriores, é preciso conhecer o Teorema de Lax-Milgram para a solução do problema desta monografia. Porém, antes do teorema ser enunciado, é fundamental definir o conceito de **forma bilinear** e sua **coercividade** e **continuidade**.

Definição 9. *Uma forma bilinear definida em um espaço vetorial V (sobre um corpo K) é uma função $b: V \times V \rightarrow K$ linear em ambas as variáveis (como mostrado*

pelas condições a seguir), com as seguintes propriedades para todos $u, v, w \in V$ e para todo $\lambda \in \mathbb{R}$.

1. $b(u + w, v) = b(u, v) + b(w, v)$
2. $b(u, v + w) = b(u, v) + b(u, w)$
3. $b(\lambda u, v) = b(u, \lambda v) = \lambda b(u, v)$.

Em particular, o produto interno em \mathbb{R}^n é um exemplo de forma bilinear.

Definição 10. Seja b uma aplicação bilinear e H um espaço de Hilbert, tal que $b : L^2 \times H^1 \rightarrow \mathbb{R}$. Então, dizemos que b é **coerciva** se

$$\exists \alpha > 0, \forall x \in H, b(x, x) \geq \alpha |x|^2. \quad (3.10)$$

Definição 11. Seja b uma aplicação bilinear e H um espaço de Hilbert, tal que $b : L^2 \times H^1 \rightarrow \mathbb{R}$. Então, dizemos que b é **contínua** se

$$\exists \alpha > 0, \forall x, y \in H, b(x, y) \leq \alpha |x| |y|. \quad (3.11)$$

Desse modo, com os conceitos de continuidade e coercividade de formas bilineares bem definidas, pode-se enunciar o Teorema de Lax-Milgram.

Teorema 3.4.1. Seja H um Espaço de Hilbert e $b : L^2 \times H^1 \rightarrow \mathbb{R}$ e $f \in H^1$, então $\forall u \in H, b(x, u) = f(u)$ existe apenas uma única solução x e a aplicação $f \rightarrow x$ é linear e contínua de H^1 para H .

Dessa maneira, têm-se o teorema fundamental para resolução do problema deste texto.

3.5 Existência e Unicidade da Equação de Poisson

Utilizando o resultado (3.6) do espaço de Sobolev $H^1(\Omega)$ e, desse modo, aplicando a função teste v na equação (3.1), transforma-se a equação de Poisson em sua forma fraca:

$$-\int_{\Omega} \Delta T v \, dV = \int_{\Omega} f v \, dV \quad (3.12)$$

Pela integração por partes, a equação (3.12) é desenvolvida para:

$$\int_{\Omega} \nabla T \cdot \nabla v \, dV - \int_{\partial\Omega} (\nabla T \cdot \mathbf{n}) v \, dS = \int_{\Omega} f v \, dV. \quad (3.13)$$

Pela condição de contorno (3.2), têm-se $\nabla T \cdot \mathbf{n} = g - \alpha T$, e, portanto, substituindo esta relação em (3.13), obtêm-se:

$$\int_{\Omega} \nabla T \cdot \nabla v \, dV + \int_{\partial\Omega} \alpha T v \, dS = \int_{\Omega} f v \, dV + \int_{\partial\Omega} g v \, dS \quad (3.14)$$

$$b(T, v) = f(v). \quad (3.15)$$

Nesse ponto, basta provar que $b(T, v)$ é contínua e coerciva para demonstrar que o problema tem solução única.

Primeiro, será provada a **continuidade** de $b(T, v)$. Para isso, será enunciada duas desigualdades importantes para a demonstração. Nesse sentido, têm-se:

1. **Desigualdade de Hölder** - Sejam $f, g \in H^1$ e $1 \leq p, q \leq \infty$, então

$$\int_{\Omega} f g \, dx \leq |f|_p |g|_q \quad (3.16)$$

em que

$$\frac{1}{p} + \frac{1}{q} = 1$$

2. **Desigualdade de Poincaré** - Para todo $u \in H^1$, têm-se

$$|u|_{L^2} \leq c |\nabla u|_{L^2}. \quad (3.17)$$

Desse modo, pelas desigualdades (3.16) e (3.17), têm-se

$$b(T, v) \leq |u|_{H^1} |v|_{H^1} + \alpha |u|_{H^1} |v|_{H^1}, \quad \alpha \in \mathbb{R}. \quad (3.18)$$

$$b(T, v) \leq (1 + \alpha)|u|_{H^1}|v|_{H^1}. \quad (3.19)$$

Logo, pela equação (3.19), a forma bilinear $b(T, v)$ é contínua.

Agora, será provado por contradição que $b(T, v)$ é coerciva. Nesse sentido, suponha que $b(T, v)$ não é coerciva. Sendo assim, deverá existir uma sequência $v_n \subset H^1$, tal que $|v_n|_H^1 = 1$ e $b(v_n, v_n) \rightarrow 0$.

Como essa sequência v_n é limitada, logo é necessário haver uma subsequência dela v_{n_i} em H^1 que converge de forma fraca da seguinte maneira:

$$\int_{\Omega} (v - v_{n_i})\omega + \int_{\Omega} \nabla(v - v_{n_i}) \cdot \nabla\omega \rightarrow 0; \forall \omega \in H^1. \quad (3.20)$$

Implicando em:

$$\int_{\Omega} |\nabla v|^2 = \lim_{i \rightarrow \infty} \int_{\Omega} \nabla v_{n_i} \cdot \nabla v \leq \lim_{i \rightarrow \infty} \left(\int_{\Omega} |\nabla v_{n_i}|^2 \right)^{1/2} \left(\int_{\Omega} |\nabla v|^2 \right)^{1/2} \quad (3.21)$$

$$|v|_{H^1} \leq \lim_{i \rightarrow \infty} |v_{n_i}|_{H^1} \quad (3.22)$$

Além disso, observe que:

$$b(v_n, v_n) = \int_{\Omega} |\nabla v_n|^2 + \int_{\partial\Omega} \alpha v_n^2 \rightarrow 0. \quad (3.23)$$

Implicando em:

$$\int_{\Omega} |\partial v_n|^2 \rightarrow 0. \quad (3.24)$$

Por (3.22), quando $n_i \rightarrow \infty$, têm-se $\int_{\Omega} |\nabla v|^2 = 0$, o que leva a $|v|_{H^1} = |v|_{L^2}$. Com isso, note que, utilizando $v_{n_i} \rightarrow v$ em L^2 , com (3.23), têm-se $|v|_{L^2} = \lim_{i \rightarrow \infty} |v_{n_i}|_{L^2} = 1$.

Desse modo, a norma de v em L^2 é igual a 1, porém, note que, pela mesma equação (3.23), também têm-se:

$$\int_{\partial\Omega} \alpha v_{n_i} = 0. \quad (3.25)$$

o que leva a conclusão de que $v = 0$, dado que $\alpha > 0$.

Dessa forma, por contradição, $b(T, v)$ é uma forma bilinear coerciva. Logo, por ser contínua e coerciva, o Teorema de Lax-Milgram garante existência de uma única solução única para o problema.

Por fim, note que, assumindo que o contorno $\partial\Omega$ seja suave o suficiente, a solução fraca única implica em uma única solução única forte (condição de máxima regularidade), como será tratado neste texto.

Caso não exista uma condição de máxima regularidade, a solução única que foi provada para formulação fraca não implicará em solução forte na forma da equação diferencial parcial.

Capítulo 4

Estudo Numérico

4.1 O Método dos Elementos Finitos

Provada a existência de uma solução única para o problema discutido neste texto, será apresentado, nesta seção, o Método dos Elementos Finitos para resolução do problema.

4.1.1 O Método de Galerkin

Dentre os métodos numéricos de elementos finitos para resolução de equações diferenciais, destaca-se o método de Galerkin por sua versatilidade e fácil aplicação dos conceitos estudados anteriormente na parte analítica.

Desse modo, considere a formulação fraca de um problema em um par de espaços de Hilbert V e W , então, é proposto o problema representado em (2.17) do capítulo anterior:

Encontrar $T \in V$, tal que $b(T, v) = f(v)$ para cada $v \in W$.

Para resolução numérica, é preciso particionar os espaços V e W para que tenham dimensão finita. Desse modo, escolhendo $V_n \subset V$ de dimensão n e $W_n \subset W$ de dimensão m , o problema original é reescrito da seguinte forma:

Encontrar $T_n \in V_n$, tal que $b(T_n, v_m) = f(v_m)$ para cada $v_m \in W_m$.

Nesse ponto, note que a equação continua essencialmente a mesma, porém a redução da dimensão dos espaços vetoriais permite que T_n seja escrito como uma combinação linear finita de uma base de vetores de V_n .

Agora, note que existe uma propriedade sutil, mas de grande importância, ao rescrever a equação em dimensão finita: a ortogonalidade do erro da aproximação em relação aos subespaços finitos definidos. Nesse contexto, denotando o erro de aproximação por $\epsilon_n = T - T_n$, repare que

$$b(\epsilon_n, v_m) = b(T, v_m) - b(T_n, v_m) = f(v_m) - f(v_m) = 0, \forall v_m \in W_m. \quad (4.1)$$

Finalmente, dado que o objetivo da aproximação do problema é construir um sistema de equações lineares para serem resolvidas computacionalmente, considere T^1, T^2, \dots, T^n uma base para o espaço V_n e v^1, v^2, \dots, v^m uma base para o espaço W_m , então, deve-se achar $T_n \in V_n$, tal que

$$b(T_n, v^j) = f(v^j), \quad j = 1, \dots, m. \quad (4.2)$$

Desse modo, pode-se expandir T_n em uma base de V_n como $T_n = \sum_{i=1}^n x^i v^i$ e substituir em (3.2), resultando em:

$$b\left(\sum_{i=1}^n x^i T^i, v^j\right) = \sum_{i=1}^n x^i b(T^i, v^j) = f(v^j) \quad j = 1, \dots, m. \quad (4.3)$$

Nesse contexto, repare que a equação (3.3) é um sistema matricial $A_{ij}x = f_j$, em que $A_{ij} = b(T^i, v^j)$ e $f_j = f(v^j)$.

Por fim, note que, dado a forma que foi definida a entrada da matriz A , ela será simétrica se $V = W$, a forma bilinear $b(., .)$ for simétrica, $n = m$, $V_n = W_n$ e $T^i = v^j$

para todo $i = j = 1, \dots, n = m$.

4.1.2 Representação Gráfica do Método

Antes de prosseguir para maiores detalhes matemáticos em relação ao método, é válido estudar a representação gráfica da discretização do domínio do problema. Para isso, considere um domínio genérico $\Omega = [0, 1] \times [0, 1]$, como representado na figura 4.1.

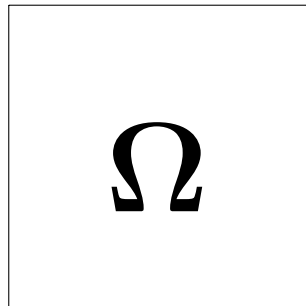


Figura 4.1: Domínio contínuo $\Omega = [0, 1] \times [0, 1]$

Uma discretização possível é, por exemplo, subdividir o domínio em elementos quadrangulares de igual área, como visto na figura 4.2.

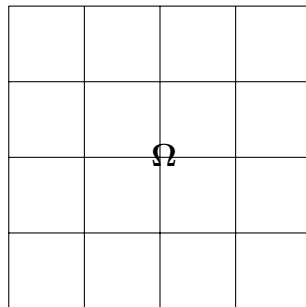


Figura 4.2: Domínio $\Omega = [0, 1] \times [0, 1]$ discretizado em 16 elementos

Note que, para essa discretização de 16 elementos da figura 4.2, cada elemento possui lado igual a $(1/\sqrt{16}) = 0.25$, o qual chama-se de passo neste contexto.

Ainda sobre a discretização, é importante definir três conceitos: graus de liberdade, células e malha.

- Graus de liberdade - é definido pelos vértices do domínio discretizado.

- Célula - é equivalente ao elemento (nesse caso, contém 4 graus de liberdade).
- Malha - é a união de células e seus graus de liberdade (matematicamente, $\Omega = \cup_e \Omega^e$).

4.1.3 Elementos e Funções Base

Neste momento, serão definidas as funções bases v^i para que a equação (4.3) seja possível de resolver em um computador. Para isso, é feito o seguinte desenvolvimento, a partir da integração por partes discutida anteriormente:

$$\sum_{j=1}^N (\nabla v^i, \nabla v^j) T_j = (v^i, f). \quad (4.4)$$

Utilizando a quadratura gaussiana, pode-se escrever as integrais que representam os dois lados da equação (4.4) como:

$$(\nabla v^i, \nabla v^j) \approx \sum_{k=1}^K \sum_{q=1}^Q J_k^{-1} \nabla v^i \cdot J_k^{-1} \nabla v^j x^i \det(J_q) w_q. \quad (4.5)$$

$$(v^i, f^j) \approx \sum_{k=1}^K \sum_{q=1}^Q J_k^{-1} v^i J_k^{-1} f^j \det(J_q) w_q. \quad (4.6)$$

em que J é a matriz jacobiana para transformação de coordenadas de integração, e w é o peso da quadratura.

Além disso, note que a equação (4.5) pode ser considerada uma extensão da regra de Simpson para o cálculo de integrais de uma variável.

Neste ponto, para entender a construção das funções base v para resolução do método, é mais simples começar pela construção de uma base em apenas uma dimensão. Nesse sentido, dado um domínio $x = [0, 1]$, pode-se mover seus pontos para um sistema de coordenadas locais

$$\xi = \frac{x - x_i}{x_{i+1} - x_i}. \quad (4.7)$$

Dado $T(x) = \sum_{i=1}^n c_i v_i$ e utilizando a equação (4.7), têm-se que

$$T(\xi) = c_1 + c_2 \xi. \quad (4.8)$$

Como $\xi_{1,2} = 0, 1$, têm-se que

$$T_1 = c_1. \quad (4.9)$$

$$T_2 = c_1 + c_2 \Rightarrow c_2 = -T_1 + T_2. \quad (4.10)$$

Logo, têm-se:

$$T(\xi) = T_1 + (-T_1 + T_2) \xi \Rightarrow T(\xi) = T_1(1 - \xi) + T_2 \xi \Rightarrow T(\xi) = T_1 N_1(\xi) + T_2 N_2(\xi). \quad (4.11)$$

Desse modo, as funções base lineares em uma dimensão são $N_1 = (1 - \xi)$ e $N_2 = \xi$ e para expandir as funções para dimensões superiores, basta fazer o produto tensorial entre as funções. Por exemplo, em duas dimensões, têm-se a seguinte base retangular de elementos lineares:

$$N_{1,1} = (1 - \xi)(1 - \eta)$$

$$N_{1,2} = (\xi)(1 - \eta)$$

$$N_{2,1} = (\eta)(\xi)$$

$$N_{2,2} = (1 - \xi)(\eta).$$

Com as funções base definidas, resolve-se o sistema linear da definido na equação (4.4), considerando as equações (4.5) e (4.6), e, portanto, resolvendo a equação de Poisson numericamente.

Capítulo 5

Resultados

5.1 Estudo de Caso

Nesta seção, será aplicado o método de Galerkin em um problema de Transferência de Calor para contextualizar e demonstrar de forma prática o que foi estudado neste texto. Nesse sentido, considere o problema abaixo:

$$-\Delta T = f \text{ em } \Omega \quad (5.1)$$

$$T = 0 \text{ em } \partial\Omega \quad (5.2)$$

$$\Omega = [0, 1] \times [0, 1] - [0.45, 0.55] \times [0.45, 0.55] . \quad (5.3)$$

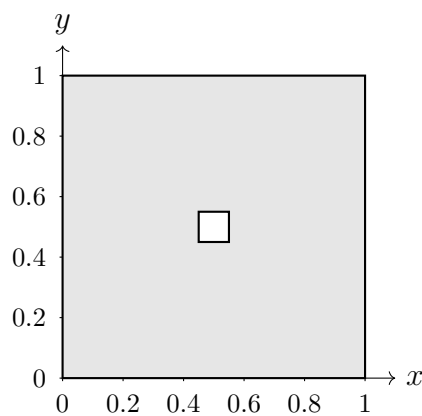


Figura 5.1: Domínio $\Omega = [0, 1] \times [0, 1] - [0.45, 0.55] \times [0.45, 0.55]$

5.1.1 Solução Manufaturada

Para resolver a equação (5.1) é necessário definir f . Para isso, tome a equação (5.4) como solução manufaturada da equação diferencial, respeitando (5.2) e (5.3).

$$T(x, y) = x(1 - x)y(1 - y)(x - 0.45)(x - 0.55)(y - 0.45)(y - 0.55). \quad (5.4)$$

Desse modo, como $f = -\Delta T$, têm-se:

$$\begin{aligned} f(x, y) = & -2x(x - 1)(x - 0.55)(x - 0.45)(y(y - 1) + y(y - 0.55) + y(y - 0.45) \\ & + (y - 1)(y - 0.55) + (y - 1)(y - 0.45) + (y - 0.55)(y - 0.45)) \\ & - 2y(y - 1)(y - 0.55)(y - 0.45)(x(x - 1) + x(x - 0.55) + x(x - 0.45) \\ & + (x - 1)(x - 0.55) + (x - 1)(x - 0.45) + (x - 0.55)(x - 0.45)). \end{aligned} \quad (5.5)$$

5.1.2 Solução Numérica

Para solução numérica do problema, utilizou-se a função base linear de elementos retangulares e a quadratura gaussiana descritas no capítulo anterior.

Dessa maneira, analisa-se a convergência para a solução analítica do problema (equação (5.4)), dado o número de elementos na malha.

Nesse sentido, seja s a solução aproximada e s_a sua solução analítica, desse modo, considera-se um erro relativo na medição como sendo $e = \sqrt{\frac{(s - s_a, s - s_a)}{(s_a, s_a)}}$

Seguem os casos analisados

1. 64 elementos (passo $h = 0.125$ na malha)

Verificação com o Método de Soluções Manufacturadas (MMS)
Erro percentual: 0.09889089781136184
Solução no centro $x=0.5, y=0.5$ é $3.906250000000055e-07$.

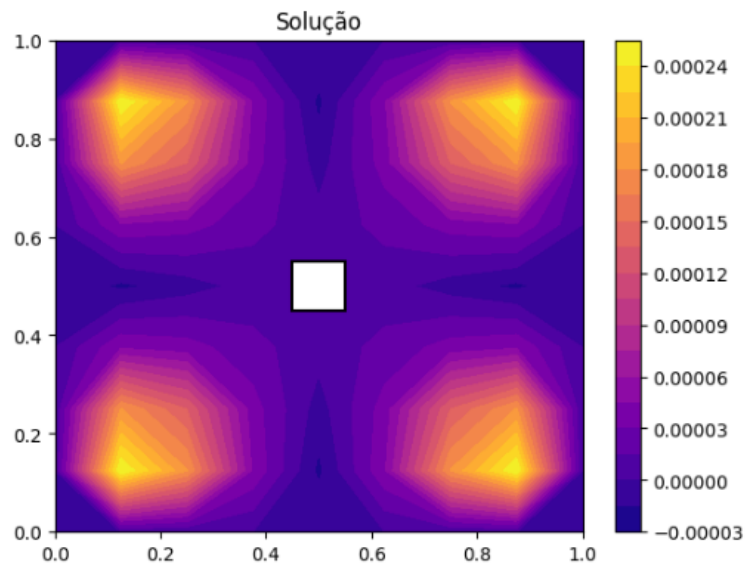


Figura 5.2: Resultado para 64 elementos

2. 256 elementos (passo $h = 0.0625$ na malha)

Verificação com o Método de Soluções Manufacturadas (MMS)
Erro percentual: 0.02159367700224443
Solução no centro $x=0.5, y=0.5$ é $3.906250000000055e-07$.

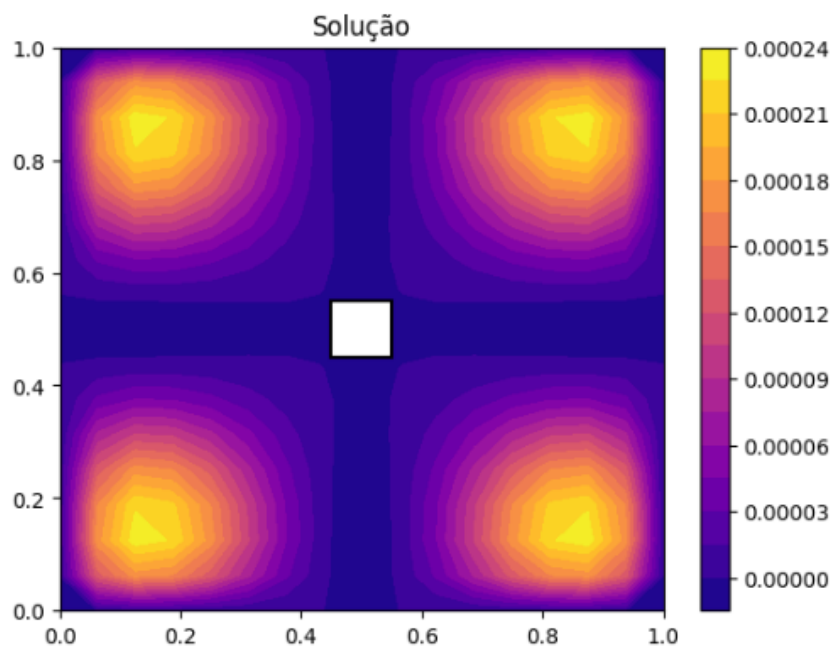


Figura 5.3: Resultado para 256 elementos

3. 1024 elementos (passo $h = 0.03125$ na malha)

Verificação com o Método de Soluções Manufacturadas (MMS)
Erro percentual: 0.006927302945256429
Solução no centro $x=0.5, y=0.5$ é $3.9062500000000055e-07$.

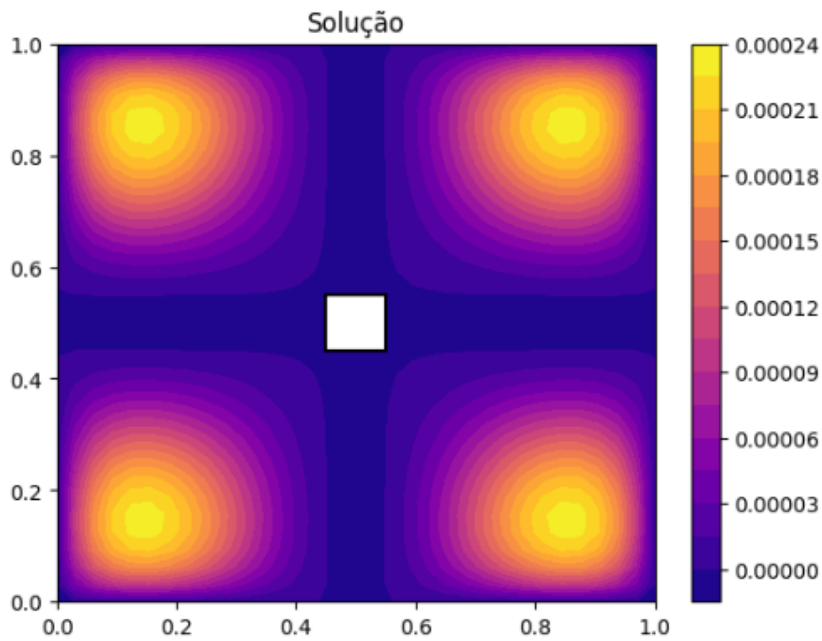


Figura 5.4: Resultado para 1024 elementos

4. 4096 elementos (passo $h = 0.015625$ na malha)

Verificação com o Método de Soluções Manufacturadas (MMS)
Erro percentual: 0.004878613617082901
Solução no centro $x=0.5, y=0.5$ é $3.9062500000000055e-07$.

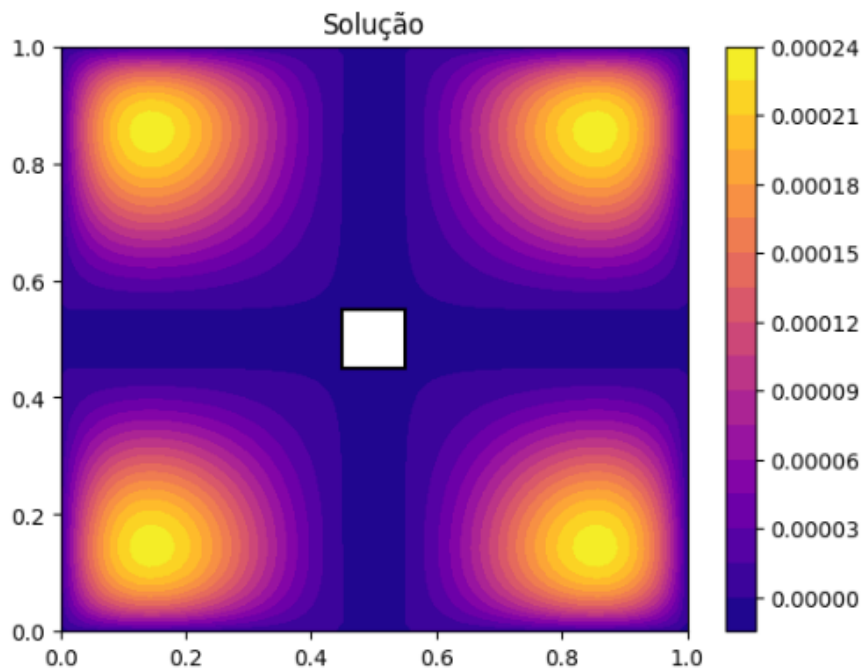


Figura 5.5: Resultado para 4096 elementos

5. 16384 elementos (passo $h = 0.0078125$ na malha)

Verificação com o Método de Soluções Manufacturadas (MMS)
Erro percentual: 0.004637279755336777
Solução no centro $x=0.5, y=0.5$ é $3.906250000000055e-07$.

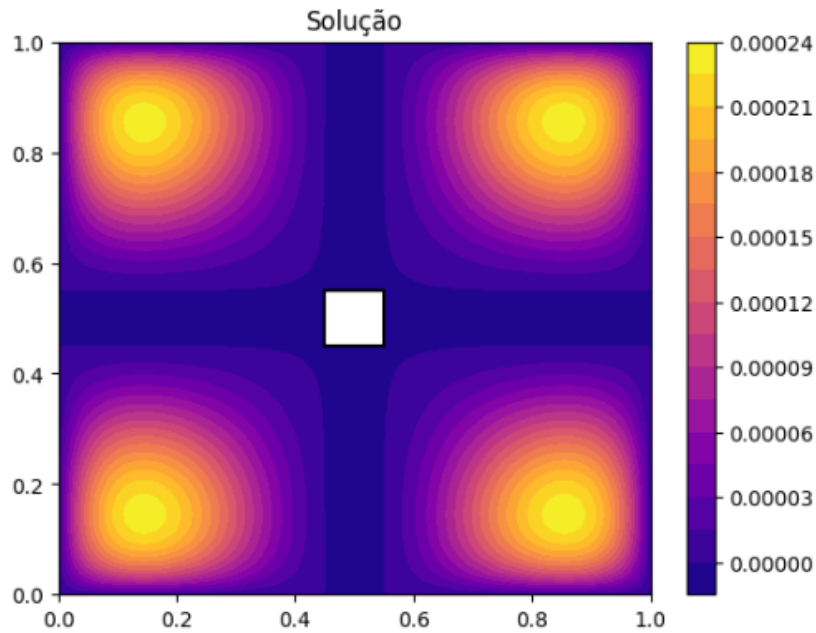


Figura 5.6: Resultado para 16384 elementos

Erro Relativo versus Elementos

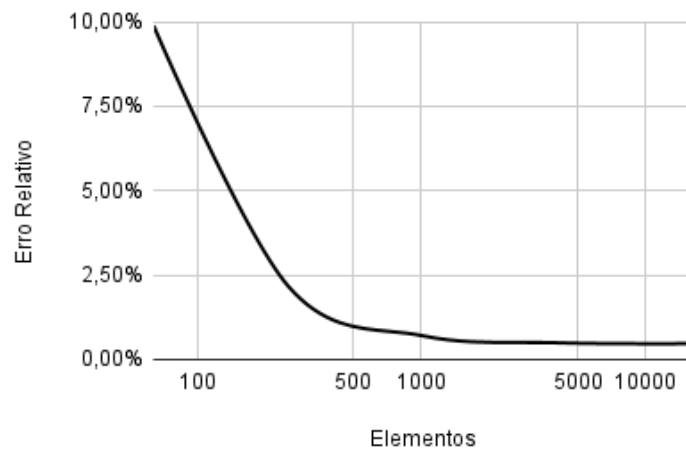


Figura 5.7: Erro relativo em função do número de elementos, com eixo horizontal em escala logarítmica

Note que o erro da solução numérica implementada converge para um erro $e \approx 0.46\%$, o que pode ser considerada uma aproximação boa dada pelo método.

Capítulo 6

Conclusões e Trabalhos Futuros

Durante o texto, foram estudados diversos conceitos matemáticos, os quais possibilitaram um tratamento rigoroso para a equação de Poisson. Dentre eles, destacam-se: o Teorema de Lax-Milgram (para a prova de solução fraca única) e o Método de Galerkin (para resolução numérica das equações diferenciais).

Nesse sentido, verifica-se que a teoria da Análise Variacional, trabalhando com a ideia de solução fraca das equações diferenciais parciais, é uma ferramenta extremamente poderosa para provar a existência de soluções para tais equações.

Além disso, foi observado que o Método dos Elementos Finitos produz resultados bastante acurados e precisos para resoluções de problemas de equações diferenciais parciais.

Por fim, para trabalhos futuros, recomenda-se um estudo dos métodos apresentados para resolução do problema de transferência de calor em estado transiente, o qual exige ainda mais sofisticação para ser abordado.

Referências Bibliográficas

- [1] EVANS, L., *Partial Differential Equations*. American Mathematical Society, 1993.
- [2] INCROPERA, FRANK; DEWITT, D., *Fundamentos de Transferência de Calor e de Massa*. LTC, 2014.
- [3] NUSSENZVEIG, M., *Curso de Física Básica: Fluidos, Oscilações e Ondas, Calor (Volume 2)*. Editora Blucher, 2014.
- [4] BREZIS, H., *Functional analysis, Sobolev spaces, and partial differential equations*. Springer Publishing, 2010.
- [5] KREYSZIG, E., *Introductory Functional Analysis with Applications*. John Wiley & Sons, 1978.
- [6] BAZÁN A.; PEREIRA A.; FERNANDES, C., *Introdução aos Espaços de Banach*. 34º Colóquio Brasileiro de Matemática, Instituto de Matemática Pura e Aplicada, 2023.
- [7] HUGHES, T., *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Dover Publications, 2000.
- [8] OZISIK, N., *Heat Transfer: A Basic Approach*. McGraw-Hill Education, 1984.
- [9] REDDY, J. N., *An introduction to the finite element method*. Mcgraw-Hill Education, 2006.
- [10] AXLER, S., *Linear Algebra Done Right*. Springer Publishing, 2015.
- [11] BRENNER, S.; SCOTT, R. L., *The Mathematical Theory of Finite Element Methods*. Springer Publishing, 1994.

- [12] LEONI, G., *A First Course in Sobolev Spaces*. American Mathematical Society, 2009.
- [13] JUGEN, J., *Partial Differential Equations*. Springer Publishing, 2007.
- [14] THOMÉE, S. L. V., *Partial Differential equations with Numerical Methods*. Springer Publishing, 2003.
- [15] DAVIS, P., *Methods of Numerical Integration*. Dover Publications, 2007.

Apêndice A

Código Fonte

```
import numpy as np
from scipy.sparse import dok_matrix
from scipy.sparse.linalg import spsolve
import matplotlib.pyplot as plt
import sympy as sp

# -----
# Pontos de quadratura e pesos (1D e 2D)
quad = [(0.0, 1/6), (0.5, 4/6), (1.0, 1/6)]
quad_2d = [(x, y, w_x * w_y) for (x, w_x) in quad for (y, w_y) in quad]

# -----
# Funcoes base 1D
phi_0 = {"eval": lambda x: 1 - x, "nabla": lambda x: -1}
phi_1 = {"eval": lambda x: x, "nabla": lambda x: 1}
phi_function = [phi_0, phi_1]

# Funcoes base 2D
class Phi2D:
    def __init__(self, x_0, y_0):
        self.x_basis = phi_function[x_0]
        self.y_basis = phi_function[y_0]
```

```

def eval(self, x, y):
    return self.x_basis["eval"](x) * self.y_basis["eval"](y)

def nabla(self, x, y):
    return np.array([self.x_basis["nabla"](x) *
                    self.y_basis["eval"](y),
                    self.x_basis["eval"](x) *
                    self.y_basis["nabla"](y)])

basis_function = {0: Phi2D(0, 0), 1: Phi2D(1, 0), 2: Phi2D(0, 1), 3:
    Phi2D(1, 1)}

# -----
# Solucao manufaturada

def u_exact(x, y):
    # Solucao manufaturada
    return x*(1-x)*y*(1-y)*(x-0.45)*(x-0.55)*(y-0.45)*(y-0.55)

# Computa f
x_sym, y_sym = sp.symbols('x y')
u_expr = x_sym*(1 - x_sym) * y_sym*(1 - y_sym) * (x_sym - 0.45)*(x_sym -
    0.55) * (y_sym - 0.45)*(y_sym - 0.55)
uxx_expr = sp.diff(u_expr, x_sym, 2)
uyy_expr = sp.diff(u_expr, y_sym, 2)
laplacian_expr = uxx_expr + uyy_expr
f_expr = -laplacian_expr # f = -Delta u
f_lambda = sp.lambdify((x_sym, y_sym), f_expr, 'numpy')

# -----
# Malha e graus de liberdade

class DoF:
    def __init__(self, x, y, ind=-1):

```

```

    self.x, self.y = x, y
    self.ind = ind

def __add__(self, other):
    return DoF(self.x + other.x, self.y + other.y)

def __sub__(self, other):
    return DoF(self.x - other.x, self.y - other.y)

def __rmul__(self, other):
    return DoF(other * self.x, other * self.y)

class Cell:
    def __init__(self, origem, direita, cima, cima_direita):
        self.dofs = [origem, direita, cima, cima_direita]

    def quadrature_point(self, x, y):
        # Referencia do dominio
        point = (self.dofs[0] +
                x * (self.dofs[1] - self.dofs[0]) +
                y * (self.dofs[2] - self.dofs[0]))
        return point.x, point.y

class Mesh:
    def __init__(self, x_min=0.0, x_max=1.0, y_min=0.0, y_max=1.0,
                 step_size=0.0078125,
                 hole_x_min=0.45, hole_x_max=0.55, hole_y_min=0.45,
                 hole_y_max=0.55):
        self.x_min, self.x_max = x_min, x_max
        self.y_min, self.y_max = y_min, y_max
        self.h = step_size
        # Define o centro do buraco
        self.hole_x_min = hole_x_min
        self.hole_x_max = hole_x_max

```

```

self.hole_y_min = hole_y_min
self.hole_y_max = hole_y_max
self.dofs, self.cells = [], []

def decompose(self):
    # Cria pontos na malha
    x_range = np.arange(self.x_min, self.x_max, self.h)
    x_range = np.append(x_range, [self.x_max])
    self.x_num = len(x_range)
    y_range = np.arange(self.y_min, self.y_max, self.h)
    y_range = np.append(y_range, [self.y_max])
    self.y_num = len(y_range)
    self.x_coord, self.y_coord = np.meshgrid(x_range, y_range)
    x_list, y_list = self.x_coord.ravel(), self.y_coord.ravel()

    # Cria graus de liberdade
    for i, (x, y) in enumerate(zip(x_list, y_list)):
        self.dofs.append(DoF(x, y, ind=i))

    # Cria celulas
    for i, dof in enumerate(self.dofs):
        if dof.x != self.x_max and dof.y != self.y_max:
            cell_nodes = [dof,
                          self.dofs[i + 1],
                          self.dofs[i + self.x_num],
                          self.dofs[i + self.x_num + 1]]

            # Passa celulas no furo
            if any((self.hole_x_min < node.x < self.hole_x_max) and
                  (self.hole_y_min < node.y < self.hole_y_max) for
                  node in cell_nodes):
                continue

            self.cells.append(Cell(*cell_nodes))

def plot_solution(self, solution):

```

```

# Cria solucao 2d
sol_2d = np.full(self.x_coord.shape, np.nan)
for dof in self.dofs:
    i = int(round((dof.y - self.y_min) / self.h))
    j = int(round((dof.x - self.x_min) / self.h))
    sol_2d[i, j] = solution[dof.ind]

# Mascara NaN
sol_masked = np.ma.masked_invalid(sol_2d)

# Colore NaN em branco
cmap = plt.cm.plasma.copy()
cmap.set_bad(color='white')

# Contorna a selecao NaN
fig, ax = plt.subplots()
cf = ax.contourf(self.x_coord, self.y_coord, sol_masked,
    levels=20, cmap=cmap)
fig.colorbar(cf, ax=ax)

# Adiciona detalhes no buraco
hole_width = self.hole_x_max - self.hole_x_min
hole_height = self.hole_y_max - self.hole_y_min
rect = Rectangle(
    (self.hole_x_min, self.hole_y_min),
    hole_width, hole_height,
    facecolor='white',
    edgecolor='black',
    linewidth=1.5,
    zorder=10
)
ax.add_patch(rect)

ax.set_title("Solucao")

```

```

plt.show()

def print_center_solution(self, solution):
    x_center = (self.x_max + self.x_min) / 2
    y_center = (self.y_max + self.y_min) / 2
    for dof in self.dofs:
        if abs(dof.x - x_center) < 1e-10 and abs(dof.y - y_center) <
            1e-10:
            print(f"Solucao no centro x={x_center}, y={y_center} e
                {solution[dof.ind]}".)
            break
    else:
        print(f"O centro x={x_center}, y={y_center} nao esta na malha")

# -----
# Resolucao da Equacao de Poisson pela solucao manufaturada

class Poisson:
    def __init__(self, mesh, f):
        self.mesh = mesh
        self.f = f # f manufaturada

    def assemble_system(self):
        num_dofs = len(self.mesh.dofs)
        A = dok_matrix((num_dofs, num_dofs), dtype=np.float64)
        F = np.zeros(num_dofs, dtype=np.float64)

        # Coloca o sistema na malha
        for cell in self.mesh.cells:
            for x, y, quad_weight in quad_2d:
                qp = cell.quadrature_point(x, y)
                for j, dof_j in enumerate(cell.dofs):
                    F[dof_j.ind] += (quad_weight * self.f(*qp) *

```

```

        basis_function[j].eval(x, y) *
        self.mesh.h**2)
    for i, dof_i in enumerate(cell.dofs):
        A[dof_i.ind, dof_j.ind] += quad_weight *
            basis_function[i].nabla(x, y).dot(
                basis_function[j].nabla(x, y))

# Identifica os nos que pertecem ao menos uma celula
used_nodes = set()
for cell in self.mesh.cells:
    for node in cell.dofs:
        used_nodes.add(node.ind)

# Coloca a condicao de contorno
for dof in self.mesh.dofs:
    if (abs(dof.x - self.mesh.x_min) < 1e-10 or abs(dof.x -
        self.mesh.x_max) < 1e-10 or
        abs(dof.y - self.mesh.y_min) < 1e-10 or abs(dof.y -
            self.mesh.y_max) < 1e-10 or
            (dof.ind not in used_nodes)):
        # Zero out the row in A
        _, nonzero_columns = A[dof.ind, :].nonzero()
        for j in nonzero_columns:
            A[dof.ind, j] = 0.0
            A[dof.ind, dof.ind] = 1.0
        # Coloca C.C em f manufacturada
        F[dof.ind] = u_exact(dof.x, dof.y)
return A.tocsr(), F

```

```
# -----
```

```
# Exemplo
```

```
h = 0.0625 # passo na malha
```

```

def exemplo():
    # Cria a malha com um furo quadrado unitario
    mesh = Mesh(step_size=h, hole_x_min=0.45, hole_x_max=0.55,
                hole_y_min=0.45, hole_y_max=0.55)
    mesh.decompose()

    # Coloca a solucao manufaturada

    pde = Poisson(mesh=mesh, f=f_lambda)
    A, F = pde.assemble_system()
    solution = spsolve(A, F)

    # Computa o erro
    used_nodes = set()
    for cell in mesh.cells:
        for node in cell.dofs:
            used_nodes.add(node.ind)

    num = 0.0
    denom = 0.0
    for ind in used_nodes:
        node = mesh.dofs[ind]
        u_ex = u_exact(node.x, node.y)
        num += (solution[ind] - u_ex)**2
        denom += u_ex**2
    error = np.sqrt(num) / np.sqrt(denom) if denom > 0 else np.inf
    print(f"Erro percentual: {error}")

    # Printa solucao no centro e plota a solucao
    mesh.print_center_solution(solution)
    mesh.plot_solution(solution)

print("Verificacao da Solucao")
exemplo()

```
